

11/23,
11/24 - *

KEF11-B CIS DIAG
CJKDHA0

AH-S433A-MC
FICHE 1 OF 3

OCT 1981
COPYRIGHT © 1981
MADE IN USA



11/23,
11/24-*

KEF11-B CIS DIAG
CJKDHA0

AH-S433A-MC
FICHE 2 OF 3

OCT 1981
COPYRIGHT © 1981
MADE IN USA



11/23.
11/24-*

KEF11-B CIS DIAG
CJKDHA0

AH-S433A-MC
FICHE 3 OF 3

OCT 1981
COPYRIGHT © 1981
MADE IN USA



.REM 8

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48

IDENTIFICATION

PRODUCT CODE: AC-S431A-MC
PRODUCT NAME: CJKDHA0 KEF11-B CIS DIAG.
PRODUCT DATE: JUNE, 1981
MAINTAINER: DIAGNOSTIC ENGINEERING

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C): 1981 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	DECX/11

49
50
51
52
53
54
55
56
57
58

HISTORY

REVISION
A

DATE
JUNE 1981

REASONS/CHANGES
FIRST RELEASE

59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93

TABLE OF CONTENTS

1.0	GENERAL PROGRAM INFORMATION
1.1	ABSTRACT
1.2	SYSTEM REQUIREMENTS
1.3	RELATED DOCUMENTS AND STANDARDS
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES
1.5	ASSUMPTIONS
2.0	OPERATING INSTRUCTIONS
2.1	LOADING AND STARTING PROCEDURE
2.2	PROGRAM OPTIONS
2.3	EXECUTION TIMES
3.0	ERROR INFORMATION
3.1	ERROR REPORTING PROCEDURES
3.2	ERROR HALTS
4.0	PERFORMANCE AND PROGRESS REPORTS
4.1	PERFORMANCE REPORTS
4.2	PROGRESS REPORTS
5.0	DEVICE INFORMATION TABLES
6.0	PROGRAM DESCRIPTION
6.1	PROGRAM EXECUTION CHARACTERISTICS
6.2	SUBTEST SUMMARIES
6.3	SPECIAL SUBROUTINE DESCRIPTION
7.0	LISTING

94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148

1.0 GENERAL PROGRAM INFORMATION

1.1 ABSTRACT

THIS DIAGNOSTIC VERIFIES THE FUNCTIONALITY OF THE KEF11-BA (CIS - COMMERCIAL INSTRUCTIONS SET) OPTION WITH BOTH MEMORY MANAGEMENT ENABLED AND DISABLED. IT ALLOWS THE USER TO CHECK OUT ANY COMBINATION OF CIS CHIPS. IN VIRTUALLY ALL CASES, FAULT ISOLATION IS TO THE CIS CHIP LEVEL. THE DIAGNOSTIC CAN BE RUN UNDER XXDP, ACT, AND APT MONITORS. THE SOFTWARE SWITCH REGISTER (LOCATION 176) IS USED.

THE CIS OPTION CONSISTS OF SIX MOS/LSI CONTROL CHIPS CONTAINED IN THREE 40-PIN HYBRID CARRIERS. ALL CIS INSTRUCTIONS ARE CHIP PARTITIONED, I.E. ALL THE CODE FOR A PARTICULAR INSTRUCTION IS CONTAINED ON A PARTICULAR CONTROL CHIP. THE EXCEPTION IS THAT ALL CIS INSTRUCTIONS MUST PASS THROUGH THE FIRST CIS CHIP (CONTROL CHIP DC303-004) BEFORE REACHING THEIR DESTINATION CHIP.

1.2 SYSTEM REQUIREMENTS

A. HARDWARE REQUIREMENTS

THIS DIAGNOSTIC IS DESIGNED TO RUN ON A SYSTEM IMPLEMENTING A DCF11-AA CHIP SET AND A MEMORY MANAGEMENT UNIT. ALSO REQUIRED ARE A SERIAL LINE INTERFACE, A CONSOLE TERMINAL, AND 24K OF MEMORY. THE FIRST CIS CHIP (DC303-004) MUST BE INSTALLED. ANY (OR NONE) OF THE OTHER CIS CHIPS MAY BE PRESENT.

B. SOFTWARE REQUIREMENTS

THE SOFTWARE ENVIRONMENTS UNDER WHICH THIS DIAGNOSTIC WILL OPERATE ARE APT SCRIPT, APT STAND-ALONE, ACT Q.V., AC AUTO-ACCEPT, ACT STAND-ALONE, XXDP CHAIN, XXDP STAND-ALONE, AND ABSOLUTE BINARY PAPER TAPE.

1.3 RELATED DOCUMENTS AND STANDARDS

THE FOLLOWING DOCUMENTS WERE USED OR REFERENCED DURING THE CREATION OF THIS DIAGNOSTIC:

DIAGNOSTIC ENGINEERING STANDARDS AND CONVENTIONS PROGRAMMING PRACTICES (DOC. NO. 175-003-009-02)
ACT11/XXDP PROGRAMMING SPECIFICATION (AUTOCAT-11-QZAUB-B-D)
BIC/BIN EXTENSION LIST
STANDARD APT SYSTEM TO PDP-11 DIAGNOSTIC INTERFACE (#APT/11-317-07-09)
DEC STD 168: PDP11 EXTENDED INSTRUCTIONS
CIS4.LIS

149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172

CIS5.LIS
CIS6.LIS
CIS7.LIS
CIS8.LIS
CIS9.LIS
CONTROL CHIP SPECIFICATION
DATA CHIP SPECIFICATION
SPMAC REFERENCE MANUAL

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

IT IS ASSUMED THAT THE CPU, MEMORY MANAGEMENT, MEMORY, SERIAL LINE INTERFACE, AND CONSOLE TERMINAL HAVE ALL PASSED THE PROPER DIAGNOSTICS.

1.5 ASSUMPTIONS

THE DIAGNOSTIC WILL ATTEMPT TO TEST ALL SIX CIS CHIPS UNLESS THE SOFTWARE SWR IS ALTERED TO INDICATE A SUBSET OF CIS CHIPS TO BE TESTED. SEE SECTION 2.2.

173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223

2.0 OPERATING INSTRUCTIONS

2.1 LOADING AND STARTING PROCEDURE

A. LOADING

USE STANDARD LOADING PROCEDURE FOR PDP-11 ABSOLUTE BINARY PAPER TAPES; OR XXDP MEDIA (FILES WITH .BIC OR .BIN EXTENSIONS ONLY).

B. STARTING

THE PROGRAM SHOULD ALWAYS BE STARTED AT 200. UPON START UP (RUNNING IN NON-AUTOMATIC MODE) THE DIAGNOSTIC INITIALIZES, PRINTS OUT ITS IDENTIFICATION (FIRST TIME ONLY), AND THEN PRINTS:

```
'SWR = 000000      NEW = ''
```

THE NUMBER FOLLOWING "SWR" IS THE CURRENT VALUE OF THE SOFTWARE SWITCH REGISTER. IF NO CHANGES ARE DESIRED IN THE SWITCH REGISTER THEN A CARRIAGE RETURN MAY BE HIT; IF CHANGES ARE DESIRED THEN A NEW VALUE MAY BE TYPED IN FOLLOWED BY A CARRIAGE RETURN. IF NOT ALL SIX CIS CHIPS ARE TO BE TESTED THEN THE SWR WILL HAVE TO BE CHANGED. SEE NEXT SECTION FOR SWR BIT ASSIGNMENTS.

2.2 PROGRAM OPTIONS

SWITCH REGISTER SETTINGS

LOCATION 176 IS USED FOR THE SOFTWARE SWITCH REGISTER (UNLESS RUNNING UNDER APT SCRIPT MODE). THE FOLLOWING ARE THE RESULTS OF SETTING THE RESPECTIVE SWR BITS:

BIT15	-HALT ON ERROR
BIT14	-SCOPE LOOP
BIT13	-INHIBIT ERROR TYPEOUT
BIT12	-NOT USED
BIT11	-INHIBIT SUBTEST ITERATIONS
BITS<10:8>	-NOT USED
BIT07	-MEMORY MANAGEMENT ALWAYS DISABLED
BIT06	-MEMORY MANAGEMENT ALWAYS ENABLED
BIT05	-TEST CIS CONTROL CHIP 9 (DC303-009)
BIT04	-TEST CIS CONTROL CHIP 8 (DC303-008)
BIT03	-TEST CIS CONTROL CHIP 7 (DC303-007)
BIT02	-TEST CIS CONTROL CHIP 6 (DC303-006)
BIT01	-TEST CIS CONTROL CHIP 5 (DC303-005)
BIT00	-TEST CIS CONTROL CHIP 4 (DC303-004)

224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267

AS CAN BE SEEN ABOVE, SWR BITS<5:0> ALLOW THE USER TO SELECT ANY COMBINATION OF CIS CONTROL CHIP(S) TO BE TESTED. TO SELECT A CIS CONTROL CHIP FOR TESTING, ITS CORRESPONDING BIT IN THE SWITCH REGISTER MUST BE SET. NOTE, HOWEVER, THAT IF SWR BITS<5:0> ARE ALL ZERO (DEFAULT MODE) THEN THE DIAGNOSTIC WILL TEST ALL SIX CIS

CONTROL CHIPS.

THE VALUE OF THE SOFTWARE SWITCH REGISTER MAY BE CHANGED DURING THE PROGRAM RUN. TO DO THIS A CONTROL/G IS TYPED (TYPE THE 'CTRL' AND 'G' KEYS SIMULTANEOUSLY); WITHIN A FEW SECONDS THE CURRENT VALUE OF THE SWR IS TYPED OUT AND THE PROGRAM WAITS FOR NEW INPUT.

NOTE: THE PROGRAM IS NOT RUNNING ANY TESTS WHILE WAITING FOR A NEW VALUE OF SWR TO BE INPUT.

2.3 EXECUTION TIMES

ALL TIMES WERE MEASURED ON A KDF11-A, RUNNING THE FULL COMPLEMENT OF CIS CONTROL CHIPS WITH MEMORY MANAGEMENT ENABLED AND DISABLED.

A. FIRST PASS (QV)

30 SECONDS

B. LONGEST TEST

50 SECONDS

C. ADDITIONAL TIME FOR UNITS

N/A

D. FULL PASS TIME (ITERATIONS)

90 SECONDS

268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316

3.0 ERROR INFORMATION

3.1 ERROR REPORTING PROCEDURES

WHEN A CIS INSTRUCTION FAILS, THE TYPEOUT GIVES THE ERROR PC, THE CONTROL CHIP NUMBER, AND WHAT KIND OF FAILURE OCCURRED. THE ERROR PC IS THE ACTUAL LOCATION OF THE ERROR CALL IN THE FAILING SUBTEST. THE CONTROL CHIP NUMBER REFERS TO WHICH CONTROL CHIP WAS BEING TESTED WHEN THE ERROR OCCURRED. THE CONTROL CHIP NUMBER WILL APPEAR AS 23-*04, 23-*05, 23-*06, 23-*07, 23-*08, OR 23-*09. THE * SIGNIFIES THE REVISION LEVEL OF THE CHIP AND SO MAY BE ANY DIGIT. THE LAST TWO DIGITS DISTINGUISHES ONE CIS CHIP FROM ANOTHER. CURRENTLY, THE SIX CIS CONTROL CHIPS ARE PACKAGED ON THREE HYBRID CARRIERS. THE PART NUMBERS FOR THE HYBRID ASSEMBLIES ARE: 5700002-00 (CIS CHIPS 23-*04 AND 23-*06), 5700003-00 (CIS CHIPS 23-*05 AND 23-*07) AND 5700004-00 (CIS CHIPS 23-*08 AND 23-*09). THE KINDS OF FAILURES REPORTED ARE:

'CIS INSTRUCTION FAILURE'
'CIS INTERRUPT FAILURE'
'NO TRAP ON ILLEGAL CIS CLASS OPCODE''

THE 'CIS INSTRUCTION FAILURE' IS THE MOST FREQUENTLY CALLED ERROR. IT USUALLY REFERS TO A CIS INSTRUCTION THAT PRODUCES THE WRONG CONDITION CODES, THE WRONG DESTINATION OUTPUT, OR THE WRONG GENERAL PURPOSE REGISTER CONTENTS. TO DETERMINE WHAT THE SPECIFIC FAILING INSTRUCTION IS, AND THE SPECIFIC TYPE OF FAILURE, LOOK IN THE DIAGNOSTIC AT THE CODE IN THE AREA OF THE ERROR PC.

3.2 ERROR HALTS

THREE UNEXPECTED FAULTS CAN ALSO OCCUR. THEY ARE:

'AN UNEXPECTED TIME-OUT TRAP''
'AN UNEXPECTED ILLEGAL AND RESERVED INSTRUCTION TRAP''
'POWER FAIL''

IN ALL THREE CASES THE PROGRAM HALTS IMMEDIATELY AFTER TYPING OUT THE APPROPRIATE MESSAGE. HITTING 'P' (PROCEED) WILL CAUSE THE DIAGNOSTIC TO START OVER IF THE FAULT WAS A POWER FAIL, AND TO CONTINUE THE DIAGNOSTIC FROM THE POINT AT WHICH IT WAS INTERRUPTED FOR THE OTHER TWO FAULTS. ANY OTHER UNEXPECTED INTERRUPTS OR TRAPS WILL BE CAUGHT BY THE TRAP CATCHER, WHICH ALSO HALTS THE DIAGNOSTIC.

SETTING SWR BIT15 WILL CAUSE THE THREE ERRORS LISTED IN SECTION 3.1 TO HALT AFTER TYPING OUT THE ERROR MESSAGE. HITTING 'P' (PROCEED) WILL CONTINUE THE DIAGNOSTIC.

317
318
319
320
321
322
323
324
325
326
327
328
329
330

4.0 PERFORMANCE AND PROGRESS REPORTS

4.1 PERFORMANCE REPORTS

N/A

4.2 PROGRESS REPORTS

AT THE END OF EACH PASS THE DIAGNOSTIC PRINTS OUT 'END PASS #'
FOLLOWED BY THE CURRENT PASS NUMBER.

331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381

5.0 DEVICE INFORMATION TABLES

THE CIS INSTRUCTIONS ARE PARTITIONED SUCH THAT NEARLY ALL THE MICRO-CODE FOR AN INSTRUCTION IS LOCATED ON ONE CHIP. (THE INSTRUCTIONS THAT RESIDE ON CONTROL CHIP DC303-004 ARE CONTAINED ENTIRELY ON THAT CHIP; ALL THE OTHER INSTRUCTIONS MUST FIRST PASS THROUGH CONTROL CHIP 4 SO THERE IS A SMALL AMOUNT OF MICRO-CODE FOR THE OTHER INSTRUCTIONS ON THAT CHIP.) THE PARTITIONING IS AS FOLLOWS:

CONTROL CHIP DC303-004:		
L2DR	07602R	LOAD 2 DESCRIPTORS
L3DR	07606R	LOAD 3 DESCRIPTORS
MOV	076030	MOVE CHARACTER
MOVCR	076031	MOVE REVERSE JUSTIFIED CHARACTER
MOVTC	076032	MOVE TRANSLATED CHARACTER
LOCC	076040	LOCATE CHARACTER
SKPC	076041	SKIP CHARACTER
SCANC	076042	SCAN CHARACTER
SPANC	076043	SPAN CHARACTER
CMPC	076044	COMPARE CHARACTER
MATC	076045	MATCH CHARACTER
CONTROL CHIP DC303-005:		
CVTNL	076053	CONVERT NUMERIC TO LONG
CVTPN	076054	CONVERT PACKED TO NUMERIC
CVTNP	076055	CONVERT NUMERIC TO PACKED
CVTLN	076057	CONVERT LONG TO NUMERIC
CVTPL	076073	CONVERT PACKED TO LONG
CVTLP	076077	CONVERT LONG TO PACKED
CONTROL CHIP DC303-006:		
ADDN	076050	ADD NUMERIC
SUBN	076051	SUBTRACT NUMERIC
CMPN	076052	COMPARE NUMERIC
ASHN	076056	ARITHMETIC SHIFT NUMERIC
CONTROL CHIP DC303-007:		
ADDP	076070	ADD PACKED
SUBP	076071	SUBTRACT PACKED
CMPP	076072	COMPARE PACKED
ASHP	076076	ARITHMETIC SHIFT PACKED
CONTROL CHIP DC303-008:		
MULP	076074	MULTIPLY PACKED
CONTROL CHIP DC303-009:		
DIVP	076075	DIVIDE PACKED

ALL OF THE ABOVE INSTRUCTIONS EXCEPT FOR L2DR AND L3DR HAVE AN IN-LINE VERSION, TOO. THE MNEMONIC IS THE SAME EXCEPT AN 'I' IS APPENDED TO IT AND THE OPCODE IS THE SAME EXCEPT BIT06 IS SET. FOR EXAMPLE, THE IN-LINE VERSION OF CVTNP (076055) IS CVTNPI (076155).

382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437

6.0 PROGRAM DESCRIPTION

THIS PROGRAM WAS WRITTEN IN MODULES. THE FIRST MODULE INCLUDES START UP AND INITIALIZATION ROUTINES, BASIC DEFINITIONS, VARIABLE DEFINITIONS, THE TRAP CATCHER, AND ACT AND APT INFORMATION. THE SECOND MODULE CONTAINS THE TESTS FOR THE CONTROL CHIP DC303-004 INSTRUCTIONS, THE TEST DATA FOR THOSE INSTRUCTIONS, AN ILLEGAL OPCODE TEST, AND AN INTERRUPTABILITY TEST. THE THIRD MODULE CONTAINS THE TESTS AND TEST DATA FOR CONTROL CHIP DC303-005 INSTRUCTIONS, A SUBROUTINE TO CONVERT LONG INTEGER TO AN INTERMEDIATE FORM, AND A SUBROUTINE TO CONVERT FROM THE INTERMEDIATE FORM TO LONG INTEGER. THE FOURTH MODULE CONTAINS THE TESTS FOR CONTROL CHIP DC303-006 INSTRUCTIONS. THE TESTS FOR CONTROL CHIP DC303-007 ARE IN THE FIFTH MODULE. THE SIXTH MODULE HAS THE TESTS AND TEST DATA FOR CONTROL CHIPS DC303-008 AND DC303-009. FINALLY, THE SEVENTH MODULE CONTAINS END OF PASS, SCOPE, ALL THE OTHER COMMON SUBROUTINES, ASCII MESSAGES, BUFFERS, TABLES, AND TEST DATA FOR CONTROL CHIPS DC303-006 AND DC303-007. WITH BITS 6 AND 7 OF THE SWR EQUAL TO 0 THE DIAGNOSTIC WILL EXECUTE EVERY INSTRUCTION WITH MEMORY MANAGEMENT ENABLED AND DISABLED.

6.1 PROGRAM EXECUTION CHARACTERISTICS

6.2 SUBTEST SUMMARIES

THERE IS ONE SUBTEST FOR EACH CIS INSTRUCTION. FIRST, THE REGISTER FORM OF THE INSTRUCTION IS TESTED WITH SEVERAL SETS OF DATA. THEN ONE CASE OF THE IN-LINE VERSION IS TESTED. THE TEST DATA FOR THE REGISTER FORM IS IN TABLES. THE FIRST WORD OF EACH TABLE IS THE NUMBER OF TEST CASES CONTAINED IN THE TABLE. THE FORMAT OF THE TEST DATA IS DIFFERENT FOR DIFFERENT INSTRUCTIONS; THE FIRST TEST CASE IN EACH TABLE HAS COMMENTS IDENTIFYING EACH ELEMENT IN THE TEST CASE.

THE FOLLOWING IS A SUMMARY OF EACH SUBTEST.

MOVC - THE TABLE OF TEST CASES CONTAINS:

<SOURCE LENGTH>
<SOURCE ADDRESS>
<DESTINATION LENGTH>
<DESTINATION ADDRESS>
<FILL CHARACTER>
<SOURCE BUFFER STATE>
<DESTINATION BUFFER STATE>

THE SOURCE AND DESTINATION BUFFER STATES DESIGNATE WHAT THOSE BUFFERS WILL BE FILLED WITH BEFORE EXECUTION OF THE MOVC INSTRUCTION. A '0' INDICATES EACH BYTE WILL CONTAIN 000, A '-1' INDICATES 377, AND A '1' INDICATES 252 (ALTERNATING 1'S AND 0'S BIT PATTERN). AN EXPECTED DESTINATION RESULT IS COMPUTED, AS WELL AS EXPECTED CONDITION CODES AND VALUES OF R0-R5. THEN THE MOVC INSTRUCTION IS EXECUTED AND THE ACTUAL RESULTS ARE COMPARED WITH THE EXPECTED RESULTS. DISAGREEMENTS ARE

438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491

FLAGGED WITH AN ERROR CALL, WHICH PRINTS OUT 'CIS INSTRUCTION FAILURE', THE PC OF THE ERROR CALL AND THE CONTROL CHIP NUMBER (IN THIS CASE, CONTROL CHIP NUMBER DC303-004). THIS WHOLE PROCEDURE IS REPEATED FOR EACH TEST CASE IN THE TABLE. THEN ONE SIMPLE IN-LINE CASE IS TESTED. THE DATA FOR THE IN-LINE CASE FOLLOWS THE TABLE OF TEST CASES FOR THE REGISTER FORM OF MOVCL, AND IT CONTAINS ONLY: SOURCE LENGTH, SOURCE ADDRESS, DESTINATION LENGTH, AND DESTINATION ADDRESS.

MOVRC - SAME AS MOVCL

MOVTC - SAME AS MOVCL, EXCEPT THAT A TRANSLATION TABLE, LOCATED AT 'TRANS', IS USED. PRIOR TO THE EXECUTION OF THE MOVTC INSTRUCTION, THE TRANSLATION TABLE IS CLEARED. THEN BYTE LOCATION TRANS+0 GETS A 63, TRANS+252 GETS A 314, AND TRANS+377 GETS A 125 (ALL NUMBERS OCTAL).

LOCC - THE TABLE OF TEST CASES CONTAINS:
<SOURCE LENGTH>
<CHARACTER SEARCHED FOR>
<OFFSET>

THE SEARCH BUFFER (BUFF1) IS FILLED WITH ZEROES, THEN AT BYTE LOCATION 'BUFF1+OFFSET' THE 'CHARACTER SEARCHED FOR' IS INSERTED. EXPECTED RESULTS FOR THE CONDITION CODES AND R0-R5 ARE COMPUTED. THE LOCC INSTRUCTION IS EXECUTED AND EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS. DISAGREEMENTS ARE FLAGGED WITH ERROR CALLS.

SKPC - SIMILAR TO LOCC, EXCEPT THAT THE TEST CASES HAVE A 'NOT SEARCHED FOR CHARACTER' IN PLACE OF A 'CHARACTER SEARCHED FOR'. THE SEARCH BUFFER (BUFF1) IS FILLED WITH SEARCHED-FOR CHARACTERS, THEN AT BYTE LOCATION 'BUFF1+OFFSET' THE 'NOT SEARCHED FOR' CHARACTER IS INSERTED.

SCANC - THE TABLE OF TEST CASES CONTAINS:
<SOURCE LENGTH>
<OFFSET>

BUFF1 IS FILLED WITH ZEROES, THEN AT BYTE LOCATION 'BUFF1+OFFSET' A 125 IS INSERTED. THE TABLE 'CHRSET' IS CLEARED, THEN A 1 IS INSERTED AT BYTE LOCATION 'CHRSET+125', TO DENOTE MEMBERSHIP IN THE CHARACTER SET. EXPECTED CONDITION CODES AND R0-R5 VALUES ARE COMPUTED, THE SCANC INSTRUCTION IS EXECUTED, AND DISAGREEMENTS BETWEEN ACTUAL AND EXPECTED RESULTS CAUSE ERROR CALLS.

492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542

SPANC - THE TABLE OF TEST CASES CONTAINS:

<SOURCE LENGTH>
<OFFSET>

BUFF1 IS FILLED WITH ZEROES, THEN A 252 IS INSERTED AT BYTE LOCATION 'BUFF1+OFFSET'. CHRSET IS FILLED WITH ZEROES, THEN BYTE LOCATION 'CHRSET+0' GETS A 40 SO THAT A 0 IN THE SOURCE BUFFER WILL BE CONSIDERED A MEMBER OF THE CHARACTER SET AND A 252 WILL NOT. EXPECTED CONDITION CODES AND R0-R5 VALUES ARE COMPUTED, SPANC IS EXECUTED, EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS, AND DISAGREEMENTS CAUSE ERROR CALLS.

CMPC - THE TABLE OF TEST CASES CONTAINS:

<SOURCE1 LENGTH>
<SOURCE2 LENGTH>

BOTH SOURCE STRINGS ARE FILLED WITH 377'S AND THE FILL CHARACTER IS SET EQUAL TO 377. THEN IN THE LONGER STRING, AT 'SRC.ADR+OFFSET', A BYTE IS CLEARED. THIS BYTE WILL BE THE INEQUALITY BETWEEN THE TWO STRINGS WHICH WILL TERMINATE THE CMPC COMPARISON. ACTUAL RESULTS WILL BE COMPARED WITH PREVIOUSLY COMPUTED EXPECTED RESULTS (CONDITION CODES AND R0-R5 VALUES).

MATC - THE TABLE OF TEST CASES CONTAINS:

<OBJECT LENGTH>
<OFFSET>

THE SOURCE STRING IS THE FIRST 70 BYTES OF BUFF1. BUFF1 IS FIRST FILLED WITH 377'S AND THEN CLEARED FROM 'BUFF1+OFFSET' TO THE END OF BUFF1. THE OBJECT STRING STARTS AT BUFF2, IS OF LENGTH 'OBJECT LENGTH' FROM THE TABLE OF TEST CASES, AND IS ALL ZEROES. EXPECTED VALUES OF CONDITION CODES AND R0-R5 ARE COMPUTED, MATC IS EXECUTED, EXPECTED RESULTS AND ACTUAL RESULTS ARE COMPARED, AND ERROR CALLS ARE MADE FOR DISAGREEMENTS.

CVTLP - THE TABLE OF TEST CASES CONTAINS:

<DESTINATION DATA TYPE AND LENGTH>
<LONG INTEGER HIGH, AND SIGN>
<LONG INTEGER LOW>

THE LONG INTEGER IN THE TEST CASE IS CONVERTED TO PACKED AND PLACED INTO BUFFER 'DS.EXP' (DECIMAL STRING EXPECTED) ACCORDING TO THE DESTINATION TYPE AND LENGTH SPECIFIED IN THE TABLE. THE EXPECTED VALUES OF THE CONDITION CODES AND R0-R5 ARE ALSO COMPUTED. THE CVTLP INSTRUCTION IS EXECUTED AND EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS. DISAGREEMENTS WILL CAUSE ERROR CALLS.

543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594

CVTLN - SAME AS THE CVTL² TEST,
EXCEPT THE LONG INTEGER IS CONVERTED TO A NUMERIC DECIMAL
FORM, ACCORDING TO THE SPECIFICATION IN THE TEST CASE.

CVTPL - THE TABLE OF TEST CASES CONTAINS:
<SOURCE DATA TYPE AND LENGTH>
<POINTER TO PACKED DECIMAL STRING>
<A PACKED DECIMAL STRING>
THE PACKED DECIMAL STRING IS CONVERTED TO A LONG INTEGER
AND THE RESULT IS PUT INTO 'R2.EXP' AND 'R3.EXP'
(EXPECTED RESULTS FOR R2 AND R3). THE EXPECTED RESULTS
FOR THE OTHER GENERAL PURPOSE REGISTER AND THE CONDITION
CODES ARE ALSO COMPUTED. THE CVTPL INSTRUCTION IS
EXECUTED, ACTUAL RESULTS ARE COMPARED WITH EXPECTED
RESULTS, AND DISAGREEMENTS ARE FLAGGED BY ERROR CALLS.

CVTNL - SAME AS THE CVTNL TEST,
EXCEPT THAT NUMERIC DECIMAL STRINGS ARE CONVERTED INSTEAD
OF PACKED DECIMAL STRINGS. ALSO, A TABLE FOR
INTERPRETING ACCEPTABLE SIGN/DIGIT COMBINATIONS FOR
OVERPUNCH IS INCLUDED.

CVTPN - THE TABLE OF TEST CASES CONTAINS:
<SOURCE DATA TYPE AND LENGTH>
<POINTER TO A PACKED SOURCE STRING>
<DESTINATION LENGTH>
EACH PACKED DECIMAL STRING IS CONVERTED INTO EACH OF THE
SIX NUMERIC DECIMAL DATA TYPES. THUS, EACH TEST CASE IS
USED SIX TIMES BEFORE GOING ON TO THE NEXT ONE. EACH
TIME, THE EXPECTED CONDITION CODES AND R0-R5 VALUES ARE
ALSO COMPUTED. THE CVTPN INSTRUCTION IS EXECUTED,
EXPECTED RESULTS ARE COMPARED WITH ACTUAL RESULTS, AND
DISAGREEMENTS ARE FLAGGED BY ERROR CALLS.

CVTNP - THE TABLE OF TEST CASES CONTAINS:
<SOURCE DATA TYPE AND LENGTH>
<POINTER TO A SOURCE STRING>
<DESTINATION DATA TYPE AND LENGTH>
<A SOURCE STRING>
THE SOURCE STRING IS IN AN INTERMEDIATE FORMAT. THE
FIRST WORD INDICATES THE SIGN (0 FOR POSITIVE, -1 FOR
NEGATIVE). FOLLOWING THAT IS THE MAGNITUDE, ONE
BINARY-CODED-DECIMAL DIGIT PER BYTE. THE SOURCE IS
CONVERTED TO A PACKED DECIMAL STRING AND PUT INTO
'DS.EXP', AND THE EXPECTED VALUES OF THE CONDITION CODES
AND R0-R5 ARE COMPUTED. THE CVTNP INSTRUCTION IS
EXECUTED, ACTUAL RESULTS COMPARED WITH EXPECTED RESULTS,
AND DISAGREEMENTS FLAGGED BY ERROR CALLS.

595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649

ADDN - THE TABLE OF TEST CASES CONTAINS:

- <SOURCE1 LENGTH>
- <SOURCE1 POINTER>
- <SOURCE2 LENGTH>
- <SOURCE2 POINTER>
- <DESTINATION LENGTH>
- <DESTINATION POINTER (SUM)>
- <EXPECTED CONDITION CODES>

THE SOURCE1, SOURCE2, AND DESTINATION STRINGS ARE IN AN INTERMEDIATE FORMAT. EACH ONE IS TESTED IN FIVE OF THE SIX DATA TYPE FORMATS (UNSIGNED ZONED IS TESTED IN THE INLINE CASE ONLY). THIS GIVES A TOTAL OF 125 TESTS PERFORMED ON EACH TEST CASE, IN ORDER TO TEST ALL COMBINATIONS OF THE FIVE DATA TYPES. SOURCE1 AND SOURCE2 ARE ADDED TOGETHER BY THE ADDN INSTRUCTION AND THE ANSWER IS COMPARED TO THE EXPECTED ANSWER WHICH IS SUPPLIED IN THE TEST TABLE (POINTED TO BY THE DESTINATION POINTER). EXPECTED CONDITION CODES ARE ALSO SUPPLIED IN THE TEST CASE TABLE AND IS COMPARED WITH THE ACTUAL CONDITION CODES. THE EXPECTED GENERAL PURPOSE REGISTER VALUES (R0-R5) ARE THE ONLY EXPECTED RESULTS THAT ARE ACTUALLY CALCULATED BY THE DIAGNOSTIC, AND THESE TOO ARE COMPARED WITH THE ACTUAL RESULTS. ANY DISAGREEMENTS BETWEEN ACTUAL AND EXPECTED RESULTS ARE FLAGGED BY AN ERROR CALL.

SUBN - THE SAME AS ADDN,

EXCEPT THAT THE TEST CASES HAVE THE EXPECTED DESTINATION STRING (IN AN INTERMEDIATE FORMAT) AND EXPECTED CONDITION CODES FOR A SUBTRACT OPERATION INSTEAD OF AN ADD OPERATION. THE SOURCE1 LENGTH AND POINTER, THE SOURCE2 LENGTH AND POINTER, AND THE DESTINATION LENGTH ARE IDENTICAL TO THOSE USED IN THE ADDN INSTRUCTION. THE TEST OPERATES IN EXACTLY THE SAME MANNER AS THE ADDN TEST.

CMPN - THE TABLE OF TEST CASES CONTAINS:

- <SOURCE1 LENGTH>
- <SOURCE1 POINTER>
- <SOURCE2 LENGTH>
- <SOURCE2 POINTER>

THE TWO SOURCE STRINGS ARE IN AN INTERMEDIATE FORMAT. EACH ONE IS TESTED IN FIVE OF THE SIX NUMERIC DATA TYPE FORMATS (UNSIGNED ZONED IS TESTED IN THE INLINE CASE ONLY). THIS GIVES A TOTAL OF 25 TESTS PERFORMED ON EACH TEST CASE. IN EACH TEST, SOURCE1 IS ARITHMETICALLY COMPARED WITH SOURCE2 AND THE CONDITIONS ARE UPDATED TO REFLECT THE RESULTS. THE EXPECTED CONDITION CODES ARE COMPARED WITH THE ACTUAL CONDITION CODES, AND R0-R5 ARE TESTED TO SEE THAT THEY ARE CLEAR. ANY DISCREPANCIES ARE FLAGGED BY ERROR CALLS.

650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690

ASHN - THE TABLE OF TEST CASES CONTAINS:

<SOURCE LENGTH>
<DESTINATION LENGTH>
<SHIFT COUNT>
<ROUNDING DIGIT>

THE SOURCE STRING IS ALWAYS A STRING OF '4''S STARTING AT
DSBUF1 AND OF THE LENGTH SPECIFIED BY THE TEST
CASE. EACH TIME, THE EXPECTED RESULTS ARE CALCULATED AND
PUT INTO DS.EXP AND CC.EXP. ACTUAL RESULTS ARE COMPARED
WITH EXPECTED RESULTS AND DISAGREEMENTS ARE FLAGGED BY
ERROR CALLS.

ADDP - SAME AS THE ADDN TEST,
EXCEPT THAT EACH TEST CASE IS RUN ONCE, USING SIGNED
PACKED FORMAT. UNSIGNED PACKED IS TESTED IN THE IN-LINE
TEST. THE IDENTICAL TABLE OF TEST CASES USED FOR ADDN IS
USED HERE.

SUBP - SAME AS THE SUBN TEST,
EXCEPT THAT EACH TEST CASE IS RUN ONCE, USING SIGNED
PACKED FORMAT. UNSIGNED PACKED IS TESTED IN THE IN-LINE
TEST. THE IDENTICAL TABLE OF TEST CASES USED FOR SUBN IS
USED HERE.

CMPP - SAME AS THE CMPN TEST,
EXCEPT THAT EACH TEST CASE IS RUN ONCE, USING SIGNED
PACKED FORMAT. UNSIGNED PACKED IS TESTED IN THE IN-LINE
TEST. THE IDENTICAL TABLE OF TEST CASES USED FOR CMPN IS
USED HERE.

ASHP - SAME AS THE ASHN TEST,
EXCEPT THAT EACH TEST CASE IS RUN ONCE, USING SIGNED
PACKED FORMAT. UNSIGNED PACKED IS TESTED IN THE IN-LINE
TEST. THE IDENTICAL TABLE OF TEST CASES USED FOR ASHN IS
USED HERE.

691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719

MULP - THE TABLE OF TEST CASES CONTAINS:

<SOURCE1 LENGTH>
<SOURCE1 POINTER>
<SOURCE2 LENGTH>
<SOURCE2 POINTER>
<DESTINATION LENGTH>
<DESTINATION POINTER (PRODUCT)>
<CC.EXP (PRODUCT)>

THE TWO SOURCES (MULTIPLIER AND MULTIPLICAND) AND EXPECTED DESTINATION (PRODUCT) ARE IN THE TABLE OF TEST CASES, IN INTERMEDIATE FORMAT. THE TWO SOURCES ARE MULTIPLIED TOGETHER (IN SIGNED PACKED FORMAT) AND THE EXPECTED PRODUCT IS COMPARED WITH THE ACTUAL RESULT. ALSO, THE EXPECTED VALUES OF R0-R5 AND THE CONDITION CODES ARE COMPARED WITH THE ACTUAL RESULTS. ANY DISCREPANCIES ARE FLAGGED BY ERROR CALLS. UNSIGNED PACKED FORMAT IS TESTED IN THE INLINE CASE ONLY.

DIVP - SAME AS THE MULP TEST, EXCEPT THAT A DIVP IS EXECUTED AND THE TABLE OF TEST CASES CONTAINS QUOTIENT RESULTS FOR THE LAST TWO TABLE ENTRIES (DESTINATION POINTER AND CC.EXP).

7.0 LISTING
%

```

720      ;*COPYRIGHT (C) 1981
721      ;*DIGITAL EQUIPMENT CORP.
722      ;*MAYNARD, MASS. 01754
723      ;*
724      ;*
725      ;*THIS PROGRAM WAS ASSEMBLED USING SPMACJ
726      ;*
727      .SBTTL  BASIC DEFINITIONS
728
729      ;*INITIAL ADDRESS OF THE STACK POINTER *** 776 ***
730      000776  STACK= 776
731      .EQUIV  EMT,ERROR      ;;BASIC DEFINITION OF ERROR CALL
732      .EQUIV  IOT,SCOPE     ;;BASIC DEFINITION OF SCOPE CALL
733
734      ;*MISCELLANEOUS DEFINITIONS
735      000011  HT= 11          ;;CODE FOR HORIZONTAL TAB
736      000012  LF= 12          ;;CODE FOR LINE FEED
737      000015  CR= 15          ;;CODE FOR CARRIAGE RETURN
738      000200  CRLF= 200       ;;CODE FOR CARRIAGE RETURN-LINE FEED
739      177776  PS= 177776     ;;PROCESSOR STATUS WORD
740      .EQUIV  PS,PSW
741      177774  STKLMT= 177774  ;;STACK LIMIT REGISTER
742      177546  LKS= 177546    ;;LINE CLOCK STATUS REGISTER
743      000100  BUFLN= 100     ;;LENGTH OF BUFF1 AND BUFF2
744
745      ;*CONSOLE REGISTER DEFINITIONS
746      177560  TKS= 177560
747      177562  TKB= 177562
748      177564  TFS= 177564
749      177566  TPB= 177566
750
751      ;*MEMORY MANAGEMENT REGISTERS
752      172300  KIPDR0= 172300
753      172302  KIPDR1= 172302
754      172304  KIPDR2= 172304
755      172306  KIPDR3= 172306
756      172310  KIPDR4= 172310
757      172312  KIPDR5= 172312
758      172314  KIPDR6= 172314
759      172316  KIPDR7= 172316
760
761      172340  KIPAR0= 172340
762      172342  KIPAR1= 172342
763      172344  KIPAR2= 172344
764      172346  KIPAR3= 172346
765      172350  KIPAR4= 172350
766      172352  KIPAR5= 172352
767      172354  KIPAR6= 172354
768      172356  KIPAR7= 172356
769
770      177572  SR1= 177572
771
772      ;*GENERAL PURPOSE REGISTER DEFINITIONS
773      000000  R0= %0          ;;GENERAL REGISTER
774      000001  R1= %1          ;;GENERAL REGISTER
775      000002  R2= %2          ;;GENERAL REGISTER
    
```

```

776      000003      R3=      %3          ;;GENERAL REGISTER
777      000004      R4=      %4          ;;GENERAL REGISTER
778      000005      R5=      %5          ;;GENERAL REGISTER
779      000006      R6=      %6          ;;GENERAL REGISTER
780      000007      R7=      %7          ;;GENERAL REGISTER
781      000006      SP=      %6          ;;STACK POINTER
782      000007      PC=      %7          ;;PROGRAM COUNTER
783
784      ;*PRIORITY LEVEL DEFINITIONS
785      000000      PR0=      0          ;;PRIORITY LEVEL 0
786      000040      PR1=      40         ;;PRIORITY LEVEL 1
787      000100      PR2=      100        ;;PRIORITY LEVEL 2
788      000140      PR3=      140        ;;PRIORITY LEVEL 3
789      000200      PR4=      200        ;;PRIORITY LEVEL 4
790      000240      PR5=      240        ;;PRIORITY LEVEL 5
791      000300      PR6=      300        ;;PRIORITY LEVEL 6
792      000340      PR7=      340        ;;PRIORITY LEVEL 7
793
794      ;*CONDITION CODE EQLATES
795      000010      NBIT=      10
796      000004      ZBIT=      4
797      000002      VBIT=      2
798      000001      CBIT=      1
799
800      ;*'SWITCH REGISTER' SWITCH DEFINITIONS
801      100000      SW15=     100000
802      040000      SW14=     40000
803      020000      SW13=     20000
804      010000      SW12=     10000
805      004000      SW11=     4000
806      002000      SW10=     2000
807      001000      SW09=     1000
808      000400      SW08=     400
809      000200      SW07=     200
810      000100      SW06=     100
811      000040      SW05=     40
812      000020      SW04=     20
813      000010      SW03=     10
814      000004      SW02=     4
815      000002      SW01=     2
816      000001      SW00=     1
817      .EQUIV      SW09,SW9
818      .EQUIV      SW08,SW8
819      .EQUIV      SW07,SW7
820      .EQUIV      SW06,SW6
821      .EQUIV      SW05,SW5
822      .EQUIV      SW04,SW4
823      .EQUIV      SW03,SW3
824      .EQUIV      SW02,SW2
825      .EQUIV      SW01,SW1
826      .EQUIV      SW00,SW0
827
828      ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
829      100000      BIT15=    100000
830      040000      BIT14=    40000
831      020000      BIT13=    20000
    
```

```

832      010000      BIT12= 10000
833      004000      BIT11= 4000
834      002000      BIT10= 2000
835      001000      BIT09= 1000
836      000400      BIT08= 400
837      000200      BIT07= 200
838      000100      BIT06= 100
839      000040      BIT05= 40
840      000020      BIT04= 20
841      000010      BIT03= 10
842      000004      BIT02= 4
843      000002      BIT01= 2
844      000001      BIT00= 1
845      .EQUIV      BIT09,BIT9
846      .EQUIV      BIT08,BIT8
847      .EQUIV      BIT07,BIT7
848      .EQUIV      BIT06,BIT6
849      .EQUIV      BIT05,BIT5
850      .EQUIV      BIT04,BIT4
851      .EQUIV      BIT03,BIT3
852      .EQUIV      BIT02,BIT2
853      .EQUIV      BIT01,BIT1
854      .EQUIV      BIT00,BIT0
  
```

```

;*BASIC 'CPU' TRAP VECTOR ADDRESSES
ERRVEC= 4          ;; TIME OUT AND OTHER ERRORS
RESVEC= 10         ;; RESERVED AND ILLEGAL INSTRUCTIONS
TRITVEC=14        ;; 'T' BIT
TRTVEC= 14        ;; TRACE TRAP
BPTVEC= 14        ;; BREAKPOINT TRAP (BPT)
IOTVEC= 20        ;; INPUT/OUTPUT TRAP (IOT) **SCOPE**
PWRVEC= 24        ;; POWER FAIL
EMTVEC= 30        ;; EMULATOR TRAP (EMT)
TRAPVEC=34        ;; 'TRAP' TRAP
TKVEC= 60         ;; TTY KEYBOARD VECTOR
TPVEC= 64         ;; TTY PRINTER VECTOR
LTCVEC= 100       ;; LTC VECTOR
MMVEC= 250        ;; MEMORY MANAGEMENT VECTOR
ASWREG =0         ;; INITIAL APT SOFTWARE SWR VALUE
  
```

```

;*EQUATES FOR ROUTINE CALLS THAT USE TRAP INSTRUCTION
874      104401      TYPE= 104401      ;; TTY TYPEOUT ROUTINE
875      104402      TYPOC= 104402     ;; TYPE OCTAL NUMBER (WITH LEADING ZEROES) ROUTINE
876      104403      TYPOS= 104403     ;; TYPE OCTAL NUMBER (NO LEADING ZEROES) ROUTINE
877      104404      TYPON= 104404     ;; TYPE OCTAL NUMBER (AS PER LAST CALL) ROUTINE
878      104405      GTSWR= 104405     ;; GET SOFT-SWR SETTING ROUTINE
879      104406      CKSWR= 104406     ;; TEST FOR CHANGE IN SOFT-SWR ROUTINE
880      100407      RDCHR= 100407     ;; TTY TYPEIN CHARACTER ROUTINE
881      104410      RDLIN= 104410     ;; TTY TYPEIN STRING ROUTINE
  
```

```

*****
*** COMMERCIAL INSTRUCTION SET OPCODE EQUATES ***
MFPT      =000007      ;;MOVE FROM PROCESSOR TYPE
  
```

```

885      000007
886
887
  
```

888		:	COMMERCIAL LOAD DESCRIPTORS	
889	076020	L2D0	=76020	:LOAD 2 DESCRIPTORS @ (R0)+
890	076021	L2D1	=76021	:LOAD 2 DESCRIPTORS @ (R1)+
891	076022	L2D2	=76022	:LOAD 2 DESCRIPTORS @ (R2)+
892	076023	L2D3	=76023	:LOAD 2 DESCRIPTORS @ (R3)+
893	076024	L2D4	=76024	:LOAD 2 DESCRIPTORS @ (R4)+
894	076025	L2D5	=76025	:LOAD 2 DESCRIPTORS @ (R5)+
895	076026	L2D6	=76026	:LOAD 2 DESCRIPTORS @ (R6)+
896	076027	L2D7	=76027	:LOAD 2 DESCRIPTORS @ (R7)+
897				
898	076060	L3D0	=76060	:LOAD 3 DESCRIPTORS @ (R0)+
899	076061	L3D1	=76061	:LOAD 3 DESCRIPTORS @ (R1)+
900	076062	L3D2	=76062	:LOAD 3 DESCRIPTORS @ (R2)+
901	076063	L3D3	=76063	:LOAD 3 DESCRIPTORS @ (R3)+
902	076064	L3D4	=76064	:LOAD 3 DESCRIPTORS @ (R4)+
903	076065	L3D5	=76065	:LOAD 3 DESCRIPTORS @ (R5)+
904	076066	L3D6	=76066	:LOAD 3 DESCRIPTORS @ (R6)+
905	076067	L3D7	=76067	:LOAD 3 DESCRIPTORS @ (R7)+
906				
907				
908		:	CHARACTER STRING INSTRUCTIONS	
909	076030	MOVC	=76030	:MOVE CHARACTER
910	076031	MOVRC	=76031	:MOVE REVERSE CHARACTER
911	076032	MOVTC	=76032	:MOVE TRANSLATED CHARACTER
912	076040	LOCC	=76040	:LOCATE CHARACTER
913	076041	SKPC	=76041	:SKIP CHARACTER
914	076042	SCANC	=76042	:SCAN CHARACTER
915	076043	SPANC	=76043	:SPAN CHARACTER
916	076044	CMPC	=76044	:COMPARE CHARACTER
917	076045	MATC	=76045	:MATCH CHARACTER
918				
919	076130	MOVCI	=76130	:IN-LINE VERSION OF MOVE CHARACTER
920	076131	MOVRCI	=76131	:IN-LINE VERSION
921	076132	MOVTCI	=76132	:
922	076140	LOCCI	=76140	::
923	076141	SKPCI	=76141	::
924	076142	SCANCI	=76142	::
925	076143	SPANCI	=76143	::
926	076144	CMPCI	=76144	::
927	076145	MATCI	=76145	::
928				
929				
930		:	NUMERIC STRING INSTRUCTIONS	
931	076050	ADDN	=76050	:ADD NUMERIC
932	076051	SUBN	=76051	:SUBTRACT NUMERIC
933	076052	CMPN	=76052	:COMPARE NUMERIC
934	076053	CVTNL	=76053	:CONVERT NUMERIC TO LONG
935	076054	CVTPN	=76054	:CONVERT PACKED TO NUMERIC
936	076055	CVTNP	=76055	:CONVERT NUMERIC TO PACKED
937	076056	ASHN	=76056	:ARITHMETIC SHIFT NUMERIC
938	076057	CVTLN	=76057	:CONVERT LONG TO NUMERIC
939				
940	076150	ADDNI	=76150	:IN-LINE VERSION OF ADD NUMERIC
941	076151	SUBNI	=76151	:IN-LINE VERSION
942	076152	CMJNI	=76152	:
943	076153	CVTNLI	=76153	::


```

944      076154      CVTPNI  =76154      :      ..
945      076155      CVTNPI  =76155      :      ..
946      076156      ASHNI   =76156      :      ..
947      076157      CVTLNI  =76157      :      ..
948
949
950      :          PACKED STRING INSTRUCTIONS
951      076070      ADDP    =76070      ;ADD PACKED
952      076071      SUBP    =76071      ;SUBTRACT PACKED
953      076072      CMPP    =76072      ;COMPARE PACKED
954      076073      CVTPL   =76073      ;CONVERT PACKED TO LONG
955      076074      MULP    =76074      ;MULTIPLY PACKED
956      076075      DIVP    =76075      ;DIVIDE PACKED
957      076076      ASHP    =76076      ;ARITHMETIC SHIFT PACKED
958      076077      CVTLP   =76077      ;CONVERT LONG TO PACKED
959
960      076170      ADDPI    =76170      ;IN-LINE VERSION OF ADD PACKED
961      076171      SUBPI    =76171      ;IN-LINE VERSION
962      076172      CMPPI    =76172      :      ..
963      076173      CVTPLI   =76173      :      ..
964      076174      MULPI    =76174      :      ..
965      076175      DIVPI    =76175      :      ..
966      076176      ASHPI    =76176      :      ..
967      076177      CVTLPI   =76177      :      ..
968
969
970      :          CIS CLASS ILLEGAL OPCODE
971      076033      ILLOP    =76033      ;ILLEGAL OPCODE
972
973
974      :*****
975
976
977
978
979      .SBTTL  TRAP CATCHER
980
981      .=0
982      ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A '+2,HALT'
983      ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
984      ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
985      .=174
986      000174      000000      DISPREG: .WORD 0      ;;SOFTWARE DISPLAY REGISTER
987      000176      000000      SWREG:   .WORD 0      ;;SOFTWARE SWITCH REGISTER
988
989
990      .-100      ;HANDLE EVENT LINE INTERRUPT
991      000100      000102      .WORD 102
992      000102      000002      .WORD 2
993
994      .=200
995      000200      000137      001200      JMP      START      ;STARTING LOCATION FOR PROGRAM
996
997      .=300
998      .SBTTL  ACT11 HOOKS
999
    
```

```
1000 ::*****
1001 :HOOKS REQUIRED BY ACT11
1002 $SVPC=. ;SAVE PC
1003 .=46
1004 000046 056432 $ENDAD ;;1)SET LOC.46 TO ADDRESS OF $ENDAD IN .$EOP
1005 000052 000052 .=52
1006 000052 000000 .WORD 0 ;;2)SET LOC.52 TO ZERO
1007 000300 000300 .=$SVPC ;; RESTORE PC
1008 .SBTTL APT PARAMETER BLOCK
1009
1010 ::*****
1011 :SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
1012 :*****
1013 .SX=. ;;SAVE CURRENT LOCATION
1014 000024 000024 .=24 ;;SET POWER FAIL TO POINT TO START OF PROGRAM
1015 000024 000200 200 ;;FOR APT START UP
1016 000044 000044 .=44 ;;POINT TO APT INDIRECT ADDRESS PNTR.
1017 000044 000300 $APTHDR ;;POINT TO APT HEADER BLOCK
1018 000300 000300 .=.SX ;;RESET LOCATION COUNTER
1019 :*****
1020 :SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 D'ACNOSTIC
1021 :INTERFACE SPEC.
1022
1023 $APTHD:
1024 000300 000000 $HIBTS: .WORD 0 ;;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
1025 000302 000632 $MBADR: .WORD $MAIL ;;ADDRESS OF APT MAILBOX (BITS 0-15)
1026 000304 000300 $STMT: .WORD 300 ;;RUN TIM OF LONGEST TEST
1027 000306 000300 $PASTM: .WORD 300 ;;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
1028 000310 000000 $UNITM: .WORD ;;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
1029 000312 000015 .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)
1030 .SBTTL COMMON TAGS
1031
1032 :*****
1033 :*THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
1034 :*USED IN THE PROGRAM.
1035
1036 000400 000400 .=400
1037 000400 000000 $CMTAG: ;;START OF COMMON TAGS
1038 000402 000000 .WORD 0
1039 000404 000000 $STNM: .WORD 0 ;;CONTAINS THE TEST NUMBER
1040 000406 000000 $TIMES: .WORD 0 ;;CONTAINS NUMBER OF ITERATIONS DESIRED
1041 000408 000000 $ICNT: .WORD 0 ;;CONTAINS SUBTEST ITERATION COUNT
1042 000410 000000 $LPADR: .WORD 0 ;;CONTAINS SCOPE LOOP ADDRESS
1043 000412 000000 $LFERR: .WORD 0 ;;CONTAINS SCOPE RETURN FOR ERRORS
1044 000414 000000 $ERTTL: .WORD 0 ;;CONTAINS TOTAL ERRORS DETECTED
1045 000416 001 $ERMAX: .BYTE 1 ;;CONTAINS MAX. ERRORS PER TEST
1046 000420 000000 .EVEN
1047 000422 000 $ERRPC: .WORD 0 ;;CONTAINS PC OF LAST ERROR INSTRUCTION
1048 000424 000 $AUTOB: .BYTE 0 ;;AUTOMATIC MODE INDICATOR
1049 000426 000 $INTAG: .BYTE 0 ;;INTERRUPT MODE INDICATOR
1050 000428 000000 .WORD 0
1051 000430 000176 SWR: .WORD SWREG ;;ADDRESS OF SWITCH REGISTER
1052 000432 000174 DISPLAY: .WORD DISPREG ;;ADDRESS OF DISPLAY REGISTER
1053 000434 000 $NULL: .BYTE 0 ;;CONTAINS NULL CHARACTER FOR FILLS
1054 000436 002 $FILLS: .BYTE 2 ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
1055 000438 012 $FILLC: .BYTE 12 ;;INSERT FILL CHARS. AFTER A 'LINE FEED'
```

1056	000435	000		\$TPFLG:	.BYTE	0	:::"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
1057	000436	177607	000377	\$BELL:	.ASCIZ	<207><377><377>	:::CODE FOR BELL
1058	000442	077		\$QUES:	.ASCII	/?/	:::QUESTION MARK
1059	000443	015		\$CRLF:	.ASCII	<15>	:::CARRIAGE RETURN
1060	000444	000012		\$LF:	.ASCIZ	<12>	:::LINE FEED
1061							
1062							
1063	000446	000000		MMCYL:	.WORD	0	:::MEMORY MANAGEMENT CYCLE FLAG: 0=KTOFF, 1=KTON
1064	000450	000000		SIGN0:	.WORD	0	:::SIGN FLAG: 0 FOR +, -1 FOR -
1065	000452	000000		SIGN1:	.WORD	0	:::
1066	000454	000000		SIGN2:	.WORD	0	:::
1067	000456	000000		ERRFLG:	.WORD	0	:::ERROR FLAG
1068	000460	000000		INTFLG:	.WORD	0	:::INTERRUPT FLAG
1069	000462	000000		INTPAS:	.WORD	0	:::FLAGS FIRST PASS OF INTERRUPT TEST
1070	000464	000000		DATTYP:	.WORD	0	:::DECIMAL STRING DATA TYPE FLAG
1071	000466	000000		XXDP:	.WORD	0	:::IF NOT 0 THEN WE ARE IN XXDP CHAIN MODE
1072	000470	000000		COUNT:	.WORD	0	:::TEMPORARY COUNTER
1073	000472	000000		DELAY:	.WORD	0	:::TIME-DELAY COUNT STORED HERE
1074	000474	000000		ERRTMP:	.WORD	0	:::
1075	000476	000000		TEMP0:	.WORD	0	:::TEMPORARY VARIABLE
1076	000500	000000		TEMP1:	.WORD	0	:::TEMPORARY VARIABLE
1077	000502	000000		TEMP2:	.WORD	0	:::TEMPORARY VARIABLE
1078	000504	000000		TEMP3:	.WORD	0	:::TEMPORARY VARIABLE
1079	000506	000000		TEMP4:	.WORD	0	:::TEMPORARY VARIABLE
1080	000510	000000		TEMP5:	.WORD	0	:::TEMPORARY VARIABLE
1081	000512	000000		OFFSET:	.WORD	0	:::
1082	000514	000000		DSLEN1:	.WORD	0	:::FIRST DECIMAL STRING LENGTH
1083	000516	000000		DSLEN2:	.WORD	0	:::SECOND DECIMAL STRING LENGTH
1084	000520	000000		CC:	.WORD	0	:::CONDITION CODES MAY BE SAVED HERE
1085	000522	000000		CC.EXP:	.WORD	0	:::EXPECTED CONDITION CODES
1086	000524	000000		R0.EXP:	.WORD	0	:::EXPECTED VALUE OF R0
1087	000526	000000		R1.EXP:	.WORD	0	:::R1
1088	000530	000000		R2.EXP:	.WORD	0	:::R2
1089	000532	000000		R3.EXP:	.WORD	0	:::R3
1090	000534	000000		R4.EXP:	.WORD	0	:::R4
1091	000536	000000		R5.EXP:	.WORD	0	:::R5
1092	000540	000000		BF.EXP:	.WORD	0	:::EXPECTED BUFFER DATA
1093	000542	000000		BFADRS:	.WORD	0	:::BUFFER POINTER
1094	000544	000000		TSTCAS:	.WORD	0	:::POINTER TO CURRENT TEST CASE IN TEST TABLE
1095	000546	000000		CTL:	.WORD	0	:::CONTROL CHIP NUMBER
1096	000550	000000		MSGADR:	.WORD	0	:::ADDRESS OF THE ASCII TO BE PRINTED
1097	000552	000000		ERR:	.WORD	0	:::ERROR TYPE NUMBER
1098	000554	000		DSTMP0:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1099	000555	000		DSTMP1:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1100	000556	000		DSTMP2:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1101	000557	000		DSTMP3:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1102	000560	000		DSTMP4:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1103	000561	000		DSTMP5:	.BYTE	0	:::DECIMAL STRING TEMPORARY
1104	000562	000000		DATA00:	.WORD	0	:::TEST DATA GETS LOADED HERE
1105	000564	000000		DATA01:	.WORD	0	:::
1106	000566	000000		DATA02:	.WORD	0	:::
1107	000570	000000		DATA03:	.WORD	0	:::
1108	000572	000000		DATA04:	.WORD	0	:::
1109	000574	000000		DATA05:	.WORD	0	:::
1110	000576	000000		DATA06:	.WORD	0	:::
1111							

```
1112 000600 000000 SAVR0: .WORD 0 ;GENERAL PURPOSE REGISTERS MAY BE SAVED HERE
1113 000602 000000 SAVR1: .WORD 0 ;
1114 000604 000000 SAVR2: .WORD 0 ;
1115 000606 000000 SAVR3: .WORD 0 ;
1116 000610 000000 SAVR4: .WORD 0 ;
1117 000612 000000 SAVR5: .WORD 0 ;
1118 000614 000000 SAVPSW: .WORD 0 ;PSW MAY BE SAVED HERE
1119 000616 000000 DESCRO: .WORD 0 ;TEMPS FOR DS DESCRIPTORS
1120 000620 000000 DESCR1: .WORD 0
1121 000622 000000 DESCR2: .WORD 0
1122 000624 000000 DESCR3: .WORD 0
1123 000626 000000 DESCR4: .WORD 0
1124 000630 000000 DESCR5: .WORD 0
1125
1126
1127 ;*****
1128 .SBTTL APT MAILBOX-ETABLE
1129
1130 ;*****
1131 ;
1132 ;
1133 ;
1134 .EVEN
1135 000632 $MAIL: ;: APT MAILBOX
1136 000632 000000 $MSGTY: .WORD 000000 ;: MESSAGE TYPE CODE
1137 000634 000000 $FATAL: .WORD 000000 ;: FATAL ERROR NUMBER
1138 000636 000000 $TESTN: .WORD 000000 ;: TEST NUMBER
1139 000640 000000 $PASS: .WORD 000000 ;: PASS COUNT
1140 000642 000000 $DEVCT: .WORD 000000 ;: DEVICE COUNT
1141 000644 000000 $UNIT: .WORD 000000 ;: I/O UNIT NUMBER
1142 000646 000000 $MSGAD: .WORD 000000 ;: MESSAGE ADDRESS
1143 000650 000000 $MSGLG: .WORD 000000 ;: MESSAGE LENGTH
1144 000652 $ETABLE: ;: APT ENVIRONMENT TABLE
1145 000652 000 $ENV: .BYTE 000 ;: ENVIRONMENT BYTE
1146 000653 000 $ENVM: .BYTE 000 ;: ENVIRONMENT MODL BITS
1147 000654 000000 $SWREG: .WORD 000000 ;: APT SWITCH REGISTER
1148 000656 000000 $USWR: .WORD 000000 ;: USER SWITCHES
1149 000660 000000 $CPUOP: .WORD 000000 ;: CPU TYPE, OPTIONS
1150 ;*
1151 ;* ;: BITS 15-11=CPU TYPE
1152 ;* ;: 11/04=01,11/05=02,11/20=03,11/40=04,11/45=05
1153 ;* ;: 11/70=06,PDQ=07,Q=10
1154 ;* ;: BIT 10=REAL TIME CLOCK
1155 ;* ;: BIT 9=FLOATING POINT PROCESSOR
1156 000662 000 $MAMS1: .BYTE 000 ;: HIGH ADDRESS, M.S. BYTE
1157 000663 000 $MTYP1: .BYTE 000 ;: MEM. TYPE, BLK#1
1158 ;* ;: MEM. TYPE BYTE -- (HIGH BYTE)
1159 ;* ;: 900 NSEC CORE=001
1160 ;* ;: 300 NSEC BIPOLAR=002
1161 ;* ;: 500 NSEC MOS=003
1162 000664 $ETEND:
1163 .EVEN
1164
1165
1166
1167 ;*****
```

```

1168 .SBTTL START UP AND INITIALIZATION ROUTINES
1169 ::*****
1170 001200 001200 .=1200
1171 001200 START:
1172
1173 .SBTTL INITIALIZE THE COMMON TAGS
1174 ::CLEAR THE COMMON TAGS (%CMTAG) AREA
1175 001200 012706 000400 MOV #CMTAG,R6 ;;FIRST LOCATION TO BE CLEARED
1176 001204 005026 CLR (R6)+ ;;CLEAR MEMORY LOCATION
1177 001206 022706 000426 CMP #SWR,R6 ;;DONE?
1178 001212 001374 BNE -6 ;;LOOP BACK IF NO
1179 001214 012706 001100 MOV #1100,SP ;;SETUP THE STACK POINTER
1180
1181 001220 012737 056554 000004 ::INITIALIZE A FEW VECTORS
1182 001226 012737 000340 000006 MOV #TIMOUT,@#ERRVEC ;TIME-OUT VECTOR
1183 001234 012737 056570 000010 MOV #340,@#ERRVEC+2 ;LEVEL 7
1184 001242 012737 000340 000012 MOV #ILLRES,@#RESVEC ;ILLEGAL/RESERVED INSTR. VECTOR
1185 001250 012737 057542 000020 MOV #340,@#RESVEC+2 ;LEVEL 7
1186 001256 012737 000340 000022 MOV $$SCOPE,@#IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
1187 001264 012737 062154 000034 MOV #340,@#IOTVEC+2 ;;LEVEL 7
1188 001272 012737 000340 000036 MOV #STRAP,@#TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
1189 001300 012737 057662 000024 MOV #340,@#TRAPVEC+2;LEVEL 7
1190 001306 012737 000340 000026 MOV #PWRDN,@#PWRVEC ;;POWER FAILURE VECTOR
1191 001314 013737 056506 056504 MOV #340,@#PWRVEC+2 ;;LEVEL 7
1192 001322 012737 001322 000410 MOV $ENDCT,$EOPCT ;;SETUP END-OF-PROGRAM COUNTER
1193 001330 012737 000102 000100 MOV #,$LADR ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
1194 001336 012737 000002 000102 MOV #102,@#LTCVEC ;GO TO 102
1195 001344 012737 000252 000250 MOV #2,@#102 ;RTI
1196 001352 012737 177777 000446 MOV #252,@#MMVEC ;SET UP KT VECTOR
1197 ::SET UP SOFTWARE SWITCH REGISTER
1198 001360 012737 000176 000426 65$: MOV #SWREG,SWR ;;POINT TO SOFTWARE SWR
1199 001366 012737 000174 000430 MOV #DISPREG,DISPLAY
1200
1201 001374 005037 000640 CLR $PASS ;;CLEAR PASS COUNT
1202 001400 132737 000200 000653 BITB #APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
1203 001406 001403 BEQ 67$ ;;YES,USE NON-APT SWITCH
1204 001410 012737 000654 000426 MOV #SWREG,SWR ;;NO,USE APT SWITCH REGISTER
1205 001416
1206
1207
1208 ;DETERMINE OPERATING ENVIRONMENT
1209 001416 005737 000042 TST @#42 ;ARE WE IN ACT/XXDP AUTO MODE?
1210 001422 001410 BEQ 1$ ;BRANCH IF NO
1211 001424 023737 000042 000046 CMP @#42,@#46 ;IS IT ACT AUTO MODE?
1212 001432 001410 BEQ 2$ ;BRANCH IF YES
1213 001434 012737 177777 000466 MOV #-1,XXDP ;SET XXDP CHAIN MODE INDICATOR
1214 001442 000404 BR 2$ ;
1215 001444 123727 000652 000001 1$: CMPB $ENV,#1 ;ARE WE IN APT AUTO MODE?
1216 001452 001003 BNE 3$ ;BRANCH IF NO
1217 001454 112737 000001 000422 2$: MOVB #1,$AUTOB ;SET AUTO MODE INDICATOR
1218 ;PRINT TITLE IF NOT IN ACT OR APT AUTO MODE
1219 001462 005227 177777 3$: INC #-1 ;FIRST TIME?
1220 001466 001012 BNE 5$ ;SKIP TITLE IF NOT
1221 001470 005737 000422 TST $AUTOB ;ARE WE IN AUTO MODE?
1222 001474 001403 BEQ 4$ ;BRANCH TO TITLE TYPEOUT IF NOT
1223 001476 005737 000466 TST XXDP ;IS THE AUTO MODE UNDER XXDP?

```

```
1224 001502 001404          BEQ      5$          ;SKIP TITLE IF NOT
1225 001504 012737 062232 000550 4$:    MOV      #TITLE,MSGADR
1226 001512 104401          TYPE          ;PRINT OUT THE TITLE
1227          ;GET THE VALUE IN THE SOFTWARE SWITCH REGISTER
1228 001514 005737 000422 5$:    TST      $AUTOB      ;ARE WE IN AN AUTOMATIC MODE?
1229 001520 001001          BNE      BEGIN        ;BRANCH IF YES
1230 001522 104405          GTSWR         ;ASK FOR SWR INPUT FROM THE CONSOLE TERMINAL
1231
1232          BEGIN:
1233 001524          CALL MMSETUP          ;GO CYCLE MEMORY MANAGEMENT
1234 001524 004737 057166          JSR      PC,MMSETJP
```

1235
 1236
 1237
 1238
 1239
 1240
 1241
 1242
 1243
 1244
 1245
 1246
 1247
 1248
 1249
 1250
 1251
 1252
 1253
 1254
 1255
 1256
 1257
 1258
 1259
 1260 001530
 1261 001530 000004
 1262
 1263 001532
 1264 001532 017737 176670 000476
 1265 001540 042737 177700 000476
 1266 001546
 1267 001546 005737 000476
 1268 001552 001406
 1269 001554
 1270 001554 032737 000001 000476
 1271 001562 001002
 1272 001564
 1273 001564 000137 016450
 1274 001570
 1275 001570
 1276 001570
 1277 001570
 1278
 1279 001570
 1280 001570 012737 000004 000546
 1281 001576
 1282 001576 005037 000456
 1283 001602
 1284 001602 012737 000340 177776

```

*****
*****
**
          C I S 4   M O D U L E
**
    THIS MODULE TESTS THE LOAD DESCRIPTOR INSTRUCTIONS AND THE CHARACTER
    INSTRUCTIONS:  L2DR, L3DR, MOVC, MOVRC, MOVTC, LOCC, SKPC, SCANL,
    SPANC, CMPC, AND MATC.
**
*****
*****

.SBTTL  CIS4 INITIALIZATION
*****
:
:  CIS4 INITIALIZATION
:
*****

:IF SWR<5:0> ARE CLEARED OR IF SWR<0> IS SET THEN EXECUTE
: THE CIS4 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.

    INLINE  <SCOPE>
                                           SCOPE
    LET TEMPO : @SWR CLR.BY #177700
    MOV      @SWR,TEMPO
    BIC     #177700,TEMPO
    IF TEMPO NE #0 THEN
    TST     TEMPO
    BEQ     $50000
    IF #BIT0 NOTSETIN TEMPO THEN
    BIT     #BIT0,TEMPO
    BNE     $50001
    INLINE  <JMP  CIS4END>
    JMP    CIS4END
    ENDIF
    ENDIF
    $50001:
    $50000:

    LET CTL :- #4           ;CONTROL CHIP NUMBER
    MOV     #4,CTL
    LET ERRFLG := #0       ;INITIALIZE ERROR FLAG
    CLR    ERRFLG
    LET PSW :- #340
    MOV    #340,PSW
    
```


1341							
1342	001706				IF COUNT EQ #2 THEN		
1343	001706	023727	000470	000002		CMP	COUNT,#2
1344	001714	001006				BNE	\$50005
1345	001716				IF CC NE #0 THEN		
1346	001716	005737	000520			TST	CC
1347	001722	001402				BEQ	\$50006
1348	001724				CALL ERROR		
1349	001724	004737	060044		ENDIF	JSR	PC,ERROR
1350	001730						\$50006:
1351	001730				ELSE		
1352	001730					BR	\$50007
1353	001730	000406				\$50005:	
1354	001732				IF CC NE #17 THEN		
1355	001732					CMP	CC,#17
1356	001732	023727	000520	000017		BEQ	\$50010
1357	001740	001402			CALL ERROR		
1358	001742				ENDIF	JSR	PC,ERROR
1359	001742	004737	060044				\$50010:
1360	001746				ENDIF		\$50007:
1361	001746				IF R0 NE ALPHA THEN		
1362	001746					CMP	R0,ALPHA
1363	001746					BEQ	\$50011
1364	001746				CALL ERROR		
1365	001746	020037	002220		ENDIF	JSR	PC,ERROR
1366	001752	001402					\$50011:
1367	001754				IF R1 NE ALPHA+2 THEN		
1368	001754	004737	060044			CMP	R1,ALPHA+2
1369	001760				CALL ERROR	BEQ	\$50012
1370	001760				ENDIF	JSR	PC,ERROR
1371	001760						\$50012:
1372	001760	020137	002222		IF R2 NE BETA THEN		
1373	001764	001402				CMP	R2,BETA
1374	001766				CALL ERROR	BEQ	\$50013
1375	001766	004737	060044		ENDIF	JSR	PC,ERROR
1376	001772						\$50013:
1377	001772				IF R3 NE BETA+2 THEN		
1378	001772					CMP	R3,BETA+2
1379	001772	020237	002224		CALL ERROR	BEQ	\$50014
1380	001776	001402			ENDIF	JSR	PC,ERROR
1381	002000						\$50014:
1382	002000	004737	060044		IF R5 NE #0 THEN		
1383	002004					TST	R5
1384	002004				CALL ERROR	BEQ	\$50015
1385	002004				ENDIF	JSR	PC,ERROR
1386	002004	020337	002226				\$50015:
1387	002010	001402			IF R5 NE #0 THEN		
1388	002012					TST	R5
1389	002012	004737	060044		CALL ERROR	BEQ	\$50015
1390	002016				ENDIF	JSR	PC,ERROR
1391	002016						\$50015:
1392	002016				IF R5 NE #0 THEN		
1393	002016	005705				TST	R5
1394	002020	001402			CALL ERROR	BEQ	\$50015
1395	002022				ENDIF	JSR	PC,ERROR
1396	002022	004737	060044				\$50015:

1397	002026			ENDIF				
1398	002026							\$50015:
1399	002026			:IF COUNT EQ #2 THEN				CMP COUNT,#2
1400	002026	023727	000470					BNE \$50016
1401	002034	001005						TST R4
1402	002036			IF R4 NE #0 THEN				BEQ \$50017
1403	002036	005704						JSR PC,ERROR
1404	002040	001402						
1405	002042			CALL ERROR				
1406	002042	004737	060044					
1407	002046			ENDIF				\$50017:
1408	002046							BR \$50020
1409	002046			ELSE				\$50016:
1410	002046	000405						CMP R4,#ADR.A+4
1411	002050							BEQ \$50021
1412	002050			IF R4 NE #ADR.A+4 THEN				JSR PC,ERROR
1413	002050	020427	002216					
1414	002054	001402						
1415	002056			CALL ERROR				
1416	002056	004737	060044					
1417	002062			ENDIF				
1418	002062							\$50021:
1419	002062			ENDIF				\$50020:
1420	002062							DEC COUNT
1421	002062			LET COUNT := COUNT - #1				
1422	002062	005337	000470					TST COUNT
1423	002066			UNTIL COUNT EQ #0				BNE \$50002
1424	002066	005737	000470					
1425	002072	001255						
1426								
1427								
1428				::+ TEST THE SPECIAL CASE OF USING R7				
1429				::-				
1430								
1431	002074			LET R4 := #0				CLR R4
1432	002074	005004						
1433	002076			LET R5 := #0				CLR R5
1434	002076	005005						
1435	002100			INLINE <L2D7>				L2D7
1436	002100	076027						
1437	002102			INLINE <.WORD 240>				.WORD 240
1438	002102	000240						
1439	002104			INLINE <.WORD 402>				.WORD 402
1440	002104	000402						
1441	002106			INLINE <.WORD 403>				.WORD 403
1442	002106	000403						
1443	002110			INLINE <.WORD 240>				.WORD 240
1444	002110	000240						
1445	002112			CALL ERROR				JSR PC,ERROR
1446	002112	004737	060044					
1447	002116			IF R0 NE @#240 THEN				
1448	002116	020037	000240					CMP R0,@#240
1449	002122	001402						BEQ \$50022
1450	002124							
1451	002124	004737	060044	CALL ERROR				JSR PC,ERROR
1452	002130			ENDIF				

```

1453 002130
1454 002130
1455 002130 020137 000242
1456 002134 001402
1457 002136
1458 002136 004737 060044
1459 002142
1460 002142
1461 002142
1462 002142 020237 000402
1463 002146 001402
1464 002150
1465 002150 004737 060044
1466 002154
1467 002154
1468 002154
1469 002154 020337 000404
1470 002160 001402
1471 002162
1472 002162 004737 060044
1473 002166
1474 002166
1475 002166
1476 002166 005704
1477 002170 001402
1478 002172
1479 002172 004737 060044
1480 002176
1481 002176
1482 002176
1483 002176 005705
1484 002200 001402
1485 002202
1486 002202 004737 060044
1487 002206
1488 002206
1489 002206
1490 002206 000137 002234
1491
1492
1493
1494
1495
1496 002212 002220
1497 002214 002224
1498 002216 002230
1499 002220 177777
1500 002222 052525
1501 002224 125252
1502 002226 177777
1503 002230 107070
1504 002232 070707
    
```

```

IF R1 NE @#242 THEN
    CALL ERROR
ENDIF
IF R2 NE @#402 THEN
    CALL ERROR
ENDIF
IF R3 NE @#404 THEN
    CALL ERROR
ENDIF
IF R4 NE #0 THEN
    CALL ERROR
ENDIF
IF R5 NE #0 THEN
    CALL ERROR
ENDIF
INLINE <JMP XL3DR>
    
```

```

:↑
: TEST TABLE FOR L2DR AND L3DR
:-
    
```

```

ADR.A: .WORD ALPHA
        .WORD BETA
        .WORD GAMMA
ALPHA: .WORD 177777      : ALPHA.LEN
        .WORD 052525    : ALPHA.ADR
BETA:  .WORD 125252    : BETA.LEN
        .WORD 177777    : BETA.ADR
GAMMA: .WORD 107070    : GAMMA.LEN
        .WORD 070707    : GAMMA.ADR
    
```

```

$50022:
CMP     R1,@#242
BEQ     $50023
JSR     PC,ERROR
$50023:
CMP     R2,@#402
BEQ     $50024
JSR     PC,ERROR
$50024:
CMP     R3,@#404
BEQ     $50025
JSR     PC,ERROR
$50025:
TST     R4
BEQ     $50026
JSR     PC,ERROR
$50026:
TST     R5
BEQ     $50027
JSR     PC,ERROR
$50027:
JMP     XL3DR
    
```

```

1505 .SBTTL L3DR
1506 :*****
1507 :
1508 :       L3DR TEST
1509 : THIS TEST USES L3D3, L3D5, AND L3D7 TO TEST L3DR.
1510 :*****
1511 :
1512 :
1513 002234 INLINE <XL3DR:>
1514 002234
1515 002234 XL3DR:
1516 002234 000004 INLINE <SCOPE> SCOPE
1517 002236 LET $TIMES := #20 ;ITERATION COUNT
1518 002236 012737 000020 000404 MOV #20,$TIMES
1519 002244 LET COUNT := #2
1520 002244 012737 000002 000470 MOV #2,COUNT
1521 002252 REPEAT
1522 002252 $50030:
1523
1524 :+
1525 : SET UP THE GPR'S AND DO
1526 : THE CIS INSTRUCTION
1527 :-
1528
1529 002252 CALL CLRREG
1530 002252 004737 056604 JSR PC,CLRREG
1531 002256 IF COUNT EQ #2 THEN
1532 002256 023727 000470 000002 CMP COUNT,#2
1533 002264 001005 BNE $50031
1534 002266 LET R3 := #ADR.A
1535 002266 012703 002212 MOV #ADR.A,R3
1536 002272 INLINE <CCC>
1537 002272 000257 CCC
1538 002274 INLINE <L3D3>
1539 002274 076063 L3D3
1540 002276 ELSE
1541 002276 000404 BR $50032
1542 002300 $50031:
1543 002300 LET R5 := #ADR.A
1544 002300 012705 002212 MOV #ADR.A,R5
1545 002304 INLINE <SCC>
1546 002304 000277 SCC
1547 002306 INLINE <L3D5>
1548 002306 076065 L3D5
1549 002310 ENDIF
1550 002310 $50032:
1551 002310 LET CC := PSW
1552 002310 013737 177776 000520 MOV PSW,CC
1553 002316 LET CC := CC CLR.BY #177760
1554 002316 042737 177760 000520 BIC #177760,CC
1555 002324 LET ERR := #1 ;CODE FOR INSTRUCTION FAILURE
1556 002324 012737 000001 000552 MOV #1,ERR
1557
1558 :+
1559 : CHECK RESULTS AND REPORT ERRORS
1560 :-
    
```

1561							
1562	002332				IF COUNT EQ #2 THEN		
1563	002332	023727	000470	000002		CMP	COUNT,#2
1564	002340	001006				BNE	\$50033
1565	002342				IF CC NE #0 THEN		
1566	002342	005737	000520			TST	CC
1567	002346	001402				BEQ	\$50034
1568	002350				CALL ERROR		
1569	002350	004737	060044			JSR	PC,ERROR
1570	002354				ENDIF		
1571	002354					\$50034:	
1572	002354				ELSE		
1573	002354	000406				BR	\$50035
1574	002356					\$50033:	
1575	002356				IF CC NE #17 THEN		
1576	002356	023727	000520	000017		CMP	CC,#17
1577	002364	001402				BEQ	\$50036
1578	002366				CALL ERROR		
1579	002366	004737	060044			JSR	PC,ERROR
1580	002372				ENDIF		
1581	002372					\$50036:	
1582	002372				ENDIF		
1583	002372					\$50035:	
1584	002372				IF R0 NE ALPHA THEN		
1585	002372	020037	002220			CMP	R0,ALPHA
1586	002376	001402				BEQ	\$50037
1587	002400				CALL ERROR		
1588	002400	004737	060044			JSR	PC,ERROR
1589	002404				ENDIF		
1590	002404					\$50037:	
1591	002404				IF R1 NE ALPHA+2 THEN		
1592	002404	020137	002222			CMP	R1,ALPHA+2
1593	002410	001402				BEQ	\$50040
1594	002412				CALL ERROR		
1595	002412	004737	060044			JSR	PC,ERROR
1596	002416				ENDIF		
1597	002416					\$50040:	
1598	002416				IF R2 NE BETA THEN		
1599	002416	020237	002224			CMP	R2,BETA
1600	002422	001402				BEQ	\$50041
1601	002424				CALL ERROR		
1602	002424	004737	060044			JSR	PC,ERROR
1603	002430				ENDIF		
1604	002430					\$50041:	
1605	002430				IF R3 NE BETA+2 THEN		
1606	002430	020337	002226			CMP	R3,BETA+2
1607	002434	001402				BEQ	\$50042
1608	002436				CALL ERROR		
1609	002436	004737	060044			JSR	PC,ERROR
1610	002442				ENDIF		
1611	002442					\$50042:	
1612	002442				IF R4 NE GAMMA THEN		
1613	002442	020437	002230			CMP	R4,GAMMA
1614	002446	001402				BEQ	\$50043
1615	002450				CALL ERROR		
1616	002450	004737	060044			JSR	PC,ERROR

1617	002454		ENDIF		
1618	002454				\$50043:
1619	002454		IF R5 NE GAMMA+2 THEN		
1620	002454	020537			CMP R5,GAMMA+2
1621	002460	001402			BEQ \$50044
1622	002462		CALL ERROR		
1623	002462	004737			JSR PC,ERROR
1624	002466		ENDIF		
1625	002466				\$50044:
1626	002466		LET COUNT := COUNT - #1		
1627	002466	005337			DEC COUNT
1628	002472		UNTIL COUNT EQ #0		
1629	002472	005737			TST COUNT
1630	002476	001265			BNE \$50030
1631					
1632			::+		
1633			:: TEST THE SPECIAL CASE OF USING R7		
1634			::-		
1635					
1636	002500		INLINE <L3D7>		L3D7
1637	002500	076067			
1638	002502		INLINE <.WORD 240> ;NOP		.WORD 240
1639	002502	000240			
1640	002504		INLINE <.WORD 240> ;NOP		.WORD 240
1641	002504	000240			
1642	002506		INLINE <.WORD 402> ;BR TO ERROR CALL		.WORD 402
1643	002506	000402			
1644	002510		INLINE <.WORD 403> ;BR OVER ERROR CALL		.WORD 403
1645	002510	000403			
1646	002512		INLINE <.WORD 240> ;NOP		.WORD 240
1647	002512	000240			
1648	002514		CALL ERROR		
1649	002514	004737			JSR PC,ERROR
1650	002520		IF R0 NE @#240 THEN		
1651	002520	020037			CMP R0,@#240
1652	002524	001402			BEQ \$50045
1653	002526		CALL ERROR		
1654	002526	004737			JSR PC,ERROR
1655	002532		ENDIF		
1656	002532				\$50045:
1657	002532		IF R1 NE @#242 THEN		
1658	002532	020137			CMP R1,@#242
1659	002536	001402			BEQ \$50046
1660	002540		CALL ERROR		
1661	002540	004737			JSR PC,ERROR
1662	002544		ENDIF		
1663	002544				\$50046:
1664	002544		IF R2 NE @#240 THEN		
1665	002544	020237			CMP R2,@#240
1666	002550	001402			BEQ \$50047
1667	002552		CALL ERROR		
1668	002552	004737			JSR PC,ERROR
1669	002556		ENDIF		
1670	002556				\$50047:
1671	002556		IF R3 NE @#242 THEN		
1672	002556	020337			CMP R3,@#242

1673	002562	001402			BEQ	\$50050
1674	002564			CALL ERROR		
1675	002564	004737	060044		JSR	PC,ERROR
1676	002570			ENDIF		
1677	002570				\$50050:	
1678	002570			:F R3 NE @#242 THEN		
1679	002570	020337	000242		CMP	R3,@#242
1680	002574	001402			BEQ	\$50051
1681	002576			CALL ERROR		
1682	002576	004737	060044		JSR	PC,ERROR
1683	002602			ENDIF		
1684	002602				\$50051:	
1685	002602			IF R4 NE @#402 THEN		
1686	002602	020437	000402		CMP	R4,@#402
1687	002606	001402			BEQ	\$50052
1688	002610			CALL ERROR		
1689	002610	004737	060044		JSR	PC,ERROR
1690	002614			ENDIF		
1691	002614				\$50052:	
1692	002614			IF R5 NE @#404 THEN		
1693	002614	020537	000404		CMP	R5,@#404
1694	002620	001402			BEQ	\$50053
1695	002622			CALL ERROR		
1696	002622	004737	060044		JSR	PC,ERROR
1697	002626			ENDIF		
1698	002626				\$50053:	
1699	002626			INLINE <JMP XMOVC>		
1700	002626	000137	002632		JMP	XMOVC

```

1701      .SBTTL  MOV
1702      :*****
1703      :
1704      :   MOV  TEST
1705      :
1706      :*****
1707
1708 002632  INLINE  <XMOV:>
1709 002632
1710 002632
1711 002632 000004
1712 002634
1713 002634 012737 000020 000404
1714 002642
1715 002642 013737 004154 000470
1716 002650
1717 002650 012737 004156 000544
1718
1719 002656
1720 002656
1721
1722
1723      :+
1724      :  GET TEST DATA FROM TABLE
1725      :-
1726 002656
1727 002656 013700 000544
1728 002662
1729 002662 012037 000562
1730 002666
1731 002666 012037 000564
1732 002672
1733 002672 012037 000566
1734 002676
1735 002676 012037 000570
1736 002702
1737 002702 012037 000572
1738 002706
1739 002706 012037 000574
1740 002712
1741 002712 011037 000576
1742
1743
1744
1745
1746
1747
1748
1749 002716
1750 002716 013700 000562
1751 002722
1752 002722 013701 000564
1753 002726
1754 002726 013702 000566
1755 002732
1756 002732 013703 000570
    
```

```

XMOV:
SCOPE
:ITERATION COUNT
MOV #20,$TIMES
:GET TEST COUNT
MOV TMOV,COUNT
:POINT TO FIRST TEST CASE
MOV #TMOV+2,TSTCAS
$5030:
    
```

```

MOV TSTCAS,R0
:SRC.LEN
MOV (R0)+,DATA00
:SRC.ADR
MOV (R0)+,DATA01
:DST.LEN
MOV (R0)+,DATA02
:DST.ADR
MOV (R0)+,DATA03
:FILL CHAR.
MOV (R0)+,DATA04
:SRC.BUFF.STATE
MOV (R0)+,DATA05
:DST.BUFF.STATE
MOV (R0),DATA06
    
```

```

MOV DATA00,R0
MOV DATA01,R1
MOV DATA02,R2
MOV DATA03,R3
    
```


1757	002736			LET R4 := DATA04		
1758	002736	013704	000572		MOV	DATA04,R4
1759	002742			IF #1 SETIN COUNT THEN		
1760	002742	032737	000001		BIT	#1,COUNT
1761	002750	001403			BEQ	\$50054
1762	002752			LET R5 := #177777		
1763	002752	012705	177777		MOV	#177777,R5
1764	002756			ELSE		
1765	002756	000401			BR	\$50055
1766	002760				\$50054:	
1767	002760			LET R5 := #0		
1768	002760	005005		ENDIF	CLR	R5
1769	002762				\$50055:	
1770	002762			INLINE <CMP R0,R2>		
1771	002762				CMP	R0,R2
1772	002762	020002		LET CC.EXP := PSW		
1773	002764				MOV	PSW,CC.EXP
1774	002764	013737	177776	LET CC.EXP := CC.EXP CLR.BY #177760		
1775	002772				BIC	#177760,CC.EXP
1776	002772	042737	177760	IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN		
1777	003000				BIT	#CBIT,CC.EXP
1778	003000	032737	000001		BNE	\$50056
1779	003006	001004			BIT	#ZBIT,CC.EXP
1780	003010	032737	000004		BEQ	\$50057
1781	003016	001403			\$50056:	
1782	003020			LET R0.EXP := #0		
1783	003020				CLR	R0.EXP
1784	003020	005037	000524	ELSE		
1785	003024				BR	\$50060
1786	003024	000404			\$50057:	
1787	003026			LET R0.EXP := R0 - R2		;SRC.LEN - DST.LEN
1788	003026				MOV	R0,R0.EXP
1789	003026	010037	000524		SUB	R2,R0.EXP
1790	003032	160237	000524			
1791	003036			ENDIF		\$50060:
1792	003036			LET R1.EXP := #0		
1793	003036				CLR	R1.EXP
1794	003036	005037	000526	LET R2.EXP := #0		
1795	003042				CLR	R2.EXP
1796	003042	005037	000530	LET R3.EXP := #0		
1797	003046				CLR	R3.EXP
1798	003046	005037	000532	LET R4.EXP := R4		
1799	003052				MOV	R4,R4.EXP
1800	003052	010437	000534	LET R5.EXP := R5		
1801	003056				MOV	R5,R5.EXP
1802	003056	010537	000536	LET BFADRS := R1		;SRC.ADR
1803	003062				MOV	R1,BFADRS
1804	003062	010137	000542	IF DATA05 EQ #0 THEN		
1805	003066				TST	DATA05
1806	003066	005737	000574		BNE	\$50061
1807	003072	001003				
1808	003074			CALL CLRBUF		
1809	003074	004737	056622		JSR	PC,CLRBUF
1810	003100			ELSE		
1811	003100	000411			BR	\$50062
1812	003102				\$50061:	

1813	003102				IF DATA05 EQ #177777 THEN		
1814	003102	023727	000574	177777		CMP	DATA05,#177777
1815	003110	001003				BNE	\$50063
1816	003112				CALL SETBUF		
1817	003112	004737	056664			JSR	PC,SETBUF
1818	003116				ELSE		
1819	003116	000402				BR	\$50064
1820	003120					\$50063:	
1821	003120				CALL ALTBUF		
1822	003120	004737	056730			JSR	PC,ALTBUF
1823	003124				ENDIF		
1824	003124					\$50064:	
1825	003124				ENDIF		
1826	003124					\$50062:	
1827	003124				LET BFADRS := R3	:DST.ADR	
1828	003124	010337	000542			MOV	R3,BFADRS
1829	003130				IF DATA06 EQ #0 THEN		
1830	003130	005737	000576			TST	DATA06
1831	003134	001003				BNE	\$50065
1832	003136				CALL CLRBUF		
1833	003136	004737	056622			JSP	PC,CLRBUF
1834	003142				ELSE		
1835	003142	000411				BR	\$50066
1836	003144					\$50065:	
1837	003144				IF DATA06 EQ #177777 THEN		
1838	003144	023727	000576	177777		CMP	DATA06,#177777
1839	003152	001003				BNE	\$50067
1840	003154				CALL SETBUF		
1841	003154	004737	056664			JSR	PC,SETBUF
1842	003160				ELSE		
1843	003160	000402				BR	\$50070
1844	003162					\$50067:	
1845	003162				CALL ALTBUF		
1846	003162	004737	056730			JSR	PC,ALTBUF
1847	003166				ENDIF		
1848	003166					\$50070:	
1849	003166				ENDIF		
1850	003166					\$50066:	
1851	003166				IF DATA02 EQ #0 THEN		
1852	003166	005737	000566			TST	DATA02
1853	003172	001003				BNE	\$50071
1854	003174				LET BF.EXP := (R3)		
1855	003174	011337	000540			MOV	(R3),BF.EXP
1856	003200				ELSE		
1857	003200	000402				BR	\$50072
1858	003202					\$50071:	
1859	003202				LET BF.EXP := (R1)		
1860	003202	011137	000540			MOV	(R1),BF.EXP
1861	003206				ENDIF		
1862	003206					\$50072:	
1863							
1864					::		
1865					:: DO THE CIS INSTRUCTION		
1866					::		
1867							
1868	003206				INLINE <MOV C>		

1869	003206	076030				MOV C
1870						
1871						
1872					:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS	
1873					:-	
1874						
1875	003210				LET CC := PSW	
1876	003210	013737	177776	000520		MOV PSW,CC
1877	003216				LET CC := CC CLR.BY #177760	
1878	003216	042737	177760	000520		BIC #177760,CC
1879	003224				IF CC NE CC.EXP THEN	
1880	003224	023737	000520	000522		CMP CC,CC.EXP
1881	003232	001402				BEQ \$50073
1882	003234				CALL ERROR	
1883	003234	004737	060044			JSR PC,ERROR
1884	003240				ENDIF	
1885	003240					\$50073:
1886	003240				IF R0 NE R0.EXP THEN	
1887	003240	020037	000524			CMP R0,R0.EXP
1888	003244	001402				BEQ \$50074
1889	003246				CALL ERROR	
1890	003246	004737	060044			JSR PC,ERROR
1891	003252				ENDIF	
1892	003252					\$50074:
1893	003252				IF R1 NE R1.EXP THEN	
1894	003252	020137	000526			CMP R1,R1.EXP
1895	003256	001402				BEQ \$50075
1896	003260				CALL ERROR	
1897	003260	004737	060044			JSR PC,ERROR
1898	003264				ENDIF	
1899	003264					\$50075:
1900	003264				IF R2 NE R2.EXP THEN	
1901	003264	020237	000530			CMP R2,R2.EXP
1902	003270	001402				BEQ \$50076
1903	003272				CALL ERROR	
1904	003272	004737	060044			JSR PC,ERROR
1905	003276				ENDIF	
1906	003276					\$50076:
1907	003276				IF R3 NE R3.EXP THEN	
1908	003276	020337	000532			CMP R3,R3.EXP
1909	003302	001402				BEQ \$50077
1910	003304				CALL ERROR	
1911	003304	004737	060044			JSR PC,ERROR
1912	003310				ENDIF	
1913	003310					\$50077:
1914	003310				IF R4 NE R4.EXP THEN	
1915	003310	020437	000534			CMP R4,R4.EXP
1916	003314	001402				BEQ \$50100
1917	003316				CALL ERROR	
1918	003316	004737	060044			JSR PC,ERROR
1919	003322				ENDIF	
1920	003322					\$50100:
1921	003322				IF R5 NE R5.EXP THEN	
1922	003322	020537	000536			CMP R5,R5.EXP
1923	003326	001402				BEQ \$50101
1924	003330				CALL ERROR	

1925	003330	004737	060044			JSR	PC,ERROR
1926	003334			ENDIF			
1927	003334					\$50101:	
1928							
1929				:+			
1930				:+ CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS			
1931				:-			
1932							
1933	003334			LET R0 := DATA01	:SRC.ADR		
1934	003334	013700	000564			MOV	DATA01,R0
1935	003340			LET R1 := DATA03	:DST.ADR		
1936	003340	013701	000570			MOV	DATA03,R1
1937	003344			LET ERRFLG := #0			
1938	003344	005037	000456			CLR	ERRFLG
1939	003350			IF DATA00 HIS DATA02	THEN		
1940	003350	023737	000562	000566		CMP	DATA00,DATA02
1941	003356	103417				BLO	\$50102
1942	003360			LET R3 := DATA02			
1943	003360	013703	000566			MOV	DATA02,R3
1944	003364			WHILE R3 NE #0 AND ERRFLG EQ #0 DO			
1945	003364					\$50103:	
1946	003364	005703				TST	R3
1947	003366	001412				BEQ	\$50104
1948	003370	005737	000456			TST	ERRFLG
1949	003374	001007				BNE	\$50104
1950	003376			IFB BF.EXP NE (R1)+	THEN		
1951	003376	123721	000540			CMPB	BF.EXP,(R1)+
1952	003402	001402				BEQ	\$50105
1953	003404			CALL ERROR			
1954	003404	004737	060044			JSR	PC,ERROR
1955	003410			ENDIF			
1956	003410					\$50105:	
1957	003410			LET R3 :- R3 - #1			
1958	003410	005303				DEC	R3
1959	003412			ENDDO			
1960	003412	000764				BR	\$50103
1961	003414					\$50104:	
1962	003414			ELSE			
1963	003414	000441				BR	\$50106
1964	003416					\$50102:	
1965	003416			LET R3 :- DATA00	:SRC.LEN		
1966	003416	013703	000562			MOV	DATA00,R3
1967	003422			WHILE R3 NE #0 AND ERRFLG EQ #0 DO			
1968	003422					\$50107:	
1969	003422	005703				TST	R3
1970	003424	001412				BEQ	\$50110
1971	003426	005737	000456			TST	ERRFLG
1972	003432	001007				BNE	\$50110
1973	003434			IFB BF.EXP NE (R1)+	THEN		
1974	003434	123721	000540			CMPB	BF.EXP,(R1)+
1975	003440	001402				BEQ	\$50111
1976	003442			CALL ERROR			
1977	003442	004737	060044			JSR	PC,ERROR
1978	003446			ENDIF			
1979	003446					\$50111:	
1980	003446			LET R3 := R3 - #1			

1981	003446	005303				DEC	R3
1982	003450			ENDDO		BR	\$50107
1983	003450	000764				\$50110:	
1984	003452			LET R3 := DATA02 - DATA00		MOV	DATA02,R3
1985	003452					SUB	DATA00,R3
1986	003452	013703	000566			MOV	DATA03,R1
1987	003456	163703	000562			ADD	DATA00,R1
1988	003462			LET R1 := DATA03 + DATA00		\$50112:	
1989	003462	013701	000570			CMPB	(R1)+,DATA04
1990	003466	063701	000562			BEQ	\$50113
1991	003472			REPEAT		JSR	PC,ERROR
1992	003472			IFB (R1)+ NE DATA04 THEN		\$50113:	
1993	003472					DEC	R3
1994	003472	122137	000572			TST	R3
1995	003476	001402		CALL ERROR		BEQ	\$50114
1996	003500			ENDIF		TST	ERRFLG
1997	003500	004737	060044			BEQ	\$50112
1998	003504			LET R3 := R3 - #1		\$50114:	
1999	003504			UNTIL R3 EQ #0 OR ERRFLG NE #0		\$50106:	
2000	003504					CLR	ERRFLG
2001	003504	005303				CMP	DATA02,#BUFLEN
2002	003506	005703				BHIS	\$50115
2003	003510	001403				CMP	DATA06,#1
2004	003512	005737	000456			BNE	\$50116
2005	003516	001765		ENDIF		MOV	#125252,BF.EXP
2006	003520					BR	\$50117
2007	003520			LET ERRFLG := #0		\$50116:	
2008	003520			IF DATA02 LO #BUFLEN THEN		MOV	DATA06,BF.EXP
2009	003520					\$50117:	
2010	003520	005037	000456			MOV	DATA02,R0
2011	003524			IF DATA06 EQ #1 THEN		ADD	DATA03,R0
2012	003524	023727	000566			\$50117:	
2013	003532	103040	000100			MOV	DATA02,R0
2014	003534			LET BF.EXP := #125252		ADD	DATA03,R0
2015	003534			ELSE		\$50117:	
2016	003534	023727	000576			MOV	DATA03,TEMPO
2017	003542	001004	000001			ADD	#BUFLEN,TEMPO
2018	003544			LET BF.EXP := DATA06		\$50120:	
2019	003544	012777	125252			CMPB	(R0)+,BF.EXP
2020	003552			ENDIF			
2021	003552	000403					
2022	003554			LET R0 := DATA02 + DATA03			
2023	003554						
2024	003554	013737	000576				
2025	003562			LET TEMPO := DATA03 + #BUFLEN			
2026	003562						
2027	003562			REPEAT			
2028	003562	013700	000566				
2029	003566	063700	000570				
2030	003572			IFB (R0)+ NE BF.EXP THEN			
2031	003572	013737	000570				
2032	003600	062737	000100				
2033	003606						
2034	003606						
2035	003606						
2036	003606	122037	000540				

2037 003612 001402
 2038 003614
 2039 003614 004737 060044
 2040 003620
 2041 003620
 2042 003620
 2043 003620 020037 000476
 2044 003624 001403
 2045 003626 005737 000456
 2046 003632 001765
 2047 003634
 2048 003634
 2049 003634
 2050 003634
 2051 003634 062737 000016 000544
 2052 003642
 2053 003642 005337 000470
 2054
 2055 003646
 2056 003646 005737 000470
 2057 003652 001402
 2058 003654
 2059 003654 000137 002656
 2060 003660
 2061 003660
 2062
 2063
 2064
 2065
 2066
 2067

CALL ERROR
 ENDIF
 UNTIL R0 EQ TEMPO OR ERRFLG NE #0
 ENDIF
 LET TSTCAS := TSTCAS + #16
 LET COUNT := COUNT - #1
 :UNTIL COUNT EQ #0
 IF COUNT NE #0 THEN
 INLINE <JMP \$5030>
 ENDIF
 :+
 : DO ONE IN-LINE CASE - MOVCI
 :-

BEQ \$50121
 JSR PC,ERROR
 \$50121:
 CMP R0,TEMPO
 BEQ \$50122
 TST ERRFLG
 BEQ \$50120
 \$50122:
 \$50115:
 ADD #16,TSTCAS
 DEC COUNT
 TST COUNT
 BEQ \$50123
 JMP \$5030
 \$50123:

2068	003660				LET BFADRS := #BUFF1		
2069	003660	012737	063072	000542		MOV	#BUFF1,BFADRS
2070	003666				CALL SETBUF		
2071	003666	004737	056664			JSR	PC,SETBUF
2072	003672				LET BFADRS := #BUFF2		
2073	003672	012737	063172	000542		MOV	#BUFF2,BFADRS
2074	003700				CALL CLRBUF		
2075	003700	004737	056622			JSR	PC,CLRBUF
2076	003704				LET R0 := #125252		
2077	003704	012700	125252			MOV	#125252,R0
2078	003710				LET R1 := #125252		
2079	003710	012701	125252			MOV	#125252,R1
2080	003714				LET R2 := #125252		
2081	003714	012702	125252			MOV	#125252,R2
2082	003720				LET R3 := #125252		
2083	003720	012703	125252			MOV	#125252,R3
2084	003724				LET R4 := #125252		
2085	003724	012704	125252			MOV	#125252,R4
2086	003730				LET R5 := #125252		
2087	003730	012705	125252			MOV	#125252,R5
2088	003734				LET ERRFLG := #0		
2089	003734	005037	000456			CLR	ERRFLG
2090	003740				INLINE <MOVCI>		
2091	003740	076130				MOVCI	
2092	003742				INLINE <.WORD TMOVCI>	:SRC.DSCR.PTR	
2093	003742	004500				.WORD	TMOVCI
2094	003744				INLINE <.WORD TMOVCI+4>	:DST.DSCR.PTR	
2095	003744	004504				.WORD	TMOVCI+4
2096	003746				INLINE <.WORD 40>	:FILL CHARACTER	
2097	003746	000040				.WORD	40
2098	003750				IFCOND CS THEN		
2099	003750	103003				BCC	\$50124
2100	003752				IFCOND EQ THEN		
2101	003752	001002				BNE	\$50125
2102	003754				CALL ERROR		
2103	003754	004737	060044			JSR	PC,ERROR
2104	003760				ENDIF		
2105	003760					\$50125:	
2106	003760				ENDIF		
2107	003760					\$50124:	
2108	003760				IF R0 NE #125252 THEN		
2109	003760	020027	*25252			CMP	R0,#125252
2110	003764	001402				BEQ	\$50126
2111	003766				CALL ERROR		
2112	003766	004737	060044			JSR	PC,ERROR
2113	003772				ENDIF		
2114	003772					\$50126:	
2115	003772				IF R1 NE #125252 THEN		
2116	003772	020127	125252			CMP	R1,#125252
2117	003776	001402				BEQ	\$50127
2118	004000				CALL ERROR		
2119	004000	004737	060044			JSR	PC,EPROR
2120	004004				ENDIF		
2121	004004					\$50127:	
2122	004004				IF R2 NE #125252 THEN		
2123	004004	020227	125252			CMP	R2,#125252

2124 004010 001402
2125 004012
2126 004012 004737 060044
2127 004016
2128 004016
2129 004016
2130 004016 020327 125252
2131 004022 001402

CALL ERROR
ENDIF
IF R3 NE #125252 THEN

BEQ \$50130
JSR PC,ERROR
\$50130:
CMP R3,#125252
BEQ \$50131

2132	004024			CALL ERROR							
2133	004024	004737	060044							JSR	PC,ERROR
2134	004030			ENDIF							
2135	004030									\$50131:	
2136	004030			IF R4 NE #125252 THEN							
2137	004030	020427	125252							CMP	R4,#125252
2138	004034	001402								BEQ	\$50132
2139	004036			CALL ERROR							
2140	004036	004737	060044							JSR	PC,ERROR
2141	004042			ENDIF							
2142	004042									\$50132:	
2143	004042			IF R5 NE #125252 THEN							
2144	004042	020527	125252							CMP	R5,#125252
2145	004046	001402								BEQ	\$50133
2146	004050			CALL ERROR							
2147	004050	004737	060044							JSR	PC,ERROR
2148	004054			ENDIF							
2149	004054									\$50133:	
2150	004054			IF BUFF2 NE #177777 THEN							
2151	004054	023727	063172	177777						CMP	BUFF2,#177777
2152	004062	001402								BEQ	\$50134
2153	004064			CALL ERROR							
2154	004064	004737	060044							JSR	PC,ERROR
2155	004070			ENDIF							
2156	004070									\$50134:	
2157	004070			IF BUFF2+2 NE #20040 THEN							
2158	004070	023727	063174	020040						CMP	BUFF2+2,#20040
2159	004076	001402								BEQ	\$50135
2160	004100			CALL ERROR							
2161	004100	004737	060044							JSR	PC,ERROR
2162	004104			ENDIF							
2163	004104									\$50135:	
2164	004104			LET R0 := #BUFF2+4						MOV	#BUFF2+4,R0
2165	004104	012700	063176								
2166	004110			LET TEMPO := #BUFF2 + #BUFLN-2							
2167	004110	012,37	063172	000476						MOV	#BUFF2,TEMPO
2168	004116	062737	000076	000476						ADD	#BUFLN-2,TEMPO
2169	004124			REPEAT							
2170	004124									\$50136:	
2171	004124			IF (R0)+ NE #0 THEN							
2172	004124	005720								TST	(R0)+
2173	004126	001402								BEQ	\$50137
2174	004130			CALL ERROR							
2175	004130	004737	060044							JSR	PC,ERROR
2176	004134			ENDIF							
2177	004134									\$50137:	
2178	004134			UNTIL R0 EQ TEMPO OR ERRFLG NE #0							
2179	004134	020037	000476							CMP	R0,TEMPO
2180	004140	001403								BEQ	\$50140
2181	004142	005737	000456							TST	ERRFLG
2182	004146	001766								BEQ	\$50136
2183	004150									\$50140:	
2184	004150			INLINE <,MP XMOVRC>							
2185	004150	000137	004510							JMP	XMOVRC
2186											
2187				;+							

2188

; TABLE OF TEST CASES FOR MOVC

2189					
2190					
2191	004154	000017	TMOV: .WORD 15.		;# OF TEST CASES FOR MOV
2192					
2193	004156	000040	.WORD 40		;SRC.LEN

2194	004160	063172	.WORD	BUFF2		:SRC.ADR
2195	004162	000000	.WORD	0		:DST.LEN
2196	004164	063072	.WORD	BUFF1		:DST.ADR
2197	004166	000040	.WORD	40		:FILL.CHAR.
2198	004170	000000	.WORD	0		:SRC.BUFF.STATE
2199	004172	177777	.WORD	-1		:DST.BUFF.STATE
2200						
2201	004174	000050	.WORD	50		;TEST CASE NUMBER 2
2202	004176	063172	.WORD	BUFF2		
2203	004200	000020	.WORD	20		
2204	004202	063072	.WORD	BUFF1		
2205	004204	000100	.WORD	100		
2206	004206	177777	.WORD	-1		
2207	004210	000000	.WORD	0		
2208						
2209	004212	000045	.WORD	45		;TEST CASE NUMBER 3
2210	004214	063172	.WORD	BUFF2		
2211	004216	000030	.WORD	30		
2212	004220	063072	.WORD	BUFF1		
2213	004222	000007	.WORD	7		
2214	004224	000000	.WORD	0		
2215	004226	000001	.WORD	1		
2216						
2217	004230	000000	.WORD	0		;TEST CASE NUMBER 4
2218	004232	063172	.WORD	BUFF2		
2219	004234	000060	.WORD	60		
2220	004236	063072	.WORD	BUFF1		
2221	004240	000010	.WORD	10		
2222	004242	000001	.WORD	1		
2223	004244	000000	.WORD	0		
2224						
2225	004246	000046	.WORD	46		;TEST CASE NUMBER 5
2226	004250	063172	.WORD	BUFF2		
2227	004252	000132	.WORD	BUFLLEN+32		
2228	004254	063072	.WORD	BUFF1		
2229	004256	000077	.WORD	77		
2230	004260	177777	.WORD	-1		
2231	004262	000001	.WORD	1		
2232						
2233	004264	000000	.WORD	0		;TEST CASE NUMBER 6
2234	004266	063172	.WORD	BUFF2		
2235	004270	000015	.WORD	15		
2236	004272	063072	.WORD	BUFF1		
2237	004274	000060	.WORD	60		
2238	004276	000001	.WORD	1		
2239	004300	177777	.WORD	-1		
2240						
2241	004302	000043	.WORD	43		;TEST CASE NUMBER 7
2242	004304	063172	.WORD	BUFF2		
2243	004306	000070	.WORD	70		
2244	004310	063072	.WORD	BUFF1		
2245	004312	000370	.WORD	370		
2246	004314	000000	.WORD	0		
2247	004316	177777	.WORD	-1		
2248						
2249	004320	000060	.WORD	60		;TEST CASE NUMBER 8

2250	004322	063072	.WORD	BUFF1	
2251	004324	000000	.WORD	0	
2252	004326	063172	.WORD	BUFF2	
2253	004330	000200	.WORD	200	
2254	004332	177777	.WORD	-1	
2255	004334	000000	.WORD	0	
2256					
2257	004336	000120	.WORD	BUFLen+20	;TEST CASE NUMBER 9
2258	004340	063072	.WORD	BUFF1	
2259	004342	000100	.WORD	100	
2260	004344	063172	.WORD	BUFF2	
2261	004346	000002	.WORD	2	
2262	004350	000000	.WORD	0	
2263	004352	000001	.WORD	1	
2264					
2265	004354	000015	.WORD	15	;TEST CASE NUMBER 10
2266	004356	063072	.WORD	BUFF1	
2267	004360	000000	.WORD	0	
2268	004362	063172	.WORD	BUFF2	
2269	004364	000016	.WORD	16	
2270	004366	000001	.WORD	1	
2271	004370	000000	.WORD	0	
2272					
2273	004372	000077	.WORD	77	;TEST CASE NUMBER 11
2274	004374	063072	.WORD	BUFF1	
2275	004376	000021	.WORD	21	
2276	004400	063172	.WORD	BUFF2	
2277	004402	000300	.WORD	300	
2278	004404	177777	.WORD	-1	
2279	004406	000001	.WORD	1	
2280					
2281	004410	000000	.WORD	0	;TEST CASE NUMBER 12
2282	004412	063072	.WORD	BUFF1	
2283	004414	000050	.WORD	50	
2284	004416	063172	.WORD	BUFF2	
2285	004420	000003	.WORD	3	
2286	004422	000001	.WORD	1	
2287	004424	177777	.WORD	-1	
2288					
2289	004426	000014	.WORD	14	;TEST CASE NUMBER 13
2290	004430	063072	.WORD	BUFF1	
2291	004432	000042	.WORD	42	
2292	004434	063172	.WORD	BUFF2	
2293	004436	000030	.WORD	30	
2294	004440	000000	.WORD	0	
2295	004442	177777	.WORD	-1	
2296					
2297	004444	000000	.WORD	0	;TEST CASE NUMBER 14
2298	004446	063072	.WORD	BUFF1	
2299	004450	000053	.WORD	53	
2300	004452	063172	.WORD	BUFF2	
2301	004454	000012	.WORD	12	
2302	004456	000001	.WORD	1	
2303	004460	000000	.WORD	0	
2304					
2305	004462	000027	.WORD	27	;TEST CASE NUMBER 15

2306	004464	063072	.WORD	BUFF1
2307	004466	000060	.WORD	60
2308	004470	063172	.WORD	BUFF2
2309	004472	000014	.WORD	14
2310	004474	177777	.WORD	-1
2311	004476	000001	.WORD	1
2312				
2313				
2314	004500	000002	TMOVCI: .WORD	2
2315	004502	063072	.WORD	BUFF1
2316	004504	000004	.WORD	4
2317	004506	063172	.WORD	BUFF2

;TEST CASE FOR IN-LINE

```

2318 .SBTTL MOVRC
2319 :*****
2320 :
2321 : MOVRC TEST
2322 :
2323 :*****
2324 :
2325 004510 INLINE <XMOVRC:> XMOVRC:
2326 004510
2327 004510 INLINE <SCOPE> SCOPE
2328 004510 000004
2329 004512 LET $TIMES := #20 ;ITERATION COUNT
2330 004512 012737 000020 000404 MOV #20,$TIMES
2331 004520 LET COUNT := TMOVRC ;GET TEST COUNT
2332 004520 013737 006020 000470 MOV TMOVRC,COUNT
2333 004526 LET TSTCAS := #TMOVRC+2 ;POINT TO FIRST TEST CASE
2334 004526 012737 006022 000544 MOV #TMOVRC+2,TSTCAS
2335 ;REPEAT
2336 004534 INLINE <$5031:> $5031:
2337 004534
2338
2339 :+
2340 : GET TEST DATA FROM TABLE
2341 :-
2342
2343 004534 LET R0 := TSTCAS
2344 004534 013700 000544 MOV TSTCAS,R0
2345 004540 LET DATA00 := (R0)+ ;SRC.LEN
2346 004540 012037 000562 MOV (R0)+,DATA00
2347 004544 LET DATA01 := (R0)+ ;SRC.ADR
2348 004544 012037 000564 MOV (R0)+,DATA01
2349 004550 LET DATA02 := (R0)+ ;DST.LEN
2350 004550 012037 000566 MOV (R0)+,DATA02
2351 004554 LET DATA03 := (R0)+ ;DST.ADR
2352 004554 012037 000570 MOV (R0)+,DATA03
2353 004560 LET DATA04 := (R0)+ ;FILL CHAR.
2354 004560 012037 000572 MOV (R0)+,DATA04
2355 004564 LET DATA05 := (R0)+ ;SRC.BUFF.STATE
2356 004564 012037 000574 MOV (R0)+,DATA05
2357 004570 LET DATA06 := (R0) ;DST.BUFF.STATE
2358 004570 011037 000576 MOV (R0),DATA06
2359
2360 :+
2361 : SET UP GPR'S, EXPECTED GPR'S AND
2362 : CONDITION CODES, AND BUFFERS.
2363 :-
2364
2365 004574 LET R0 := DATA00
2366 004574 013700 000562 MOV DATA00,R0
2367 004600 LET R1 := DATA01
2368 004600 013701 000564 MOV DATA01,R1
2369 004604 LET R2 := DATA02
2370 004604 013702 000566 MOV DATA02,R2
2371 004610 LET R3 := DATA03
2372 004610 013703 000570 MOV DATA03,R3
2373 004614 LET R4 := DATA04
    
```

2374	004614	013704	000572			MOV	DATA04,R4
2375	004620				IF #1 SETIN COUNT THEN	BIT	#1,COUNT
2376	004620	032737	000001	000470		BEQ	\$50141
2377	004626	001403			LET R5 := #177777	MOV	#177777,R5
2378	004630				ELSE	BR	\$50142
2379	004630	012705	177777			\$50141:	
2380	004634					CLR	R5
2381	004634	000401			LET R5 :- #0	\$50142:	
2382	004636				ENDIF	CMP	R0,R2
2383	004636				INLINE <CMP R0,R2>	MOV	PSW,CC.EXP
2384	004636	005005			LET CC.EXP := PSW	BIC	#177760,CC.EXP
2385	004640				LET CC.EXP := CC.EXP CLR.BY #177760	BIT	#CBIT,CC.EXP
2386	004640				IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN	BNE	\$50143
2387	004640					BIT	#ZBIT,CC.EXP
2388	004640	020002				BEQ	\$50144
2389	004642					\$50143:	
2390	004642	013737	177776	000522	LET R0.EXP := #0	CLR	R0.EXP
2391	004650				ELSE	BR	\$50145
2392	004650	042737	177760	000522		\$50144:	
2393	004656				LET R0.EXP := R0 - R2	MOV	R0,R0.EXP
2394	004656	032737	000001	000522	ENDIF	SUB	R2,R0.EXP
2395	004664	001004			LET R1.EXP :- #0	\$50145:	
2396	004666	032737	000004	000522	LET R2.EXP := #0	CLR	R1.EXP
2397	004674	001403			LET R3.EXP := #0	CLR	R2.EXP
2398	004676				LET R4.EXP :- R4	CLR	R3.EXP
2399	004676				LET R5.EXP := R5	MOV	R4,R4.EXP
2400	004676	005037	000524		LET BFADRS := DATA01	MOV	R5,R5.EXP
2401	004702				IF DATA05 EQ #0 THEN	MOV	
2402	004702	000404			CALL CLRBUF	TST	DATA05
2403	004704				ELSE	BNE	\$50146
2404	004704					JSR	PC,CLRBUF
2405	004704	010037	000524			BR	\$50147
2406	004710	160237	000524		IF DATA05 EQ #177777 THEN	\$50146:	
2407	004714						
2408	004714						
2409	004714						
2410	004714	005037	000526				
2411	004720						
2412	004720	005037	000530				
2413	004724						
2414	004724	005037	000532				
2415	004730						
2416	004730	010437	000534				
2417	004734						
2418	004734	010537	000536				
2419	004740						
2420	004740	013737	000564	000542			
2421	004746						
2422	004746	005737	000574				
2423	004752	001003					
2424	004754						
2425	004754	004737	056622				
2426	004760						
2427	004760	000411					
2428	004762						
2429	004762						

2430	004762	023727	000574	177777		CMP	DATA05,#177777
2431	004770	001003				BNE	\$50150
2432	004772				CALL SETBUF		
2433	004772	004737	056664			JSR	PC,SETBUF
2434	004776				ELSE		
2435	004776	000402				BR	\$50151
2436	005000					\$50150:	
2437	005000				CALL ALTBUF		
2438	005000	004737	056730			JSR	PC,ALTBUF
2439	005004				ENDIF		
2440	005004					\$50151:	
2441	005004				ENDIF		
2442	005004					\$50147:	
2443	005004				LET BFADRS := DATA03		
2444	005004	013737	000570	000542		MOV	DATA03,BFADRS
2445	005012				IF DATA06 EQ #0 THEN		
2446	005012	005737	000576			TST	DATA06
2447	005016	001003				BNE	\$50152
2448	005020				CALL CLRBUF		
2449	005020	004737	056622			JSR	PC,CLRBUF
2450	005024				ELSE		
2451	005024	000411				BR	\$50153
2452	005026					\$50152:	
2453	005026				IF DATA06 EQ #177777 THEN		
2454	005026	023727	000576	177777		CMP	DATA06,#177777
2455	005034	001003				BNE	\$50154
2456	005036				CALL SETBUF		
2457	005036	004737	056664			JSR	PC,SETBUF
2458	005042				ELSE		
2459	005042	000402				BR	\$50155
2460	005044					\$50154:	
2461	005044				CALL ALTBUF		
2462	005044	004737	056730			JSR	PC,ALTBUF
2463	005050				ENDIF		
2464	005050					\$50155:	
2465	005050				ENDIF		
2466	005050					\$50153:	
2467	005050				LET BF.EXP := @DATA03		
2468	005050	017737	173514	000540		MOV	@DATA03,BF.EXP
2469							
2470					::+		
2471					:: DO THE CIS INSTRUCTION		
2472					::-		
2473							
2474	005056				INLINE <MOVRC>		
2475	005056	076031				MOVRC	
2476							
2477					::+		
2478					:: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS		
2479					::-		
2480							
2481	005060				LET CC := PSW		
2482	005060	013737	177766	000520		MOV	PSW,CC
2483	005066				LET CC := CC CLR.BY #177760		
2484	005066	042737	177760	000520		BIC	#177760,CC
2485	005074				IF CC NE CC.EXP THEN		

2486	005074	023737	000520	000522		CMP	CC,CC.EXP
2487	005102	001402				BEQ	\$50156
2488	005104				CALL ERROR		
2489	005104	004737	060044			JSR	PC,ERROR
2490	005110				ENDIF		\$50156:
2491	005110						
2492	005110				IF R0 NE R0.EXP THEN		
2493	005110	020037	000524			CMP	R0,R0.EXP
2494	005114	001402				BEQ	\$50157
2495	005116				CALL ERROR		
2496	005116	004737	060044			JSR	PC,ERROR
2497	005122				ENDIF		\$50157:
2498	005122						
2499	005122				IF R1 NE R1.EXP THEN		
2500	005122	020137	000526			CMP	R1,R1.EXP
2501	005126	001402				BEQ	\$50160
2502	005130				CALL ERROR		
2503	005130	004737	060044			JSR	PC,ERROR
2504	005134				ENDIF		\$50160:
2505	005134						
2506	005134				IF R2 NE R2.EXP THEN		
2507	005134	020237	000530			CMP	R2,R2.EXP
2508	005140	001402				BEQ	\$50161
2509	005142				CALL ERROR		
2510	005142	004737	060044			JSR	PC,ERROR
2511	005146				ENDIF		\$50161:
2512	005146						
2513	005146				IF R3 NE R3.EXP THEN		
2514	005146	020337	000532			CMP	R3,R3.EXP
2515	005152	001402				BEQ	\$50162
2516	005154				CALL ERROR		
2517	005154	004737	060044			JSR	PC,ERROR
2518	005160				ENDIF		\$50162:
2519	005160						
2520	005160				IF R4 NE R4.EXP THEN		
2521	005160	020437	000534			CMP	R4,R4.EXP
2522	005164	001402				BEQ	\$50163
2523	005166				CALL ERROR		
2524	005166	004737	060044			JSR	PC,ERROR
2525	005172				ENDIF		\$50163:
2526	005172						
2527	005172				IF R5 NE R5.EXP THEN		
2528	005172	020537	000536			CMP	R5,R5.EXP
2529	005176	001402				BEQ	\$50164
2530	005200				CALL ERROR		
2531	005200	004737	060044			JSR	PC,ERROR
2532	005204				ENDIF		\$50164:
2533	005204						
2534							
2535					:+		
2536					: CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS		
2537					:-		
2538							
2539	005204				LET ERRFLG := #0		
2540	005204	005037	000456			CLR	ERRFLG
2541	005210				LET R1 := DATA03		

2542	005210	013701	000570			MOV	DATA03,R1
2543	005214				IF DATA00 HIS DATA02 THEN		
2544	005214	023737	000562	000566		CMP	DATA00,DATA02
2545	005222	103424				BLO	\$50165
2546	005224				LET R3 :- DATA02		
2547	005224	013703	000566			MOV	DATA02,R3
2548	005230				LET R0 := DATA01 + DATA00 - DATA02		
2549	005230	013700	000564			MOV	DATA01,R0
2550	005234	063700	000562			ADD	DATA00,R0
2551	005240	163700	000566			SUB	DATA02,R0
2552	005244				WHILE R3 NE #0 AND ERRFLG EQ #0 DO		
2553	005244					\$50166:	
2554	005244	005703				TST	R3
2555	005246	001411				BEQ	\$50167
2556	005250	005737	000456			TST	ERRFLG
2557	005254	001006				BNE	\$50167
2558	005256				IFB (R0)+ NE (R1)+ THEN		
2559	005256	122021				CMPB	(R0)+,(R1)+
2560	005260	001402				BEQ	\$50170
2561	005262				CALL ERROR		
2562	005262	004737	060044			ISP	PC,ERROR
2563	005266				ENDIF		
2564	005266				LET R3 := R3 - #1	\$50170:	
2565	005266					DEC	R3
2566	005266	005303			ENDDO	BR	\$50166
2567	005270					\$50167:	
2568	005270	000765			ELSE	BR	\$50171
2569	005272					\$50165:	
2570	005272				REPEAT	\$50172:	
2571	005272	000455			IFB (R1)+ NE DATA04 THEN	CMPB	(R1)+,DATA04
2572	005274					BEQ	\$50173
2573	005274				CALL ERROR		
2574	005274				ENDIF	JSR	PC,ERROR
2575	005274				LET TEMPO := DATA03 + DATA02 - DATA00	\$50173:	
2576	005274	122137	000577			MOV	DATA03,TEMPO
2577	005300	001402				ADD	DATA02,TEMPO
2578	005302				UNTIL R1 EQ TEMPO OR ERRFLG NE #0	SUB	DATA00,TEMPO
2579	005302	004737	060044			CMP	R1,TEMPO
2580	005306					BEQ	\$50174
2581	005306					TST	ERRFLG
2582	005306					BEQ	\$50172
2583	005306	013737	000570	000476		\$50174:	
2584	005314	063737	000566	000476		MOV	DATA01,R0
2585	005322	163737	000562	000476		ADD	DATA03,R1
2586	005330					SUB	DATA02,R1
2587	005330	020137	000476			SUB	DATA00,R1
2588	005334	001403					
2589	005336	005737	000456				
2590	005342	001754					
2591	005344				LET R0 :- DATA01		
2592	005344						
2593	005344	013700	000564		LET R1 := DATA03 + DATA02 - DATA00		
2594	005350						
2595	005350	013701	000570				
2596	005354	063701	000566				
2597	005360	163701	000562				

2598	005364				LET TEMPO := DATA03 + DATA02		
2599	005364	013737	000570	000476		MOV	DATA03,TEMPO
2600	005372	063737	000566	000476		ADD	DATA02,TEMPO
2601	005400				WHILE R1 NE TEMPO AND ERRFLG NE #0 DO		
2602	005400					\$50175:	
2603	005400	020137	000476			CMP	R1,TEMPO
2604	005404	001410				BEQ	\$50176
2605	005406	005737	000456			TST	ERRFLG
2606	005412	001405				BEQ	\$50176
2607	005414				IFB (R0)+ NE (R1)+ THEN		
2608	005414	122021				CMPB	(R0)+,(R1)+
2609	005416	001402				BEQ	\$50177
2610	005420				CALL ERROR		
2611	005420	004737	060044			JSR	PC,ERROR
2612	005424				ENDIF		
2613	005424					\$50177:	
2614	005424				ENDDO		
2615	005424	000765				BR	\$50175
2616	005426					\$50176:	
2617	005426				ENDIF		
2618	005426					\$50171:	
2619	005426				IF DATA02 LO #BUFLEN THEN		
2620	005426	023727	000566	000100		CMP	DATA02,#BUFLEN
2621	005434	103025				BHIS	\$50200
2622	005436				LET R0 := DATA03 + DATA02		
2623	005436	013700	000570			MOV	DATA03,R0
2624	005442	063700	000566			ADD	DATA02,R0
2625	005446				REPEAT		
2626	005446					\$50201:	
2627	005446				IFB (R0)+ NE BF.EXP THEN		
2628	005446	122037	000540			CMPB	(R0)+,BF.EXP
2629	005452	001402				BEQ	\$50202
2630	005454				CALL ERROR		
2631	005454	004737	060044			JSR	PC,ERROR
2632	005460				ENDIF		
2633	005460					\$50202:	
2634	005460				LET TEMPO := DATA03 + #BUFLEN		
2635	005460	013737	000570	000476		MOV	DATA03,TEMPO
2636	005466	062737	000100	000476		ADD	#BUFLEN,TEMPO
2637	005474				UNTIL R0 EQ TEMPO OR ERRFLG NE #0		
2638	005474	020037	000476			CMP	R0,TEMPO
2639	005500	001403				BEQ	\$50203
2640	005502	005737	000456			TST	ERRFLG
2641	005506	001757				BEQ	\$50201
2642	005510					\$50203:	
2643	005510				ENDIF		
2644	005510					\$50200:	
2645	005510				LET TSTCAS := TSTCAS + #16		
2646	005510	062737	000016	000544		ADD	#16,TSTCAS
2647	005516				LET COUNT := COUNT - #1		
2648	005516	005337	000470			DEC	COUNT
2649					:UNTIL COUNT EQ #0		
2650	005522				IF COUNT NE #0 THEN		
2651	005522	005737	000470			TST	COUNT
2652	005526	001402				BEQ	\$50204
2653	005530				INLINE <JMP \$5031>		

```
2654 005530 000137 004534
2655 005534
2656 005534
2657
2658
2659
2660
2661
2662
2663 005534
2664 005534 012737 063072 000542
2665 005542
2666 005542 004737 056664
2667 005546
2668 005546 012737 063172 000542
2669 005554
2670 005554 004737 056622
2671 005560
2672 005560 012700 125252
2673 005564
2674 005564 012701 125252
2675 005570
2676 005570 012702 125252
2677 005574
2678 005574 012703 125252
2679 005600
2680 005600 012704 125252
2681 005604
2682 005604 012705 125252
2683 005610
2684 005610 005037 000456
2685 005614
2686 005614 076131
2687 005616
2688 005616 006112
2689 005620
2690 005620 006116
2691 005622
2692 005622 000040
2693 005624
2694 005624 103003
2695 005626
2696 005626 001002
2697 005630
2698 005630 004737 060044
2699 005634
2700 005634
2701 005634
2702 005634
2703 005634
2704 005634 020027 125252
2705 005640 001402
2706 005642
2707 005642 004737 060044
2708 005646
2709 005646
```

```
MOVRC
ENDIF
:
: DO ONE INLINE CASE - MOVRCI
:
LET BFADRS := #BUFF1
CALL SETBUF
LET BFADRS := #BUFF2
CALL CLRBUF
LET R0 := #125252
LET R1 := #125252
LET R2 := #125252
LET R3 := #125252
LET R4 := #125252
LET R5 := #125252
LET ERRFLG := #0
INLINE <MOVRCI>
INLINE <.WORD TMVRCI> ;SRC.DSCR.PTR
INLINE <.WORD TMVRCI+4> ;DST.DSCR.PTR
INLINE <.WORD 40> ;FILL CHARACTER
IFCOND CS THEN
    IFCOND EQ THEN
        CALL ERROR
    ENDIF
ENDIF
IF R0 NE #125252 THEN
    CALL ERROR
ENDIF
```

```
JMP $5031
$50204:
MOV #BUFF1,BFADRS
JSR PC,SETBUF
MOV #BUFF2,BFADRS
JSR PC,CLRBUF
MOV #125252,R0
MOV #125252,R1
MOV #125252,R2
MOV #125252,R3
MOV #125252,R4
MOV #125252,R5
CLR ERRFLG
MOVRCI
.WORD TMVRCI
.WORD TMVRCI+4
.WORD 40
BCC $50205
BNE $50206
JSR PC,ERROR
$50206:
$50205:
CMP R0,#125252
BEQ $50207
JSR PC,ERROR
$50207:
```

2710	005646			IF R1 NE #125252 THEN	
2711	005646	020127	125252		CMP R1,#125252
2712	005652	001402			BEQ \$50210
2713	005654			CALL ERROR	
2714	005654	004737	060044		JSR PC,ERROR
2715	005660			ENDIF	
2716	005660				\$50210:
2717	005660			IF R2 NE #125252 THEN	
2718	005660	020227	125252		CMP R2,#125252
2719	005664	001402			BEQ \$50211
2720	005666			CALL ERROR	
2721	005666	004737	060044		JSR PC,ERROR
2722	005672			ENDIF	
2723	005672				\$50211:
2724	005672			IF R3 NE #125252 THEN	
2725	005672	020327	125252		CMP R3,#125252
2726	005676	001402			BEQ \$50212
2727	005700			CALL ERROR	
2728	005700	004737	060044		JSR PC,ERROR
2729	005704			ENDIF	
2730	005704				\$50212:
2731	005704			IF R4 NE #125252 THEN	
2732	005704	020427	125252		CMP R4,#125252
2733	005710	001402			BEQ \$50213
2734	005712			CALL ERROR	
2735	005712	004737	060044		JSR PC,ERROR
2736	005716			ENDIF	
2737	005716				\$50213:
2738	005716			IF R5 NE #125252 THEN	
2739	005716	020527	125252		CMP R5,#125252
2740	005722	001402			BEQ \$50214
2741	005724			CALL ERROR	
2742	005724	004737	060044		JSR PC,ERROR
2743	005730			ENDIF	
2744	005730				\$50214:
2745	005730			IF BUFF2 NE #20040 THEN	
2746	005730	023727	063172 020040		CMP BUFF2,#20040
2747	005736	001402			BEQ \$50215
2748	005740			CALL ERROR	
2749	005740	004737	060044		JSR PC,ERROR
2750	005744			ENDIF	
2751	005744				\$50215:
2752	005744			IF BUFF2+2 NE #177777 THEN	
2753	005744	023727	063174 177777		CMP BUFF2+2,#177777
2754	005752	001402			BEQ \$50216
2755	005754			CALL ERROR	
2756	005754	004737	060044		JSR PC,ERROR
2757	005760			ENDIF	
2758	005760				\$50216:
2759	005760			LET ERRFLG := #0	
2760	005760	005037	000456		CLR ERRFLG
2761	005764			LET R0 := #BUFF2+4	
2762	005764	012700	063176		MOV #BUFF2+4,R0
2763	005770			REPEAT	
2764	005770				\$50217:
2765	005770			IF (R0)+ NE #0 THEN	

2766 005770 005720
 2767 005772 001402
 2768 005774
 2769 005774 004737 060044
 2770 006000
 2771 006000
 2772 006000
 2773 006000 020027 063270
 2774 006004 001403
 2775 006006 005737 000456
 2776 006012 001766
 2777 006014
 2778 006014
 2779 006014 000137 006122
 2780
 2781
 2782
 2783
 2784
 2785 006020 000004
 2786
 2787 006022 000060
 2788 006024 063172
 2789 006026 000025
 2790 006030 063072
 2791 006032 000040
 2792 006034 000000
 2793 006036 000001
 2794
 2795 006040 000017
 2796 006042 063172
 2797 006044 000050
 2798 006046 063072
 2799 006050 000100
 2800 006052 000001
 2801 006054 000000
 2802
 2803 006056 000030
 2804 006060 063072
 2805 006062 000012
 2806 006064 063172
 2807 006066 000077
 2808 006070 000000
 2809 006072 177777
 2810
 2811 006074 000022
 2812 006076 063072
 2813 006100 000037
 2814 006102 063072
 2815 006104 000020
 2816 006106 177777
 2817 006110 000001
 2818
 2819 006112 000002
 2820 006114 063072
 2821 006116 000004

CALL ERROR
 ENDIF
 UNTIL R0 EQ #BUFF2+BUFLen-2 OR ERRFLG NE #0

INLINE <JMP XMOVTC>

:+
 ;TABLE OF TEST CASES FOR MOVRC
 :-

TMOVRC: .WORD 4 ;#OF TEST CASES FOR MOVRC
 .WORD 60 ;SRC.LEN
 .WORD BUFF2 ;SRC.ADR
 .WORD 25 ;DST.LEN
 .WORD BUFF1 ;DST.ADR
 .WORD 40 ;FILL CHAR.
 .WORD 0 ;SRC.BUFF.STATE
 .WORD 1 ;DST.BUFF.STATE
 ;TEST CASE NUMBER 2
 .WORD 17
 .WORD BUFF2
 .WORD 50
 .WORD BUFF1
 .WORD 100
 .WORD 1
 .WORD 0
 ;TEST CASE NUMBER 3
 .WORD 30
 .WORD BUFF1
 .WORD 12
 .WORD BUFF2
 .WORD 77
 .WORD 0
 .WORD -1
 ;TEST CASE NUMBER 4
 .WORD 22
 .WORD BUFF1
 .WORD 37
 .WORD BUFF1
 .WORD 20
 .WORD -1
 .WORD 1
 ;TEST CASE FOR IN-LINE
 TMVRCI: .WORD 2
 .WORD BUFF1
 .WORD 4

TST (R0)+
 BEQ \$50220
 JSR PC,ERROR
 \$50220:
 CMP R0,#BUFF2+BUFLen-2
 BEQ \$50221
 TST ERRFLG
 BEQ \$50217
 \$50221:
 JMP XMOVTC

CJKDHA0 REF11-B CIS DIAGNOSTIC MACY11 30A(1052) 14-JUL-81 14:19 J 5 PAGE 61
CJKDHA.P11 29-JUN-81 16:54 MOVRC

SEQ 0061

2822 006120 063172

.WORD BUFF2


```

2823      .SBTTL MOVTC
2824      :*****
2825      :
2826      :   MOVTC TEST
2827      :
2828      :*****
2829      :   INLINE <XMOVTC:>
2830      :
2831      :   INLINE <SCOPE>
2832      :
2833      :   LET $TIMES := #20           ;ITERATION COUNT
2834      :   MOV #20,$TIMES
2835      :   LET COUNT := TMOVTC      ;GET TEST COUNT
2836      :   MOV TMOVTC,COUNT
2837      :   LET TSTCAS := #TMOVTC+2 ;POINT TO FIRST TEST CASE
2838      :   MOV #TMOVTC+2,TSTCAS
2839      :   ;REPEAT
2840      :   INLINE <$5032:>
2841      :
2842      :
2843      :   ;+
2844      :   ; GET TEST DATA FROM TABLE
2845      :   ;-
2846      :
2847      :   LET R0 := TSTCAS
2848      :   MOV TSTCAS,R0
2849      :   LET DATA00 := (R0)+      ;SRC.LEN
2850      :   MOV (R0)+,DATA00
2851      :   LET DATA01 := (R0)+      ;SRC.ADR
2852      :   MOV (R0)+,DATA01
2853      :   LET DATA02 := (R0)+      ;DST.LEN
2854      :   MOV (R0)+,DATA02
2855      :   LET DATA03 := (R0)+      ;DST.ADR
2856      :   MOV (R0)+,DATA03
2857      :   LET DATA04 := (R0)+      ;FILL CHAR.
2858      :   MOV (R0)+,DATA04
2859      :   LET DATA05 := (R0)+      ;SRC.BUFF.STATE
2860      :   MOV (R0)+,DATA05
2861      :   LET DATA06 := (R0)        ;DST.BUFF.STATE
2862      :   MOV (R0),DATA06
2863      :
2864      :   ;+
2865      :   ; SET UP GPR'S,
2866      :   ; EXPECTED GPR'S AND CONDITION CODES,
2867      :   ; AND BUFFERS.
2868      :   ;-
2869      :
2870      :   LET R0 := DATA00
2871      :   MOV DATA00,R0
2872      :   LET R1 := DATA01
2873      :   MOV DATA01,R1
2874      :   LET R2 := DATA02
2875      :   MOV DATA02,R2
2876      :   LET R3 := DATA03
2877      :   MOV DATA03,R3
2878      :   LET R4 := DATA04
    
```

2879	006226	013704	000572			MOV	DATA04,R4
2880	006232				LET R5 := #TRANS	MOV	#TRANS,R5
2881	006232	012705	063272		INLINE <CMP R0,R2>	CMP	R0,R2
2882	006236				LET CC.EXP := PSW	MOV	PSW,CC.EXP
2883	006236	020002			LET CC.EXP := CC.EXP CLR.BY #177760	BIC	#177760,CC.EXP
2884	006240				IF #CBIT SETIN CC.EXP OR #ZBIT SETIN CC.EXP THEN	BIT	#CBIT,CC.EXP
2885	006240	013737	177776	000522		BNE	\$50222
2886	006246					BIT	#ZBIT,CC.EXP
2887	006246	042737	177760	000522		BEQ	\$50223
2888	006254						\$50222:
2889	006254	032737	000001	000522	LET R0.EXP := #0	CLR	R0.EXP
2890	006262	001004			ELSE	BR	\$50224
2891	006264	032737	000004	000522	LET R0.EXP := R0 - R2		\$50223:
2892	006272	001403			ENDIF	MOV	R0,R0.EXP
2893	006274				LET R1.EXP := #0	SUB	R2,R0.EXP
2894	006274	005037	000524		LET R2.EXP := #0		\$50224:
2895	006274	000404			LET R3.EXP := #0	CLR	R1.EXP
2896	006300				LET R4.EXP := #0	CLR	R2.EXP
2897	006300				LET R5.EXP := #0	CLR	R3.EXP
2898	006302				LET R4.EXP := R4	MOV	R4,R4.EXP
2899	006302	010037	000524		LET R5.EXP := R5	MOV	R5,R5.EXP
2900	006306	160237	000524		LET BFADRS := DATA01	MOV	DATA01,BFADRS
2901	006312				IF DATA05 EQ #0 THEN	TST	DATA05
2902	006312				CALL CLRBUF	BNE	\$50225
2903	006312				ELSE	JSR	PC,CLRBUF
2904	006312	005037	000526		IF DATA05 EQ #177777 THEN	BR	\$50226
2905	006312	005037	000530		CALL SETBUF		\$50225:
2906	006316				ELSE	CMP	DATA05,#177777
2907	006316	005037	000532		CALL ALTBUF	BNE	\$50227
2908	006322					JSR	PC,SETBUF
2909	006322	005037	000532			BR	\$50230
2910	006326						\$50227:
2911	006326	010437	000534			JSR	PC,ALTBUF
2912	006332						
2913	006332	010537	000536				
2914	006336						
2915	006336	013737	000564	000542			
2916	006344						
2917	006344	005737	000574				
2918	006350	001003					
2919	006352	004737	056622				
2920	006352	000411					
2921	006356						
2922	006356						
2923	006360						
2924	006360	023727	000574	177777			
2925	006366	001003					
2926	006370						
2927	006370	004737	056664				
2928	006374						
2929	006374	000402					
2930	006376						
2931	006376						
2932	006376	004737	056730				
2933	006376						
2934	006402						

2991					:+			
2992					: DO THE CIS INSTRUCTION			
2993					:-			
2994								
2995	006530				INLINE <MOVTC>			
2996	006530	076032					MOVTC	
2997								
2998					:+			
2999					: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS			
3000					:-			
3001								
3002	006532				LET CC := PSW			
3003	006532	013737	177776	000520			MOV	PSW,CC
3004	006540				LET CC := CC CLR.BY #177760			
3005	006540	042737	177760	000520			BIC	#177760,CC
3006	006546				IF CC NE CC.EXP THEN			
3007	006546	023737	000520	000522			CMP	CC,CC.EXP
3008	006554	001402					BEQ	\$50240
3009	006556				CALL ERROR			
3010	006556	004737	060044				JSR	PC,ERROR
3011	006562				ENDIF			
3012	006562						\$50240:	
3013	006562				IF R0 NE R0.EXP THEN			
3014	006562	020037	000524				CMP	R0,R0.EXP
3015	006566	001402					BEQ	\$50241
3016	006570				CALL ERROR			
3017	006570	004737	060044				JSR	PC,ERROR
3018	006574				ENDIF			
3019	006574						\$50241:	
3020	006574				IF R1 NE R1.EXP THEN			
3021	006574	020137	000526				CMP	R1,R1.EXP
3022	006600	001402					BEQ	\$50242
3023	006602				CALL ERROR			
3024	006602	004737	060044				JSR	PC,ERROR
3025	006606				ENDIF			
3026	006606						\$50242:	
3027	006606				IF R2 NE R2.EXP THEN			
3028	006606	020237	000530				CMP	R2,R2.EXP
3029	006612	001402					BEQ	\$50243
3030	006614				CALL ERROR			
3031	006614	004737	060044				JSR	PC,ERROR
3032	006620				ENDIF			
3033	006620						\$50243:	
3034	006620				IF R3 NE R3.EXP THEN			
3035	006620	020337	000532				CMP	R3,R3.EXP
3036	006624	001402					BEQ	\$50244
3037	006626				CALL ERROR			
3038	006626	004737	060044				JSR	PC,ERROR
3039	006632				ENDIF			
3040	006632						\$50244:	
3041	006632				IF R4 NE R4.EXP THEN			
3042	006632	020437	000534				CMP	R4,R4.EXP
3043	006636	001402					BEQ	\$50245
3044	006640				CALL ERROR			
3045	006640	004737	060044				JSR	PC,ERROR
3046	006644				ENDIF			

3047	006644					\$50245:
3048	006644			IF R5 NE R5.EXP THEN		
3049	006644	020537	000536			CMP R5,R5.EXP
3050	006650	001402				BEQ \$50246
3051	006652			CALL ERROR		
3052	006652	004737	060044			JSR PC,ERROR
3053	006656			ENDIF		
3054	006656					\$50246:
3055						
3056				:+		
3057				: CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS		
3058				:-		
3059						
3060	006656			LET R0 := DATA03		
3061	006656	013700	000570			MOV DATA03,R0
3062	006662			LET R2 :B= @DATA01		
3063	006662	117702	171676			MOVB @DATA01,R2
3064	006666			LET R2 := R2 CLR.BY #177400		
3065	006666	042702	177400			BIC #177400,R2
3066	006672			LET R1 :B= TRANS(R2)		
3067	006672	116201	063272			MOVB TRANS(R2),R1
3068	006676			LET ERRFLG := #0		
3069	006676	005037	000456			CLR ERRFLG
3070	006702			IF DATA00 HIS DATA02 THEN		
3071	006702	023737	000562 000566			CMP DATA00,DATA02
3072	006710	103416				BLO \$50247
3073	006712			LET R3 := DATA02		
3074	006712	013703	000566			MOV DATA02,R3
3075	006716			WHILE R3 NE #0 AND ERRFLG EQ #0 DO		
3076	006716					\$50250:
3077	006716	005703				TST R3
3078	006720	001411				BEQ \$50251
3079	006722	005737 000456				TST ERRFLG
3080	006726	001006				BNE \$50251
3081	006730			IFB (R0)+ NE R1 THEN		
3082	006730	122001				CMPB (R0)+,R1
3083	006732	001402				BEQ \$50252
3084	006734			CALL ERROR		
3085	006734	004737 060044				JSR PC,ERROR
3086	006740			ENDIF		
3087	006740			LET R3 := R3 - #1		\$50252:
3088	006740					DEC R3
3089	006740	005303				BR \$50250
3090	006742			ENDDO		\$50251:
3091	006742	000765				BR \$50253
3092	006744			ELSE		\$50247:
3093	006744					BR \$50253
3094	006744	000440				\$50247:
3095	006746			LET R3 := DATA00		MOV DATA00,R3
3096	006746					
3097	006746	013703 000562				
3098	006752			WHILE R3 NE #0 AND ERRFLG EQ #0 DO		
3099	006752					\$50254:
3100	006752	005703				TST R3
3101	006754	001411				BEQ \$50255
3102	006756	005737 000456				TST ERRFLG

3103	006762	001006				BNE	\$50255	
3104	006764			IFB (R0)+ NE R1 THEN		CMPB	(R0)+,R1	
3105	006764	122001				BEQ	\$50256	
3106	006766	001402		CALL ERROR		JSR	PC,ERROR	
3107	006770							
3108	006770	004737	060044					
3109				ENDIF				
3110	006774						\$50256:	
3111	006774			LET R3 := R3 - #1				
3112	006774						DEC	R3
3113	006774	005303		ENDDO			BR	\$50254
3114	006776						\$50255:	
3115	006776	000765		LET R0 := DATA03 + DATA00		MOV	DATA03,R0	
3116	007000					ADD	DATA00,R0	
3117	007000			LET R3 := DATA02 - DATA00		MOV	DATA02,R3	
3118	007000	013700	000570			SUB	DATA00,R3	
3119	007004	063700	000562	REPEAT			\$50257:	
3120	007010			IFB (R0)+ NE DATA04 THEN		CMPB	(R0)+,DATA04	
3121	007010	013703	000566			BEQ	\$50260	
3122	007014	163703	000562	CALL ERROR		JSR	PC,ERROR	
3123	007020			ENDIF			\$50260:	
3124	007020			LET R3 := R3 - #1		DEC	R3	
3125	007020			UNTIL R3 EQ #0 OR ERRFLG NE #0		TST	R3	
3126	007020	122037	000572			BEQ	\$50261	
3127	007024	001402				TST	ERRFLG	
3128	007026					BEQ	\$50257	
3129	007026	004737	060044	ENDIF			\$50261:	
3130	007032			IF DATA02 LO #BUFLEN THEN			\$50253:	
3131	007032			LET TEMPO := DATA03 + #BUFLEN - #1		CMP	DATA02,#BUFLEN	
3132	007032					BHIS	\$50262	
3133	007032	005303		LET R0 := DATA03 + DATA02		MOV	DATA03,TEMPO	
3134	007034			REPEAT		ADD	#BUFLEN,TEMPO	
3135	007034	005703		IFB (R0)+ NE BF.EXP THEN		DEC	TEMPO	
3136	007036	001403				MOV	DATA03,R0	
3137	007040	005737	000456	CALL ERROR		ADD	DATA02,R0	
3138	007044	001765					\$50263:	
3139	007046					CMPB	(R0)+,BF.EXP	
3140	007046					BEQ	\$50264	
3141	007046					JSR	PC,ERROR	
3142	007046							
3143	007046	023727	000566					
3144	007054	103027						
3145	007056							
3146	007056	013737	000570					
3147	007064	062737	000100					
3148	007072	005337	000476					
3149	007076							
3150	007076	013700	000570					
3151	007102	063700	000566					
3152	007106							
3153	007106							
3154	007106							
3155	007106	122037	000540					
3156	007112	001402						
3157	007114							
3158	007114	004737	060044					

3159	007120				ENDIF	
3160	007120				UNTIL R0 EQ TEMPO OR ERRFLG NE #0	\$50264:
3161	007120					CMP R0,TEMPO
3162	007120	020037	000476			BEQ \$50265
3163	007124	001403				TST ERRFLG
3164	007126	005737	000456			BEQ \$50263
3165	007132	001765				\$50265:
3166	007134				ENDIF	\$50262:
3167	007134				LET TSTCAS := TSTCAS + #16	ADD #16,TSTCAS
3168	007134				LET COUNT := COUNT - #1	DEC COUNT
3169	007134					
3170	007134	062737	000016	000544	:UNTIL COUNT EQ #C	TST COUNT
3171	007142				IF COUNT NE #0 THEN	BEQ \$50266
3172	007142	005337	000470			JMP \$5032
3173					ENDIF	\$50266:
3174	007146				INLINE <JMP \$5032>	
3175	007146	005737	000470			
3176	007152	001402				
3177	007154					
3178	007154	000137	006146			
3179	007160					
3180	007160					
3181						
3182						
3183					:+	
3184					: DO ONE INLINE CASE - MOVTCI	
3185					:-	
3186						
3187	007160				LET BFADRS := #BUFF1	MOV #BUFF1,BFADRS
3188	007160	012737	063072	000542	CALL CLRBUF	JSR PC,CLRBUF
3189	007166					
3190	007166	004737	056622		LET BFADRS := #BUFF2	MOV #BUFF2,BFADRS
3191	007172					
3192	007172	012737	063172	000542	CALL SETBUF	JSR PC,SETBUF
3193	007200					
3194	007200	004737	056664		LET R0 := #125252	MOV #125252,R0
3195	007204				LET R1 := #125252	MOV #125252,R1
3196	007204	012700	125252		LET R2 := #125252	MOV #125252,R2
3197	007210				LET R3 := #125252	MOV #125252,R3
3198	007210	012701	125252		LET R4 := #125252	MOV #125252,R4
3199	007214				LET R5 := #125252	MOV #125252,R5
3200	007214	012702	125252		LET TRANS := #70707	MOV #70707,TRANS
3201	007220				INLINE <MOVTCI>	MOVTCI
3202	007220	012703	125252		INLINE <.WORD TMVTCI>	.WORD TMVTCI
3203	007224				INLINE <.WORD TMVTCI+4>	.WORD TMVTCI+4
3204	007224	012704	125252			
3205	007230					
3206	007230	012705	125252			
3207	007234					
3208	007234	012737	070707	063272		
3209	007242					
3210	007242	076132				
3211	007244					
3212	007244	007550				
3213	007246					
3214	007246	007554				

3215	007250			INLINE <.WORD 40>		.WORD	40
3216	007250	000040					
3217	007252			INLINE <.WORD TRANS>		.WORD	TRANS
3218	007252	063272					
3219	007254			IFCOND CS THEN			
3220	007254	103003				BCC	\$50267
3221	007256			IFCOND EQ THEN			
3222	007256	001002				BNE	\$50270
3223	007260			CALL ERROR			
3224	007260	004737	060044			JSR	PC,ERROR
3225	007264			ENDIF			
3226	007264					\$50270:	
3227	007264			ENDIF			
3228	007264					\$50267:	
3229	007264			IF R0 NE #125252 THEN			
3230	007264	020027	125252			CMP	R0,#125252
3231	007270	001402				BEQ	\$50271
3232	007272			CALL ERROR			
3233	007272	004737	060044			JSR	PC,ERROR
3234	007276			ENDIF			
3235	007276					\$50271:	
3236	007276			IF R1 NE #125252 THEN			
3237	007276	020127	125252			CMP	R1,#125252
3238	007302	001402				BEQ	\$50272
3239	007304			CALL ERROR			
3240	007304	004737	060044			JSR	PC,ERROR
3241	007310			ENDIF			
3242	007310					\$50272:	
3243	007310			IF R2 NE #125252 THEN			
3244	007310	020227	125252			CMP	R2,#125252
3245	007314	001402				BEQ	\$50273
3246	007316			CALL ERROR			
3247	007316	004737	060044			JSR	PC,ERROR
3248	007322			ENDIF			
3249	007322					\$50273:	
3250	007322			IF R3 NE #125252 THEN			
3251	007322	020327	125252			CMP	R3,#125252
3252	007326	001402				BEQ	\$50274
3253	007330			CALL ERROR			
3254	007330	004737	060044			JSR	PC,ERROR
3255	007334			ENDIF			
3256	007334					\$50274:	
3257	007334			IF R4 NE #125252 THEN			
3258	007334	020427	125252			CMP	R4,#125252
3259	007340	001402				BEQ	\$50275
3260	007342			CALL ERROR			
3261	007342	004737	060044			JSR	PC,ERROR
3262	007346			ENDIF			
3263	007346					\$50275:	
3264	007346			IF R5 NE #125252 THEN			
3265	007346	020527	125252			CMP	R5,#125252
3266	007352	001402				BEQ	\$50276
3267	007354			CALL ERROR			
3268	007354	004737	060044			JSR	PC,ERROR
3269	007360			ENDIF			
3270	007360					\$50276:	


```

3271 007360          IF BUFF2 NE #143707 THEN
3272 007360 023727 063172 143707          CMP     BUFF2,#143707
3273 007366 001402          BEQ     $50277
3274 007370          CALL ERROR
3275 007370 004737 060044          JSR     PC,ERROR
3276 007374          ENDF
3277 007374          $50277:
3278 007374          IF BUFF2+2 NE #20040 THEN
3279 007374 023727 063174 020040          CMP     BUFF2+2,#20040
3280 007402 001402          BEQ     $50300
3281 007404          CALL ERROR
3282 007404 004737 060044          JSR     PC,ERROR
3283 007410          ENDF
3284 007410          $50300:
3285 007410          LET R0 :- #BUFF2 + #4
3286 007410 012700 063172          MOV     #BUFF2,R0
3287 007414 062700 000004          ADD     #4,R0
3288 007420          LET ERRFLG :- #0
3289 007420 005037 000456          CLR     ERRFLG
3290 007424          REPEAT
3291 007424          $50301:
3292 007424          IF (R0)+ NE #177777 THEN
3293 007424 022027 177777          CMP     (R0)+,#177777
3294 007430 001402          BEQ     $50302
3295 007432          CALL ERROR
3296 007432 004737 060044          JSR     PC,ERROR
3297 007436          ENDF
3298 007436          $50302:
3299 007436          UNTIL R0 EQ #BUFF2+BUFLN OR ERRFLG NE #0
3300 007436 020027 063272          CMP     R0,#BUFF2+BUFLN
3301 007442 001403          BEQ     $50303
3302 007444 005737 000456          TST     ERRFLG
3303 007450 001765          BEQ     $50301
3304 007452          $50303:
3305 007452          INLINE <JMP XLOCC>
3306 007452 000737 007560          JMP     XLOCC
3307
3308
3309      ;+
3310      ; TABLE OF TEST CASES FOR MOVTC
3311      ; -
3312 007456 000004          TMOVTC: .WORD 4          ;#OF TEST CASES FOR MOVTC
3313
3314 007460 000061          .WORD 61          ;SRC.LEN
3315 007462 063172          .WORD BUFF2        ;SRC.ADR
3316 007464 000027          .WORD 27          ;DST.LEN
3317 007466 063072          .WORD BUFF1        ;DST.ADR
3318 007470 000040          .WORD 40          ;FILL CHARACTER
3319 007472 000000          .WORD 0          ;SRC.BUFF.STATE
3320 007474 000001          .WORD 1          ;DST.BUFF.STATE
3321
3322 007476 000055          .WORD 55          ;TEST CASE NUMBER 2
3323 007500 063172          .WORD BUFF2
3324 007502 000075          .WORD 75
3325 007504 063072          .WORD BUFF1
3326 007506 000100          .WORD 100
    
```

3327	007510	000001	.WORD	1	
3328	007512	000000	.WORD	0	
3329					
3330	007514	000032	.WORD	32	:TEST CASE NUMBER 3
3331	007516	063072	.WORD	BUFF1	
3332	007520	000031	.WORD	31	
3333	007522	063172	.WORD	BUFF2	
3334	007524	000073	.WORD	73	
3335	007526	000000	.WORD	0	
3336	007530	177777	.WORD	-1	
3337					
3338	007532	000005	.WORD	5	:TEST CASE NUMBER 4
3339	007534	063072	.WORD	BUFF1	
3340	007536	000026	.WORD	26	
3341	007540	063172	.WORD	BUFF2	
3342	007542	000020	.WORD	20	
3343	007544	177777	.WORD	-1	
3344	007546	000001	.WORD	1	
3345					
3346	007550	000002	.WORD	2	:TEST CASE FOR INLINE
3347	007552	063072	.WORD	BUFF1	
3348	007554	000004	.WORD	4	
3349	007556	063172	.WORD	BUFF2	

TMVTCI:

```

3350 .SBTTL LOCC
3351 :*****
3352 :
3353 : LOCC TEST
3354 :
3355 :*****
3356 007560 INLINE <XLOCC:> XLOCC:
3357 007560
3358 007560
3359 007560 000004 INLINE <SCOPE> SCOPE
3360 007562 LET $TIMES := #20 ;ITERATION COUNT
3361 007562 012737 000020 000404 MOV #20,$TIMES
3362 007570 LET COUNT := TLOCC ;GET TEST COUNT
3363 007570 013737 010412 000470 MOV TLOCC,COUNT
3364 007576 LET TSTCAS := #TLOCC+2 ;POINT TO FIRST TEST CASE
3365 007576 012737 010414 000544 MOV #TLOCC+2,TSTCAS
3366 ;REPEAT
3367 007604 INLINE <$5040:> $5040:
3368 007604
3369
3370
3371 :+
3372 : GET TEST DATA FROM TABLE
3373 :
3374 007604 013700 000544 LET R0 := TSTCAS
3375 007610
3376 007610 012037 000562 LET DATA00 := (R0)+ ;SRC.LEN MOV TSTCAS,R0
3377 007614 LET DATA01 := (R0)+ ;CHAR.SEARCHED FOR MOV (R0)+,DATA00
3378 007614 012037 000564 LET DATA02 := (R0) ;OFFSET MOV (R0)+,DATA01
3379 007620
3380 007620 011037 000566 MOV (R0),DATA02
3381
3382 :+
3383 : SET UP GPR'S,
3384 : EXPECTED GPR'S AND CONDITION CODES,
3385 : AND SOURCE BUFFER.
3386 :-
3387 007624 LET R0 := DATA00
3388 007624 013700 000562 MOV DATA00,R0
3389 007630 LET R1 := #BUFF1
3390 007630 012701 063072 MOV #BUFF1,R1
3391 007634 LET R4 := DATA01
3392 007634 013704 000564 MOV DATA01,R4
3393 007640 INLINE <BIT #BIT00,COUNT>
3394 007640 032737 000001 000470 BIT #BIT00,COUNT
3395 007646 IFCOND EQ THEN
3396 007646 001004 BNE $50304
3397 007650 LET R2 := #0
3398 007650 005002 CLR R2
3399 007652 LET R3 := #0
3400 007652 005003 CLR R3
3401 007654 LET R5 := #0
3402 007654 005005 CLR R5
3403 007656 ELSE
3404 007656 000406 BR $50305
3405 007660 $50304:
    
```

3406	007660				LET R2 := #177777		
3407	007660	012702	177777			MOV	#177777,R2
3408	007664				LET R3 := #177777		
3409	007664	012703	177777			MOV	#177777,R3
3410	007670				LET R5 := #177777		
3411	007670	012705	177777			MOV	#177777,R5
3412	007674				ENDIF		
3413	007674					\$50305:	
3414	007674				IF DATA02 HI DATA00 THEN		
3415	007674	023737	000566	000562		CMP	DATA02,DATA00
3416	007702	101411				BLOS	\$50306
3417	007704				LET R0.EXP := #0		
3418	007704	005037	000524			CLR	R0.EXP
3419	007710				LET R1.EXP := #BUFF1 + DATA00		
3420	007710	012737	063072	000526		MOV	#BUFF1,R1.EXP
3421	007716	063737	000562	000526		ADD	DATA00,R1.EXP
3422	007724				ELSE		
3423	007724	000414				BR	\$50307
3424	007726					\$50306:	
3425	007726				LET R0.EXP := DATA00 - DATA02		
3426	007726	013737	000562	000524		MOV	DATA00,R0.EXP
3427	007734	163737	000566	000524		SUB	DATA02,R0.EXP
3428	007742				LET R1.EXP := #BUFF1 + DATA02		
3429	007742	012737	063072	000526		MOV	#BUFF1,R1.EXP
3430	007750	063737	000566	000526		ADD	DATA02,R1.EXP
3431	007756				ENDIF		
3432	007756					\$50307:	
3433	007756				LET R2.EXP := R2		
3434	007756	010237	000530			MOV	R2,R2.EXP
3435	007762				LET R3.EXP := R3		
3436	007762	010337	000532			MOV	R3,R3.EXP
3437	007766				LET R4.EXP := R4		
3438	007766	010437	000534			MOV	R4,R4.EXP
3439	007772				LET R5.EXP := R5		
3440	007772	010537	000536			MOV	R5,R5.EXP
3441	007776				INLINE <TST R0.EXP>		
3442	007776	005737	000524			TST	R0.EXP
3443	010002				LET CC.EXP := PSW		
3444	010002	013737	177776	000522		MOV	PSW,CC.EXP
3445	010010				LET CC.EXP := CC.EXP CLR.BY #177760		
3446	010010	042737	177760	000522		BIC	#177760,CC.EXP
3447	010016				LET BFADRS := #BUFF1		
3448	010016	012737	063072	000542		MOV	#BUFF1,BFADRS
3449	010024				CALL CLRBUF		
3450	010024	004737	056622			JSR	PC,CLRBUF
3451	010030				LET TEMPO := #BUFF1 + DATA02		
3452	010030	012737	063072	000476		MOV	#BUFF1,TEMPO
3453	010036	063737	000566	000476		ADD	DATA02,TEMPO
3454	010044				LET @TEMPO :=B- DATA01		
3455	010044	113777	000564	170424		MOVB	DATA01,@TEMPO
3456							
3457					:+		
3458					: DO THE CIS INSTRUCTION.		
3459					:-		
3460							
3461	010052				INLINE <LOCC>		

3462	010052	076040			LOCC
3463					
3464					
3465				:+	
3466				: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS	
3467				:-	
3468	010054			LET CC : PSW	
3469	010054	013737	177776	000520	MOV PSW,CC
3470	010062				
3471	010062	042737	177760	000520	BIC #177760,CC
3472	010070				
3473	010070	023737	000520	000522	LMP CC,CC.EXP
3474	010076	001402			BEQ \$50310
3475	010100			CALL ERROR	
3476	010100	004737	060044		JSR PC,ERROR
3477	010104			ENDIF	
3478	010104				\$50310:
3479	010104			IF R0 NE R0.EXP THEN	
3480	010104	020037	000524		CMP R0,R0.EXP
3481	010110	001402			BLQ \$50311
3482	010112			CALL ERROR	
3483	010112	004737	060044		JSR PC,ERROR
3484	010116			ENDIF	
3485	010116				\$50311:
3486	010116			IF R1 NE R1.EXP THEN	
3487	010116	020137	000526		CMP R1,R1.EXP
3488	010122	001402			BEQ \$50312
3489	010124			CALL ERROR	
3490	010124	004737	060044		JSR PC,ERROR
3491	010130			ENDIF	
3492	010130				\$50312:
3493	010130			IF R2 NE R2.EXP THEN	
3494	010130	020237	000530		CMP R2,R2.EXP
3495	010134	001402			BEQ \$50313
3496	010136			CALL ERROR	
3497	010136	004737	060044		JSR PC,ERROR
3498	010142			ENDIF	
3499	010142				\$50313:
3500	010142			IF R3 NE R3.EXP THEN	
3501	010142	020337	000532		CMP R3,R3.EXP
3502	010146	001402			BEQ \$50314
3503	010150			CALL ERROR	
3504	010150	004737	060044		JSR PC,ERROR
3505	010154			ENDIF	
3506	010154				\$50314:
3507	010154			IF R4 NE R4.EXP THEN	
3508	010154	020437	000534		CMP R4,R4.EXP
3509	010160	001402			BEQ \$50315
3510	010162			CALL ERROR	
3511	010162	004737	060044		JSR PC,ERROR
3512	010166			ENDIF	
3513	010166				\$50315:
3514	010166			IF R5 NE R5.EXP THEN	
3515	010166	020537	000536		CMP R5,R5.EXP
3516	010172	001402			BEQ \$50316
3517	010174			CALL ERROR	

3518	010174	004737	060044			JSR	PC,ERROR
3519	010200				ENDIF		
3520	010200					\$50316:	
3521	010200				LET TSTCAS := TSTCAS + #6		
3522	010200	062737	000006	000544		ADD	#6,TSTCAS
3523	010206				LET COUNT := COUNT - #1		
3524	010206	005337	000470			DEC	COUNT
3525					;UNTIL COUNT EQ #0		
3526	010212				IF COUNT NE #0 THEN		
3527	010212	005737	000470			TST	COUNT
3528	010216	001402				BEQ	\$50317
3529	010220				INLINE <JMP \$5040>		
3530	010220	000137	007604			JMP	\$5040
3531	010224				ENDIF		
3532	010224					\$50317:	
3533							
3534							
3535					:+		
3536					: DO ONE INLINE CASE - LOCCI		
3537					:-		
3538							
3539	010224				LET BFADRS := #BUFF1		
3540	010224	012737	063072	000542		MOV	#BUFF1,BFADRS
3541	010232				CALL SETBUF		
3542	010232	004737	056664			JSR	PC,SETBUF
3543	010236				CALL CLRREG		
3544	010236	004737	056604			JSR	PC,CLRREG
3545	010242				LET TEMPO := #BUFF1 + #25		
3546	010242	012737	063072	000476		MOV	#BUFF1,TEMPO
3547	010250	062737	000025	000476		ADD	#25,TEMPO
3548	010256				LET @TEMPO :B= #40		
3549	010256	112777	000040	170212		MOVB	#40,@TEMPO
3550	010264				INLINE <LOCCI>		
3551	010264	076140				LOCCI	
3552	010266				INLINE <.WORD TLOCCI>		
3553	010266	010436				.WORD	TLOCCI
3554	010270				INLINE <.WORD 40>		
3555	010270	000040				.WORD	40
3556	010272				LET TEMPO := PSW		
3557	010272	013737	177776	000476		MOV	PSW,TEMPO
3558	010300				INLINE <BIT #17,TEMPO>		
3559	010300	032737	000017	000476		BIT	#17,TEMPO
3560	010306				IFCOND NE THEN		
3561	010306	001402				BEQ	\$50320
3562	010310				CALL ERROR		
3563	010310	004737	060044			JSR	PC,ERROR
3564	010314				ENDIF		
3565	010314					\$50320:	
3566	010314				LET TEMPO := #BUFLN-25		
3567	010314	012737	000053	000476		MOV	#BUFLN-25,TEMPO
3568	010322				IF RO NE TEMPO THEN		
3569	010322	020037	000476			CMP	RO,TEMPO
3570	010326	001402				BEQ	\$50321
3571	010330				CALL ERROR		
3572	010330	004737	060044			JSR	PC,ERROR
3573	010334				ENDIF		

3574	010334					\$50321:
3575	010334			IF R1 NE #BUFF1+25 THEN		CMP R1,#BUFF1+25
3576	010334	020127	063117			BEQ \$50322
3577	010340	001402				
3578	010342			CALL ERROR		JSR PC,ERROR
3579	010342	004737	060044			
3580	010346			ENDIF		\$50322:
3581	010346					TST R2
3582	010346			IF R2 NE #0 THEN		BEQ \$50323
3583	010346	005702				
3584	010350	001402				JSR PC,ERROR
3585	010352			CALL ERROR		
3586	010352	004737	060044			\$50323:
3587	010356			ENDIF		TST R3
3588	010356					BEQ \$50324
3589	010356			IF R3 NE #0 THEN		
3590	010356	005703				JSR PC,ERROR
3591	010360	001402				
3592	010362			CALL ERROR		\$50324:
3593	010362	004737	060044			TST R4
3594	010366			ENDIF		BEQ \$50325
3595	010366					
3596	010366			IF R4 NE #0 THEN		JSR PC,ERROR
3597	010366	005704				
3598	010370	001402				\$50325:
3599	010372			CALL ERROR		TST R5
3600	010372	004737	060044			BEQ \$50326
3601	010376			ENDIF		
3602	010376					JSR PC,ERROR
3603	010376			IF R5 NE #0 THEN		
3604	010376	005705				\$50326:
3605	010400	001402				JMP XSKPC
3606	010402			CALL ERROR		
3607	010402	004737	060044			
3608	010406			ENDIF		
3609	010406					
3610	010406			INLINE <JMP XSKPC>		
3611	010406	000137	010442			
3612						
3613				::+ :: TABLE OF TEST CASES FOR LOCC. ::-		
3614						
3615						
3616						
3617	010412	000003		TLOCC: .WORD 3		;# OF TEST CASES FOR LOCC
3618						
3619	010414	000000		.WORD 0		;SRC.LEN
3620	010416	000101		.WORD 101		;CHAR.SEARCHED FOR
3621	010420	000041		.WORD 41		;OFFSET
3622						
3623	010422	000060		.WORD 60		;TEST CASE NUMBER 2
3624	010424	000040		.WORD 40		
3625	010426	000026		.WORD 26		
3626						
3627	010430	000034		.WORD 34		;TEST CASE NUMBER 3
3628	010432	000077		.WORD 77		
3629	010434	000040		.WORD 40		

3630
3631
3632 010436 000100
3633 010440 063072

TLOCCI: .WORD BUFLN
.WORD BUFF1

;TEST CASE FOR INLINE


```

3634 .SBTTL SKPC
3635 :*****
3636 :
3637 : SKPC TEST
3638 :
3639 :*****
3640 010442 INLINE <XSKPC:>
3641 010442 XSKPC:
3642 010442 INLINE <SCOPE> SCOPE
3643 010442 000004 LET $TIMES := #20 ;ITERATION COUNT
3644 010444 012737 000020 000404 MOV #20,$TIMES
3645 010444 012737 000020 000404 LET COUNT := *SKPC ;GET TEST COUNT
3646 010452 013737 011302 000470 MOV TSKPC,COUNT
3647 010452 013737 011302 000470 LET TSTCAS := #TSKPC+2 ;POINT TO FIRST TEST CASE
3648 010460 012737 011304 000544 MOV #TSKPC+2,TSTCAS
3649 010460 012737 011304 000544 ;REPEAT
3650 010466 INLINE <$5041:>
3651 010466 $5041:
3652 010466
3653
3654 :+
3655 : GET TEST DATA FROM TABLE
3656 :-
3657
3658 010466 LET R0 := TSTCAS
3659 010466 013700 000544 MOV TSTCAS,R0
3660 010472 LET DATA00 := (R0)+ ;SRC.LEN
3661 010472 012037 000562 MOV (R0)+,DATA00
3662 010476 LET DATA01 := (R0)+ ;NON-SEARCH CHAR
3663 010476 012037 000564 MOV (R0)+,DATA01
3664 010502 LET DATA02 := (R0) ;OFFSET
3665 010502 011037 000566 MOV (R0),DATA02
3666
3667 :+
3668 : SET UP GPR'S
3669 : EXPECTED GPR'S AND CONDITION CODES,
3670 : AND SOURCE BUFFER.
3671 :-
3672
3673 010506 LET R0 := DATA00
3674 010506 013700 000562 MOV DATA00,R0
3675 010512 LET R1 := #BUFF1
3676 010512 012701 063072 MOV #BUFF1,R1
3677 010516 LET R4 := #377
3678 010516 012704 000377 MOV #377,R4
3679 010522 INLINE <BIT #BIT00,COUNT>
3680 010522 032737 000001 000470 BIT #BIT00,COUNT
3681 010530 IF COND EQ THEN
3682 010530 001004 BNE $50327
3683 010532 LET R2 := #0
3684 010532 005002 CLR R2
3685 010534 LET R3 := #0
3686 010534 005003 CLR R3
3687 010536 LET R5 := #0
3688 010536 005005 CLR R5
3689 010540 ELSE
    
```

```

3690 010540 000406
3691 010542
3692 010542
3693 010542 012702 177777
3694 010546
3695 010546 012703 177777
3696 010552
3697 010552 012705 177777
3698 010556
3699 010556
3700 010556
3701 010556 023737 000566 000562
3702 010564 101411
3703 010566
3704 010566 005037 000524
3705 010572
3706 010572 012737 063072 000526
3707 010600 063737 000562 000526
3708 010606
3709 010606 000414
3710 010610
3711 010610
3712 010610 013737 000562 000524
3713 010616 163737 000566 000524
3714 010624
3715 010624 012737 063072 000526
3716 010632 063737 000566 000526
3717 010640
3718 010640
3719 010640
3720 010640 010237 000530
3721 010644
3722 010644 010337 000532
3723 010650
3724 010650 010437 000534
3725 010654
3726 010654 010537 000536
3727 010660
3728 010660 005737 000524
3729 010664
3730 010664 013737 177776 000522
3731 010672
3732 010672 042737 177760 000522
3733 010700
3734 010700 012737 063072 000542
3735 010706
3736 010706 004737 056664
3737 010712
3738 010712 012737 063072 000476
3739 010720 063737 000566 000476
3740 010726
3741 010726 113777 000564 167542
3742
3743
3744
3745
    
```

```

LET R2 := #177777
LET R3 := #177777
LET R5 := #177777
ENDIF
IF DATA02 HI DATA00 THEN
LET R0.EXP := #0
LET R1.EXP := #BUFF1 + DATA00
ELSE
LET R0.EXP := DATA00 - DATA02
LET R1.EXP := #BUFF1 + DATA02
ENDIF
LET R2.EXP := R2
LET R3.EXP := R3
LET R4.EXP := R4
LET R5.EXP := R5
INLINE <TST R0.EXP>
LET CC.EXP := PSW
LET CC.EXP := CC.EXP CLR.BY #177760
LET BFADRS := #BUFF1
CALL SETBUF ;FILL BUFFER WITH SEARCH CHAR
LET TEMPO := #BUFF1 + DATA02
LET @TEMPO :=B= DATA01
;+
; DO THE CIS INSTRUCTION
;-
    
```

```

BR $50330
$50327:
MOV #177777,R2
MOV #177777,R3
MOV #177777,R5
$50330:
CMP DATA02,DATA00
BLOS $50331
CLR R0.EXP
MOV #BUFF1,R1.EXP
ADD DATA00,R1.EXP
BR $50332
$50331:
MOV DATA00,R0.EXP
SUB DATA02,R0.EXP
MOV #BUFF1,R1.EXP
ADD DATA02,R1.EXP
$50332:
MOV R2,R2.EXP
MOV R3,R3.EXP
MOV R4,R4.EXP
MOV R5,R5.EXP
TST R0.EXP
MOV PSW,CC.EXP
BIC #177760,CC.EXP
MOV #BUFF1,BFADRS
JSR PC,SETBUF
MOV #BUFF1,TEMPO
ADD DATA02,TEMPO
MOVB DATA01,@TEMPO
    
```

3746					INLINE <SKPC>		
3747	010734						
3748	010734	076041				SKPC	
3749							
3750					:+		
3751					:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESJLTS		
3752					:-		
3753							
3754	010736				LET CC := PSW		
3755	010736	013737	177776	000520		MOV	PSW,CC
3756	010744				LET CC := CC CLR.BY #177760		
3757	010744	042737	177760	000520		BIC	#177760,CC
3758	010752				IF CC NE CC.EXP THEN		
3759	010752	023737	000520	000522		CMP	CC,CC.EXP
3760	010760	001402				BEQ	\$50333
3761	010762				CALL ERROR		
3762	010762	004737	060044			JSR	PC,ERROR
3763	010766				ENDIF		\$50333:
3764	010766						
3765	010766				IF R0 NE R0.EXP THEN		
3766	010766	020037	000524			CMP	R0,R0.EXP
3767	010772	001402				BEQ	\$50334
3768	010774				CALL ERROR		
3769	010774	004737	060044			JSR	PC,ERROR
3770	011000				ENDIF		\$50334:
3771	011000						
3772	011000				IF R1 NE R1.EXP THEN		
3773	011000	020137	000526			CMP	R1,R1.EXP
3774	011004	001402				BEQ	\$50335
3775	011006				CALL ERROR		
3776	011006	004737	060044			JSR	PC,ERROR
3777	011012				ENDIF		\$50335:
3778	011012						
3779	011012				IF R2 NE R2.EXP THEN		
3780	011012	020237	000530			CMP	R2,R2.EXP
3781	011016	001402				BEQ	\$50336
3782	011020				CALL ERROR		
3783	011020	004737	060044			JSR	PC,ERROR
3784	011024				ENDIF		\$50336:
3785	011024						
3786	011024				IF R3 NE R3.EXP THEN		
3787	011024	020337	000532			CMP	R3,R3.EXP
3788	011030	001402				BEQ	\$50337
3789	011032				CALL ERROR		
3790	011032	004737	060044			JSR	PC,ERROR
3791	011036				ENDIF		\$50337:
3792	011036						
3793	011036				IF R4 NE R4.EXP THEN		
3794	011036	020437	000534			CMP	R4,R4.EXP
3795	011042	001402				BEQ	\$50340
3796	011044				CALL ERROR		
3797	011044	004737	060044			JSR	PC,ERROR
3798	011050				ENDIF		\$50340:
3799	011050						
3800	011050				IF R5 NE R5.EXP THEN		
3801	011050	020537	000536			CMP	R5,R5.EXP

3802	011054	001402							BEQ	\$50341
3803	011056					CALL ERROR			JSR	PC,ERROR
3804	011056	004737	060044							
3805	011062					ENDIF				\$50341:
3806	011062									
3807	011062					LET TSTCAS := TSTCAS + #6			ADD	#6,TSTCAS
3808	011062	062737	000006	000544		LET COUNT := COUNT - #1			DEC	COUNT
3809	011070									
3810	011070	005337	000470							
3811						:UNTIL COUNT EQ #0				
3812	011074					IF COUNT NE #0 THEN			TST	COUNT
3813	011074	005737	000470						BEQ	\$50342
3814	011100	001402								
3815	011102					INLINE <JMP \$5041>			JMP	\$5041
3816	011102	000137	010466							\$50342:
3817	011106					ENDIF				
3818	011106									
3819										
3820										
3821						:+				
3822						:DO ONE INLINE CASE - SKPCI				
3823						:-				
3824										
3825	011106					LET BFADRS :- #BUFF1			MOV	#BUFF1,BFADRS
3826	011106	012737	063072	000542		CALL SETBUF			JSR	PC,SETBUF
3827	011114									
3828	011114	004737	056664			CALL CLRREG			JSR	PC,CLRREG
3829	011120									
3830	011120	004737	056604			LET TEMPO := #BUFF1 + #25			MOV	#BUFF1,TEMPO
3831	011124								ADD	#25,TEMPO
3832	011124	012737	063072	000476		LET @TEMPO :B= #40			MOVB	#40,@TEMPO
3833	011132	062737	000025	000476		INLINE <SKPCI>				
3834	011140								SKPCI	
3835	011140	1,12777	000040	167330		INLINE <.WORD TSKPCI>			.WORD	TSKPCI
3836	011146									
3837	011146	076141				INLINE <.WORD 377>			.WORD	377
3838	011150					LET TEMPO := PSW			MOV	PSW,TEMPO
3839	011150	011320				INLINE <BIT #17,TEMPO>			BIT	#17,TEMPO
3840	011152					IFCOND NE THEN			BEQ	\$50343
3841	011152	000377				CALL ERROR			JSR	PC,ERROR
3842	011154									\$50343:
3843	011154	013737	177776	000476		LET TEMPO :- #BUFLN - #25			MOV	#BUFLN,TEMPO
3844	011162								SUB	#25,TEMPO
3845	011162	032737	000017	000476		IF RO NE TEMPO THEN			CMP	RO,TEMPO
3846	011170								BEQ	\$50344
3847	011170	001402								
3848	011172									
3849	011172	004737	060044							
3850	011176									
3851	011176									
3852	011176									
3853	011176	012737	000100	000476						
3854	011204	162737	000025	000476						
3855	011212									
3856	011212	020037	000476							
3857	011216	001402								

```

3858 011220          CALL ERROR
3859 011220 004737 060044          JSR      PC,ERROR
3860 011224          ENDIF
3861 011224          IF R1 NE #BUFF1+25 THEN
3862 011224          IF R1 NE #BUFF1+25 THEN
3863 011224 020127 063117          CMP      R1,#BUFF1+25
3864 011230 001402          BEQ      $50344:
3865 011232          CALL ERROR
3866 011232 004737 060044          JSR      PC,ERROR
3867 011236          ENDIF
3868 011236          IF R2 NE #0 THEN
3869 011236          IF R2 NE #0 THEN
3870 011236 005702          TST      R2
3871 011240 001402          BEQ      $50345:
3872 011242          CALL ERROR
3873 011242 004737 060044          JSR      PC,ERROR
3874 011246          ENDIF
3875 011246          IF R3 NE #0 THEN
3876 011246          IF R3 NE #0 THEN
3877 011246 005703          TST      R3
3878 011250 001402          BEQ      $50346:
3879 011252          CALL ERROR
3880 011252 004737 060044          JSR      PC,ERROR
3881 011256          ENDIF
3882 011256          IF R4 NE #0 THEN
3883 011256          IF R4 NE #0 THEN
3884 011256 005704          TST      R4
3885 011260 001402          BEQ      $50347:
3886 011262          CALL ERROR
3887 011262 004737 060044          JSR      PC,ERROR
3888 011266          ENDIF
3889 011266          IF R5 NE #0 THEN
3890 011266          IF R5 NE #0 THEN
3891 011266 005705          TST      R5
3892 011270 001402          BEQ      $50350:
3893 011272          CALL ERROR
3894 011272 004737 060044          JSR      PC,ERROR
3895 011276          ENDIF
3896 011276          INLINE <JMP XSCANC>
3897 011276          INLINE <JMP XSCANC>
3898 011276 000137 011324          JMP      XSCANC
3899
3900          ;+
3901          ; TABLE OF TEST CASES FOR SKPC.
3902          ; -
3903
3904 011302 000002          TSKPC:  .WORD  2          ;# OF TEST CASES FOR SKPC
3905
3906 011304 000040          .WORD  40          ;SRC.LEN
3907 011306 000100          .WORD 100          ;NON-SEARCH CHAR.
3908 011310 000060          .WORD  60          ;OFFSET
3909
3910 011312 000050          .WORD  50          ;TEST CASE NUMBER 2
3911 011314 000077          .WORD  77
3912 011316 000030          .WORD  30
3913
    
```

3914
3915 01:320 000100
3916 011322 063072
3917

TSKPCI: .WORD BUFLN
.WORD BUFF1

;TEST CASE FOR INLINE

```
3918 .SBTTL SCANC
3919 :*****
3920 :
3921 : SCANC TEST
3922 :
3923 :*****
3924 011324 INLINE <XSCANC:> XSCANC:
3925 011324
3926 011324 INLINE <SCOPE> SCOPE
3927 011324 000004
3928 011326 LET $TIMES := #20 ;ITERATION COUNT
3929 011326 012737 000020 000404 MOV #20,$TIMES
3930 011334 LET COUNT := ^SCANC ;GET TEST COUNT
3931 011334 013737 012250 000470 MOV TSCANC,COUNT
3932 011342 LET TSTCAS := #TSCANC+2 ;POINT TO FIRST TEST CASE
3933 011342 012737 012252 000544 MOV #TSCANC+2,TSTCAS
3934 :REPEAT
3935 011350 INLINE <$5042:> $5042:
3936 011350
3937
3938 :+
3939 : GET TEST DATA FROM TABLE
3940 :-
3941
3942 011350 LET R0 := TSTCAS
3943 011350 013700 000544 MOV TSTCAS,R0
3944 011354 LET DATA00 := (R0)+ ;SRC.LEN
3945 011354 012037 000562 MOV (R0)+,DATA00
3946 011360 LET DATA01 := (R0) ;OFFSET
3947 011360 011037 000564 MOV (R0),DATA01
3948
```

3949					;	SET UP GPR'S,		
3950					;	EXPECTED GPR'S AND CONDITION CODES		
3951					;	SOURCE BUFFER, AND CHARACTER SET TABLE.		
3952					;			
3953					;			
3954					;			
3955	011364				LET R0 := DATA00			
3956	011364	013700	000562			MOV	DATA00,R0	
3957	011370				LET R1 := #BUFF1			
3958	011370	012701	063072			MOV	#BUFF1,R1	
3959	011374				LET R4 := #1			
3960	011374	012704	000001			MOV	#1,R4	
3961	011400				LET R5 := #CHRSET			
3962	011400	012705	063672			MOV	#CHRSET,R5	
3963	011404				INLINE <BIT #BIT00,COUNT>			
3964	011404	032737	000001	000470	IFCOND EQ THEN	BIT	#BIT00,COUNT	
3965	011412					BNE	\$50352	
3966	011412	001003			LET R2 := #0	CLR	R2	
3967	011414				LET R3 := #0	CLR	R3	
3968	011414	005002			ELSE	BR	\$50353	
3969	011416					\$50352:		
3970	011416	005003			LET R2 := #177777	MOV	#177777,R2	
3971	011420				LET R3 := #177777	MOV	#177777,R3	
3972	011420	000404			ENDIF	\$50353:		
3973	011422				IF DATA01 HI DATA00 THEN	CMP	DATA01,DATA00	
3974	011422	012702	177777			BLOS	\$50354	
3975	011422				LET R0.EXP := #0	CLR	R0.EXP	
3976	011426				LET R1.EXP := #BUFF1 + DATA00	MOV	#BUFF1,R1.EXP	
3977	011426	012703	177777		ELSE	ADD	DATA00,R1.EXP	
3978	011432					BR	\$50355	
3979	011432				LET R0.EXP := DATA00 - DATA01	\$50354:		
3980	011432	023737	000564	000562		MOV	DATA00,R0.EXP	
3981	011432	101411				SUB	DATA01,R0.EXP	
3982	011440				LET R1.EXP := #BUFF1 + DATA01	MOV	#BUFF1,R1.EXP	
3983	011442				ENDIF	ADD	DATA01,R1.EXP	
3984	011442	005037	000524			\$50355:		
3985	011446				LET R2.EXP := R2	MOV	R2,R2.EXP	
3986	011446	012737	063072	000526		MOV	R3,R3.EXP	
3987	011454	063737	000562	000526		MOV	R4,R4.EXP	
3988	011462							
3989	011462	000414						
3990	011464							
3991	011464							
3992	011464	013737	000562	000524				
3993	011472	163737	000564	000524				
3994	011500							
3995	011500	012737	063072	000526				
3996	011506	063737	000564	000526				
3997	011514							
3998	011514							
3999	011514							
4000	011514	010237	000530					
4001	011520							
4002	011520	010337	000532					
4003	011524							
4004	011524	010437	000534					


```

4005 011530          LET R5.EXP := R5
4006 011530 010537 000536          MOV     R5,R5.EXP
4007 011534          INLINE <TST  R0.EXP>
4008 011534 005737 000524          TST    R0.EXP
4009 011540          LET CC.EXP := PSW
4010 011540 013737 177776 000522          MOV    PSW,CC.EXP
4011 011546          LET CC.EXP := CC.EXP CLR.BY #177760
4012 011546 042737 177760 000522          BIC    #177760,CC.EXP
4013 011554          LET BFADRS := #BUFF1
4014 011554 012737 063072 000542          MOV    #BUFF1,BFADRS
4015 011562          CALL CLRBUF
4016 011562 004737 056622          JSR    PC,CLRBUF
4017 011566          LET TEMPO := #BUFF1 + DATA01
4018 011566 012737 063072 000476          MOV    #BUFF1,TEMPO
4019 011574 063737 000564 000476          ADD    DATA01,TEMPO
4020 011602          LET @TEMPO :B- #125
4021 011602 112777 000125 166666          MOVB   #125,@TEMPO
4022 011610          WHILE R5 LO #CHRSET+400 DO
4023 011610
4024 011610 020527 064272          $50356:
4025 011614 103002          CMP    R5,#CHRSET+400
4026 011616          INLINE <CLR (R5)+>          ;CLEAR CHAR SET TABLE
4027 011616 005025          BHS   $50357
4028 011620          ENDDO
4029 011620 000773          CLR   (R5)+
4030 011622          BR    $50356
4031 011622          $50357:
4032 011622 012705 063672          ;RESTORE R5
4033 011626          LET R5 := #CHRSET
4034 011626 012737 063672 000476          MOV    #CHRSET,R5
4035 011634 062737 000125 000476          LET TEMPO := #CHRSET + #125
4036 011642          LET @TEMPO :B- #1
4037 011642 112777 000001 166626          MOV    #CHRSET,TEMPO
4038
4039
4040          ;SET MASK IN CHAR SET TABLE
4041          MOVB   #1,@TEMPO
4042
4043 011650          ;+
4044 011650 076042          ; DO THE CIS INSTRUCTION
4045
4046          ;-
4047          INLINE <SCANC>
4048
4049          SCANC
4050 011652          ;+
4051 011652 013737 177776 000520          ; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS
4052 011660          ;-
4053 011660 042737 177760 000520          LET CC := PSW
4054 011666          LET CC := CC CLR.BY #177760
4055 011666 023737 000520 000522          MOV    PSW,CC
4056 011674 001402          IF CC NE CC.EXP THEN
4057 011676          CALL ERROR
4058 011676 004737 060044          BIC    #177760,CC
4059 011702          CMP    CC,CC.EXP
4060 011702          BEQ   $50360
          JSR    PC,ERROR
          $50360:
    
```

4061	011702			IF R0 NE R0.EXP THEN		
4062	011702	020037	000524		CMP	R0,R0.EXP
4063	011706	001402			BEQ	\$50361
4064	011710			CALL ERROR		
4065	011710	004737	060044		JSR	PC,ERROR
4066	011714			ENDIF		\$50361:
4067	011714					
4068	011714			IF R1 NE R1.EXP THEN		
4069	011714	020137	000526		CMP	R1,R1.EXP
4070	011720	001402			BEQ	\$50362
4071	011722			CALL ERROR		
4072	011722	004737	060044		JSR	PC,ERROR
4073	011726			ENDIF		\$50362:
4074	011726					
4075	011726			IF R2 NE R2.EXP THEN		
4076	011726	020237	000530		CMP	R2,R2.EXP
4077	011732	001402			BEQ	\$50363
4078	011734			CALL ERROR		
4079	011734	004737	060044		JSR	PC,ERROR
4080	011740			ENDIF		\$50363:
4081	011740					
4082	011740			IF R3 NE R3.EXP THEN		
4083	011740	020337	000532		CMP	R3,R3.EXP
4084	011744	001402			BEQ	\$50364
4085	011746			CALL ERROR		
4086	011746	004737	060044		JSR	PC,ERROR
4087	011752			ENDIF		\$50364:
4088	011752					
4089	011752			IF R4 NE R4.EXP THEN		
4090	011752	020437	000534		CMP	R4,R4.EXP
4091	011756	001402			BEQ	\$50365
4092	011760			CALL ERROR		
4093	011760	004737	060044		JSR	PC,ERROR
4094	011764			ENDIF		\$50365:
4095	011764					
4096	011764			IF R5 NE R5.EXP THEN		
4097	011764	020537	000536		CMP	R5,R5.EXP
4098	011770	001402			BEQ	\$50366
4099	011772			CALL ERROR		
4100	011772	004737	060044		JSR	PC,ERROR
4101	011776			ENDIF		\$50366:
4102	011776					
4103	011776			LET TSTCAS := TSTCAS + #4		
4104	011776	062737	000004 000544		ADD	#4,TSTCAS
4105	012004			LET COUNT :- COUNT - #1		
4106	012004	005337	000470		DEC	COUNT
4107				:UNTIL COUNT EQ #0		
4108	012010			IF COUNT NE #0 THEN		
4109	012010	005737	000470		TST	COUNT
4110	012014	001402			BEQ	\$50367
4111	012016			INLINE <JMP \$5042>		
4112	012016	000137	011350		JMP	\$5042
4113	012022			ENDIF		\$50367:
4114	012022					
4115						
4116						

```

4117
4118
4119
4120
4121 012022
4122 012022 012737 063072 000542
4123 012030
4124 012030 004737 056622
4125 012034
4126 012034 012737 063072 000476
4127 012042 062737 000050 000476
4128 012050
4129 012050 112777 000027 166420
4130 012056
4131 012056 012700 063672
4132 012062
4133 012062
4134 012062 020027 064272
4135 012066 103002
4136 012070
4137 012070 005020
4138 012072
4139 012072 000773
4140 012074
4141 012074
4142 012074 012737 063672 000476
4143 012102 062737 000027 000476
4144 012110
4145 012110 112777 000010 166360
4146 012116
4147 012116 004737 056604
4148 012122
4149 012122 076142
4150 012124
4151 012124 012266
4152 012126
4153 012126 012272
4154 012130
4155 012130 013737 177776 000476
4156 012136
4157 012136 032737 000017 000476
4158 012144
4159 012144 001402
4160 012146
4161 012146 004737 060044
4162 012152
4163 012152
4164 012152
4165 012152 012737 000030 000476
4166 012160
4167 012160 020037 000476
4168 012164 001402
4169 012166
4170 012166 004737 060044
4171 012172
4172 012172
    
```

```

: +
: DO ONE INLINE CASE - SCANCI.
: -
LET BFADRS := #BUFF1
CALL CLRBUF
LET TEMPO := #BUFF1 + #50
LET @TEMPO :B= #27
LET RO := #CHRSET
WHILE RO LO #CHRSET+400 DO
    INLINE <CLR (RO)+>
ENDDO
LET TEMPO := #CHRSET + #27
LET @TEMPO :B #10
CALL CLRREG
INLINE <SCANCI>
INLINE <.WORD TSCNCI>
INLINE <.WORD TSCNCI+4>
LET TEMPO := PSW
INLINE <BIT #17,TEMPO>
IFCOND NE THEN
    CALL ERROR
ENDIF
LET TEMPO := #BUFLEN-50
IF RO NE TEMPO THEN
    CALL ERROR
ENDIF
    
```

```

MOV #BUFF1,BFADRS
JSR PC,CLRBUF
MOV #BUFF1,TEMPO
ADD #50,TEMPO
MOVB #27,@TEMPO
MOV #CHRSET,RO
$50370:
CMP RO,#CHRSET+400
BHS $50371
TABLE CLR (RO)+
BR $50370
$50371:
MOV #CHRSET,TEMPO
ADD #27,TEMPO
;SET MASK IN CHAR SET TABLE
MOVB #10,@TEMPO
JSR PC,CLRREG
SCANCI
.WORD TSCNCI
.WORD TSCNCI+4
MOV PSW,TEMPO
BIT #17,TEMPO
BEQ $50372
JSR PC,ERROR
$50372:
MOV #BUFLEN-50,TEMPO
CMP RO,TEMPO
BEQ $50373
JSR PC,ERROR
$50373:
    
```

```

4173 012172          IF R1 NE #BUFF1+50 THEN
4174 012172 020127 063142          CMP      R1,#BUFF1+50
4175 012176 001402          BEQ      $50374
4176 012200          CALL ERROR
4177 012200 004737 060044          JSR      PC,ERROR
4178 012204          ENDIF
4179 012204          $50374:
4180 012204          IF R2 NE #0 THEN
4181 012204 005702          TST      R2
4182 012206 001402          BEQ      $50375
4183 012210          CALL ERROR
4184 012210 004737 060044          JSR      PC,ERROR
4185 012214          ENDIF
4186 012214          $50375:
4187 012214          IF R3 NE #0 THEN
4188 012214 005703          TST      R3
4189 012216 001402          BEQ      $50376
4190 012220          CALL ERROR
4191 012220 004737 060044          JSR      PC,ERROR
4192 012224          ENDIF
4193 012224          $50376:
4194 012224          IF R4 NE #0 THEN
4195 012224 005704          TST      R4
4196 012226 001402          BEQ      $50377
4197 012230          CALL ERROR
4198 012230 004737 060044          JSR      PC,ERROR
4199 012234          ENDIF
4200 012234          $50377:
4201 012234          IF R5 NE #0 THEN
4202 012234 005705          TST      R5
4203 012236 001402          BEQ      $50400
4204 012240          CALL ERROR
4205 012240 004737 060044          JSR      PC,ERROR
4206 012244          ENDIF
4207 012244          $50400:
4208 012244          INLINE <JMP      XSPANC>
4209 012244 000137 012276          JMP      XSPANC
4210
4211          ;+
4212          ; TABLE OF TEST CASES FOR SCANC.
4213          ; -
4214
4215 012250 000003          TSCANC: .WORD 3          ;#OF TEST CASES FOR SCANC
4216          .WORD 0          ;SRC.LEN
4217 012252 000000          .WORD 20          ;OFFSET
4218 012254 000020
4219
4220 012256 000100          .WORD BUFLen          ;TEST CASE NUMBER 2
4221 012260 000030          .WORD 30
4222
4223 012262 000100          .WORD BUFLen          ;TEST CASE NUMBER 3
4224 012264 000110          .WORD BUFLen+10
4225
4226
4227 012266 000100          TSCNCI: .WORD BUFLen          ;TEST CASE FOR INLINE
4228 012270 063072          .WORD BUFF1
    
```

4229 012272 000010
4230 012274 063672

.WORD 10
.WORD CHRSET

```

4231      .SBTTL SPANC
4232      :*****
4233      :
4234      : SPANC TEST
4235      :
4236      :*****
4237 012276  INLINE <XSPANC:>
4238 012276
4239 012276
4240 012276 000004
4241 012300
4242 012300 012737 000020 000404  LET $TIMES := #20 ;ITERATION COUNT
4243 012306 012737 000020 000404  LET COUNT := ^SPANC ;GET TEST COUNT
4244 012306 013737 013172 000470  LET TSTCAS := #TSPANC+2 ;POINT TO FIRST TEST CASE
4245 012314
4246 012314 012737 013174 000544  ;REPEAT
4247
4248 012322  INLINE <$5043:>
4249 012322
4250
4251
4252      :+
4253      : GET TEST DATA FROM TABLE
4254      :-
4255 012322
4256 012322 013700 000544  LET R0 := TSTCAS
4257 012326
4258 012326 012037 000562  LET DATA00 := (R0)+ ;SRC.LEN
4259 012332
4260 012332 011037 000564  LET DATA01 := (R0) ;OFFSET
4261
4262
4263      :+
4264      : SET UP GPR'S,
4265      : EXPECTED GPR'S AND CONDITION CODES,
4266      : SOURCE BUFFER, AND CHARACTER SET TABLE.
4267      :-
4268 012336
4269 012336 013700 000562  LET R0 := DATA00
4270 012342
4271 012342 012701 063072  LET R1 := #BUFF1
4272 012346
4273 012346 012704 000040  LET R4 := #40 ;MASK
4274 012352
4275 012352 012705 063672  LET R5 := #CHRSET
4276 012356
4277 012356 032737 000001 000470  INLINE <BIT #BIT00,COUNT>
4278 012364
4279 012364 001003  IFCOND EQ THEN
4280 012366
4281 012366 005002  LET R2 := #0
4282 012370
4283 012370 005003  LET R3 := #0
4284 012372
4285 012372 000404  ELSE
4286 012374

```

```

XSPANC:
SCOPE
MOV #20,$TIMES
MOV TSPANC,COUNT
MOV #TSPANC+2,TSTCAS
$5043:
MOV TSTCAS,R0
MOV (R0)+,DATA00
MOV (R0),DATA01
MOV DATA00,R0
MOV #BUFF1,R1
MOV #40,R4
MOV #CHRSET,R5
BIT #BIT00,COUNT
BNE $50401
CLR R2
CLR R3
BR $50402
$50401:

```

4287 012374
 4288 012374 012702 177777
 4289 012400
 4290 012400 012703 177777
 4291 012404
 4292 012404
 4293 012404
 4294 012404 023737 000564 000562
 4295 012412 101411
 4296 012414
 4297 012414 005037 000524
 4298 012420
 4299 012420 012737 063072 000526
 4300 012426 063737 000562 000526
 4301 012434
 4302 012434 000414
 4303 012436
 4304 012436
 4305 012436 013737 000562 000524
 4306 012444 163737 000564 000524
 4307 012452
 4308 012452 012737 063072 000526
 4309 012460 063737 000564 000526
 4310 012466
 4311 012466
 4312 012465
 4313 012465 010237 000530
 4314 012472
 4315 012472 010337 000532
 4316 012476
 4317 012476 010437 000534
 4318 012502
 4319 012502 010537 000536
 4320 012506
 4321 012506 005737 000524
 4322 012512
 4323 012512 013737 177776 000522
 4324 012520
 4325 012520 042737 177760 000522
 4326 012526
 4327 012526 012737 063072 000542
 4328 012534
 4329 012534 004737 056622
 4330 012540
 4331 012540 012737 063072 000476
 4332 012546 063737 000564 000476
 4333 012554
 4334 012554 112777 000252 165714
 4335 012562
 4336 012562
 4337 012562 020527 064272
 4338 012566 103002
 4339 012570
 4340 012570 005025
 4341 012572
 4342 012572 000773

LET R2 := #177777
 LET R3 := #177777
 ENDIF
 IF DATA01 HI DATA00 THEN
 LET R0.EXP := #0
 LET R1.EXP := #BUFF1 + DATA00
 ELSE
 LET R0.EXP := DATA00 - DATA01
 LET R1.EXP := #BUFF1 + DATA01
 ENDIF
 LET R2.EXP := R2
 LET R3.EXP := R3
 LET R4.EXP := R4
 LET R5.EXP := R5
 INLINE <TST R0.EXP>
 LET CC.EXP := PSW
 LET CC.EXP := CC.EXP CLR.BY #177760
 LET BFADRS := #BUFF1
 CALL CLRBUF
 LET TEMPO := #BUFF1 + DATA01
 LET @TEMPO :B= #252
 WHILE R5 LO #CHRSET+400 DO
 INLINE <CLR (R5)+>
 ENDDO

MOV #177777,R2
 MOV #177777,R3
 \$50402:
 CMP DATA01,DATA00
 BLOS \$50403
 CLR R0.EXP
 MOV #BUFF1,R1.EXP
 ADD DATA00,R1.EXP
 BR \$50404
 \$50403:
 MOV DATA00,R0.EXP
 SUB DATA01,R0.EXP
 MOV #BUFF1,R1.EXP
 ADD DATA01,R1.EXP
 \$50404:
 MOV R2,R2.EXP
 MOV R3,R3.EXP
 MOV R4,R4.EXP
 MOV R5,R5.EXP
 TST R0.EXP
 MOV PSW,CC.EXP
 BIC #177760,CC.EXP
 MOV #BUFF1,BFADRS
 JSR PC,CLRBUF
 MOV #BUFF1,TEMPO
 ADD DATA01,TEMPO
 MOVB #252,@TEMPO
 \$50405:
 CMP R5,#CHRSET+400
 BHIS \$50406
 ;CLEAR CHAR. SET TABLE
 CLR (R5)+
 BR \$50405

4343	012574				LET R5 := #CHRSET		\$50406:
4344	012574					;RESTORE R5	
4345	012574	012705	063672				MOV #CHRSET,R5
4346	012600				LET CHRSET :B= #40	;SET MASK IN CHAR. SET	
4347	012600	112737	000040	063672			MOVB #40,CHRSET
4348							
4349							
4350					:+		
4351					: DO THE CIS INSTRUCTION		
4352					: -		
4353	012606				INLINE <SPANC>		
4354	012606	076043					SPANC
4355							
4356							
4357					:+		
4358					: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS		
4359					: -		
4360	012610				LET CC := PSW		
4361	012610	013737	177776	000520			MOV PSW,CC
4362	012616				LET CC := CC CLR.BY #177760		
4363	012616	042737	177760	000520			BIC #177760,CC
4364	012624				IF CC NE CC.EXP THEN		
4365	012624	023737	000520	000522			CMP CC,CC.EXP
4366	012632	001402					BEQ \$50407
4367	012634				CALL ERROR		
4368	012634	004737	060044				JSR PC,ERROR
4369	012640				ENDIF		
4370	012640						\$50407:
4371	012640				IF R0 NE R0.EXP THEN		
4372	012640	020037	000524				CMP R0,R0.EXP
4373	012644	001402					BEQ \$50410
4374	012646				CALL ERROR		
4375	012646	004737	060044				JSR PC,ERROR
4376	012652				ENDIF		
4377	012652						\$50410:
4378	012652				IF R1 NE R1.EXP THEN		
4379	012652	020137	000526				CMP R1,R1.EXP
4380	012656	001402					BEQ \$50411
4381	012660				CALL ERROR		
4382	012660	004737	060044				JSR PC,ERROR
4383	012664				ENDIF		
4384	012664						\$50411:
4385	012664				IF R2 NE R2.EXP THEN		
4386	012664	020237	000530				CMP R2,R2.EXP
4387	012670	001402					BEQ \$50412
4388	012672				CALL ERROR		
4389	012672	004737	060044				JSR PC,ERROR
4390	012676				ENDIF		
4391	012676						\$50412:
4392	012676				IF R3 NE R3.EXP THEN		
4393	012676	020337	000532				CMP R3,R3.EXP
4394	012702	001402					BEQ \$50413
4395	012704				CALL ERROR		
4396	012704	004737	060044				JSR PC,ERROR
4397	012710				ENDIF		
4398	012710						\$50413:

4399	012710				IF R4 NE R4.EXP THEN		
4400	012710	020437	000534			CMP	R4,R4.EXP
4401	012714	001402				BEQ	\$50414
4402	012716				CALL ERROR		
4403	012716	004737	060044			JSR	PC,ERROR
4404	012722				ENDIF		
4405	012722					\$50414:	
4406	012722				IF R5 NE R5.EXP THEN		
4407	012722	020537	000536			CMP	R5,R5.EXP
4408	012726	001402				BEQ	\$50415
4409	012730				CALL ERROR		
4410	012730	004737	060044			JSR	PC,ERROR
4411	012734				ENDIF		
4412	012734					\$50415:	
4413	012734				LET TSTCAS := TSTCAS + #4		
4414	012734	062737	000004	000544		ADD	#4,TSTCAS
4415	012742				LET COUNT := COUNT - #1		
4416	012742	005337	000470			DEC	COUNT
4417					:UNTIL COUNT EQ #0		
4418	012746				IF COUNT NE #0 THEN		
4419	012746	005737	000470			TST	COUNT
4420	012752	001402				BEQ	\$50416
4421	012754				INLINE <JMP \$5043>		
4422	012754	000137	012322			JMP	\$5043
4423	012760				ENDIF		
4424	012760					\$50416:	
4425							
4426							
4427					:+		
4428					: DO ONE INLINE CASE - SPANCI		
4429					:-		
4430							
4431	012760				LET BFADRS := #BUFF1		
4432	012760	012737	063072	000542		MOV	#BUFF1,BFADRS
4433	012766				CALL CLRBUF		
4434	012766	004737	056622			JSR	PC,CLRBUF
4435	012772				LET TEMPO := #BUFF1 + #50		
4436	012772	012737	063072	000476		MOV	#BUFF1,TEMPO
4437	013000	062737	000050	000476		ADD	#50,TEMPO
4438	013006				LET @TEMPO :B= #27		
4439	013006	112777	000027	165462		MOVB	#27,@TEMPO
4440	013014				LET R0 := #CHRSET		
4441	013014	012700	063672			MOV	#CHRSET,R0
4442	013020				WHILE R0 LO #CHRSET+400 DO		
4443	013020					\$50417:	
4444	013020	020027	064272			CMP	R0,#CHRSET+400
4445	013024	103002				BHIS	\$50420
4446	013026				LET (R0)+ := #0 ;CLEAR CHAR. SET TABLE		
4447	013026	005020				CLR	(R0)+
4448	013030				ENDDO		
4449	013030	000773				BR	\$50417
4450	013032					\$50420:	
4451	013032				CALL CLRREG		
4452	013032	004737	056604			JSR	PC,CLRREG
4453	013036				LET CHRSET :B= #10		
4454	013036	112737	000010	063672		MOVB	#10,CHRSET

4455	013044				INLINE <SPANCI>		SPANCI
4456	013044	076143					
4457	013046				INLINE <.WORD TSPNCI>		.WORD TSPNCI
4458	013046	013210					
4459	013050				INLINE <.WORD TSPNCI+4>		.WORD TSPNCI+4
4460	013050	013214					
4461	013052				LET TEMPO := PSW		MOV PSW,TEMPO
4462	013052	013737	177776	000476			
4463	013060				INLINE <BIT #17,TEMPO>		BIT #17,TEMPO
4464	013060	032737	000017	000476			
4465	013066				IFCOND NE THEN		BEQ \$50421
4466	013066	001402			CALL ERROR		JSR PC,ERROR
4467	013070						
4468	013070	004737	060044		ENDIF		\$50421:
4469	013074						
4470	013074				LET TEMPO : #BUFLEN-50		MOV #BUFLEN-50,TEMPO
4471	013074						
4472	013074	012737	000030	000476	IF R0 NE TEMPO THEN		CMP R0,TEMPO
4473	013102						BEQ \$50422
4474	013102	020037	000476		CALL ERROR		JSR PC,ERROR
4475	013106	001402					
4476	013110				ENDIF		\$50422:
4477	013110	004737	060044		IF R1 NE #BUFF1+50 THEN		CMP R1,#BUFF1+50
4478	013114						BEQ \$50423
4479	013114				CALL ERROR		JSR PC,ERROR
4480	013114				ENDIF		\$50423:
4481	013114	020127	063142		IF R2 NE #0 THEN		TST R2
4482	013120	001402					BEQ \$50424
4483	013122				CALL ERROR		JSR PC,ERROR
4484	013122	004737	060044		ENDIF		\$50424:
4485	013126						
4486	013126				IF R3 NE #0 THEN		TST R3
4487	013126						BEQ \$50425
4488	013126	005702			CALL ERROR		JSR PC,ERROR
4489	013130	001402					
4490	013132				ENDIF		\$50425:
4491	013132	004737	060044		IF R4 NE #0 THEN		TST R4
4492	013136						BEQ \$50426
4493	013136				CALL ERROR		JSR PC,ERROR
4494	013136				ENDIF		\$50426:
4495	013136	005703			IF R5 NE #0 THEN		TST R5
4496	013140	001402					BEQ \$50427
4497	013142				CALL ERROR		
4498	013142	004737	060044		ENDIF		
4499	013146						
4500	013146				IF R4 NE #0 THEN		TST R4
4501	013146						BEQ \$50426
4502	013146	005704			CALL ERROR		JSR PC,ERROR
4503	013150	001402			ENDIF		\$50426:
4504	013152						
4505	013152	004737	060044		IF R5 NE #0 THEN		TST R5
4506	013156						BEQ \$50427
4507	013156						
4508	013156						
4509	013156	005705					
4510	013160	001402					

```

4511 013162          CALL ERROR
4512 013162 004737 060044          JSR    PC,ERROR
4513 013166          ENDIF
4514 013166          $50427:
4515 013166          INLINE <JMP XCMPG>
4516 013166 000137 013220          JMP    XCMPG
4517
4518
4519      ;+
4520      ; TABLE OF TEST CASES FOR SPANC
4521      ; -
4522 013172 000003      TSPANC: .WORD 3          ;# OF TEST CASES FOR SPANC
4523
4524 013174 000000          .WORD 0          ;SRC.LEN
4525 013176 000030          .WORD 30          ;OFFSET
4526
4527 013200 000100          .WORD BUFLN      ;TEST CASE NUMBER 2
4528 013202 000040          .WORD 40
4529
4530 013204 000100          .WORD BUFLN      ;TEST CASE NUMBER 3
4531 013206 000104          .WORD BUFLN+4
4532
4533
4534 013210 000100      TSPNCI: .WORD BUFLN      ;TEST CASE FOR INLINE
4535 013212 063072          .WORD BUFLN+1
4536 013214 000010          .WORD 10
4537 013216 063672          .WORD CHRSET
    
```

```

4538          .SBTTL  CMPC
4539          :*****
4540          :
4541          :   CMPC TEST
4542          :
4543          :*****
4544 013220      INLINE  <XCMPCL:>
4545 013220
4546 013220      XCMPCL:
4547 013220 000004  INLINE  <SCOPE>
4548 013222      SCOPE
4549 013222 012737 000020 000404 LET $TIMES := #20
4550 013230      :ITERATION COUNT
4551 013230 013737 014172 000470 LET COUNT := *CMPC
4552 013236      :GET TEST COUNT
4553 013236 012737 014174 000544 LET TSTCAS := #TCMPC+2
4554          :POINT TO FIRST TEST CASE
4555 013244      :REPEAT
4556 013244      INLINE  <$5044:>
4557
4558          :
4559          :+
4560          : GET TEST DATA FROM TABLE
4561          :-
4562 013244      LET R0 := TSTCAS
4563 013244 013700 000544      MOV      TSTCAS,R0
4564 013250      LET DATA00 := (R0)+
4565 013250 012037 000562      :SRC1.LEN      MOV      (R0)+,DATA00
4566 013254      LET DATA01 := (R0)+
4567 013254 012037 000564      :SRC2.LEN      MOV      (R0)+,DATA01
4568 013260      LET DATA02 := (R0)
4569 013260 011037 000566      :OFFSET      MOV      (R0),DATA02
4570
4571          :
4572          :+
4573          : SET UP GPR'S
4574          :-
4575 013264      LET R0 := DATA00
4576 013264 013700 000562      :SRC1      MOV      DATA00,R0
4577 013270      LET R1 := #BUFF1
4578 013270 012701 063072      :SRC2      MOV      #BUFF1,R1
4579 013274      LET R2 := DATA01
4580 013274 013702 000564      :SRC2      MOV      DATA01,R2
4581 013300      LET R3 := #BUFF2
4582 013300 012703 063172      :SRC2      MOV      #BUFF2,R3
4583 013304      LET R4 := #377
4584 013304 012704 000377      :FILL CHARACTER MOV      #377,R4
4585 013310      LET OFFSET := DATA02
4586 013310 013737 000566 000512      :OFFSET      MOV      DATA02,OFFSET
4587 013316      INLINE  <BIT  #BIT00,COUNT>
4588 013316 032737 000001 000470      BIT      #BIT00,COUNT
4589 013324      IFCOND EQ THEN
4590 013324 001002      BNE      $50430
4591 013326      LET R5 := #0
4592 013326 005005      CLR      R5
4593 013330      ELSE

```

4594	013330	000402		
4595	013332			
4596	013332			
4597	013332	012705	177777	
4598	013336			
4599	013336			
4600				
4601				
4602				
4603				
4604				
4605	013336			
4606	013336	012737	063072	000542
4607	013344			
4608	013344	004737	056664	
4609	013350			
4610	013350	012737	063172	000542
4611	013356			
4612	013356	004737	056664	
4613	013362			
4614	013362	023737	000562	000564
4615	013370	101412		
4616	013372			
4617	013372	013737	000512	000476
4618	013400	062737	063072	000476
4619	013406			
4620	013406	012737	000001	000522
4621	013414			
4622	013414	000411		
4623	013416			
4624	013416			
4625	013416	013737	000512	000476
4626	013424	062737	063172	000476
4627	013432			
4628	013432	012737	000010	000522
4629	013440			
4630	013440			
4631	013440			
4632	013440	105077	165032	
4633				
4634				
4635				
4636				
4637				
4638	013444			
4639	013444	010037	000524	
4640	013450	163737	000566	000524
4641	013456			
4642	013456	010137	000526	
4643	013462	063737	000566	000526
4644	013470			
4645	013470	010237	000530	
4646	013474	163737	000566	000530
4647	013502			
4648	013502	010337	000532	
4649	013506	063737	000566	000532

```

LET R5 := #177777
ENDIF

```

```

: +
: SET UP BUFFERS
: -

```

```
LET BFADRS := #BUFF1
```

```
CALL SETBUF
```

```
LET BFADRS := #BUFF2
```

```
CALL SETBUF
```

```
IF DATA00 HI DATA01 THEN
```

```
LET TEMPO := OFFSET + #BUFF1
```

```
LET CC.EXP := #1
```

```
ELSE
```

```
LET TEMPO := OFFSET + #BUFF2
```

```
LET CC.EXP := #10
```

```
ENDIF
```

```
LET @TEMPO := B- #0
```

```

: +
: SET UP EXPECTED GPR'S AND CONDITION CODES
: -

```

```
LET R0.EXP := R0 - DATA02
```

```
LET R1.EXP := R1 + DATA02
```

```
LET R2.EXP := R2 - DATA02
```

```
LET R3.EXP := R3 + DATA02
```

```

BR $50431
$50430:
MOV #177777,R5
$50431:
MOV #BUFF1,BFADRS
JSR PC,SETBUF
MOV #BUFF2,BFADRS
JSR PC,SETBUF
CMP DATA00,DATA01
BLOS $50432
MOV OFFSET,TEMPO
ADD #BUFF1,TEMPO
MOV #1,CC.EXP
BR $50433
$50432:
MOV OFFSET,TEMPO
ADD #BUFF2,TEMPO
MOV #10,CC.EXP
$50433:
CLRB @TEMPO
MOV R0,R0.EXP
SUB DATA02,R0.EXP
MOV R1,R1.EXP
ADD DATA02,R1.EXP
MOV R2,R2.EXP
SUB DATA02,R2.EXP
MOV R3,R3.EXP
ADD DATA02,R3.EXP

```

4650	013514				LET R4.EXP := R4		
4651	013514	010437	000534			MOV	R4,R4.EXP
4652	013520				LET R5.EXP := R5		
4653	013520	010537	000536			MOV	R5,R5.EXP
4654	013524				IF DATA02 HIS DATA00 THEN		
4655	013524	023737	000566	000562		CMP	DATA02,DATA00
4656	013532	103406				BLO	\$50434
4657	013534				LET R0.EXP := #0		
4658	013534	005037	000524			CLR	R0.EXP
4659	013540				LET R1.EXP := R1 + R0		
4660	013540	010137	000526			MOV	R1,R1.EXP
4661	013544	060037	000526			ADD	R0,R1.EXP
4662	013550				ENDIF		
4663	013550						\$50434:
4664	013550				IF DATA02 HIS DATA01 THEN		
4665	013550	023737	000566	000564		CMP	DATA02,DATA01
4666	013556	103406				BLO	\$50435
4667	013560				LET R2.EXP := #0		
4668	013560	005037	000530			CLR	R2.EXP
4669	013564				LET R3.EXP := R3 + R2		
4670	013564	010337	000532			MOV	R3,R3.EXP
4671	013570	060237	000532			ADD	R2,R3.EXP
4672	013574				ENDIF		
4673	013574						\$50435:
4674	013574				IF DATA02 HIS DATA00 AND DATA02 HIS DATA01 THEN		
4675	013574	023737	000566	000562		CMP	DATA02,DATA00
4676	013602	103407				BLO	\$50436
4677	013604	023737	000566	000564		CMP	DATA02,DATA01
4678	013612	103403				BLO	\$50436
4679	013614				LET CC.EXP := #4		
4680	013614	012737	000004	000522		MOV	#4,CC.EXP
4681	013622				ENDIF		
4682	013622						\$50436:
4683							
4684					:+		
4685					: DO THE CIS INSTRUCTION		
4686					:-		
4687							
4688	013622				INLINE <CMPC>		
4689	013622	076044				CMPC	
4690							
4691					:+		
4692					: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS		
4693					:-		
4694							
4695	013624				LET CC := PSW		
4696	013624	013737	177776	000520		MOV	PSW,CC
4697	013632				LET CC := CC CLR.BY #177760		
4698	013632	042737	177760	000520		BIC	#177760,CC
4699	013640				IF CC NE CC.EXP THEN		
4700	013640	023737	000520	000522		CMP	CC,CC.EXP
4701	013646	001402				BEQ	\$50437
4702	013650				CALL ERROR		
4703	013650	004737	060044			JSR	PC,ERROR
4704	013654				ENDIF		
4705	013654						\$50437:

4706	013654			IF R0 NE R0.EXP THEN		
4707	013654	020037	000524		CMP	R0,R0.EXP
4708	013660	001402			BEQ	\$50440
4709	013662			CALL ERROR		
4710	013662	004737	060044		JSR	PC,ERROR
4711	013666			ENDIF		
4712	013666					\$50440:
4713	013666			IF R1 NE R1.EXP THEN		
4714	013666	020137	000526		CMP	R1,R1.EXP
4715	013672	001402			BEQ	\$50441
4716	013674			CALL ERROR		
4717	013674	004737	060044		JSR	PC,ERROR
4718	013700			ENDIF		
4719	013700					\$50441:
4720	013700			IF R2 NE R2.EXP THEN		
4721	013700	020237	000530		CMP	R2,R2.EXP
4722	013704	001402			BEQ	\$50442
4723	013706			CALL ERROR		
4724	013706	004737	060044		JSR	PC,ERROR
4725	013712			ENDIF		
4726	013712					\$50442:
4727	013712			IF R3 NE R3.EXP THEN		
4728	013712	020337	000532		CMP	R3,R3.EXP
4729	013716	001402			BEQ	\$50443
4730	013720			CALL ERROR		
4731	013720	004737	060044		JSR	PC,ERROR
4732	013724			ENDIF		
4733	013724					\$50443:
4734	013724			IF R4 NE R4.EXP THEN		
4735	013724	020437	000534		CMP	R4,R4.EXP
4736	013730	001402			BEQ	\$50444
4737	013732			CALL ERROR		
4738	013732	004737	060044		JSR	PC,ERROR
4739	013736			ENDIF		
4740	013736					\$50444:
4741	013736			IF R5 NE R5.EXP THEN		
4742	013736	020537	000536		CMP	R5,R5.EXP
4743	013742	001402			BEQ	\$50445
4744	013744			CALL ERROR		
4745	013744	004737	060044		JSR	PC,ERROR
4746	013750			ENDIF		
4747	013750					\$50445:
4748	013750			LET TSTCAS :- TSTCAS + #6		
4749	013750	062737	000006 000544		ADD	#6,TSTCAS
4750	013756			LET COUNT : COUNT - #1		
4751	013756	005337	000470		DEC	COUNT
4752				:UNTIL COUNT EQ #0		
4753	013762			IF COUNT NE #0 THEN		
4754	013762	005737	000470		TST	COUNT
4755	013766	001402			BEQ	\$50446
4756	013770			INLINE <JMP \$5044>		
4757	013770	000137	013244		JMP	\$5044
4758	013774			ENDIF		
4759	013774					\$50446:
4760						
4761						

```
4762  
4763  
4764  
4765  
4766 013774  
4767 013774 012737 063072 000542  
4768 014002  
4769 014002 004737 056622  
4770 014006  
4771 014006 012737 063172 000542  
4772 014014  
4773 014014 004737 056622  
4774 014020  
4775 014020 004737 056604  
4776 014024  
4777 014024 012737 063072 000476  
4778 014032 062737 000040 000476  
4779 014040  
4780 014040 112777 000377 164430  
4781 014046  
4782 014046 076144  
4783 014050  
4784 014050 014254  
4785 014052  
4786 014052 014260  
4787 014054  
4788 014054 000000  
4789 014056  
4790 014056 013737 177776 000476  
4791 014064  
4792 014064 042737 177760 000476  
4793 014072  
4794 014072 023727 000476 000010  
4795 014100 001402  
4796 014102  
4797 014102 004737 060044  
4798 014106  
4799 014106  
4800 014106  
4801 014106 005700  
4802 014110 001402  
4803 014112  
4804 014112 004737 060044  
4805 014116  
4806 014116  
4807 014116  
4808 014116 005701  
4809 014120 001402  
4810 014122  
4811 014122 004737 060044  
4812 014126  
4813 014126  
4814 014126  
4815 014126 005702  
4816 014130 001402  
4817 014132
```

```
:+  
: DO ONE INLINE CASE - CMPCI  
:-  
LET BFADRS := #BUFF1  
CALL CLRBUF  
LET BFADRS := #BUFF2  
CALL CLRBUF  
CALL CLRREG  
LET TEMPO := #BUFF1 + #40  
LET @TEMPO :B= #377  
INLINE <CMPCI>  
INLINE <.WORD TCMPCI>  
INLINE <.WORD TCMPCI+4>  
INLINE <.WORD 0>  
LET TEMPO := PSW  
LET TEMPO :- TEMPO CLR.BY #177760  
IF TEMPO NE #10 THEN  
CALL ERROR  
ENDIF  
IF R0 NE #0 THEN  
CALL ERROR  
ENDIF  
IF R1 NE #0 THEN  
CALL ERROR  
ENDIF  
IF R2 NE #0 THEN  
CALL ERROR
```

```
MOV #BUFF1,BFADRS  
JSR PC,CLRBUF  
MOV #BUFF2,BFADRS  
JSR PC,CLRBUF  
JSR PC,CLRREG  
MOV #BUFF1,TEMPO  
ADD #40,TEMPO  
MOVB #377,@TEMPO  
CMPCI  
.WORD TCMPCI  
.WORD TCMPCI+4  
.WORD 0  
MOV PSW,TEMPO  
BIC #177760,TEMPO  
CMP TEMPO,#10  
BEQ $50447  
JSR PC,ERROR  
$50447:  
TST R0  
BEQ $50450  
JSR PC,ERROR  
$50450:  
TST R1  
BEQ $50451  
JSR PC,ERROR  
$50451:  
TST R2  
BEQ $50452
```


4818	014132	004737	060044		JSR	PC,ERROR
4819	014136			ENDIF		
4820	014136				\$50452:	
4821	014136			IF R3 NE #0 THEN		
4822	014136	005703			TST	R3
4823	014140	001402			BEQ	\$50453
4824	014142			CALL ERROR		
4825	014142	004737	060044		JSR	PC,ERROR
4826	014146			ENDIF		
4827	014146				\$50453:	
4828	014146			IF R4 NE #0 THEN		
4829	014146	005704			TST	R4
4830	014150	001402			BEQ	\$50454
4831	014152			CALL ERROR		
4832	014152	004737	060044		JSR	PC,ERROR
4833	014156			ENDIF		
4834	014156				\$50454:	
4835	014156			IF R5 NE #0 THEN		
4836	014156	005705			TST	R5
4837	014160	001402			BEQ	\$50455
4838	014162			CALL ERROR		
4839	014162	004737	060044		JSR	PC,ERROR
4840	014166			ENDIF		
4841	014166				\$50455:	
4842	014166			INLINE <JMP XMATC>		
4843	014166	000137	014264		JMP	XMATC
4844						
4845				:+		
4846				: TABLE OF TEST CASES FOR CMPC		
4847				:-		
4848						
4849	014172	000010		TCMPC: .WORD 8.		:# OF TEST CASES FOR CMPC
4850						
4851	014174	000000		.WORD 0		:SRC1.LEN
4852	014176	000020		.WORD 20		:SRC2.LEN
4853	014200	000040		.WORD 40		:OFFSET
4854						
4855	014202	000010		.WORD 10		:TEST CASE NUMBER 2
4856	014204	000030		.WORD 30		
4857	014206	000007		.WORD 7		
4858						
4859	014210	000044		.WORD 44		:TEST CASE NUMBER 3
4860	014212	000070		.WORD 70		
4861	014214	000060		.WORD 60		
4862						
4863	014216	000030		.WORD 30		:TEST CASE NUMBER 4
4864	014220	000050		.WORD 50		
4865	014222	000060		.WORD 60		
4866						
4867	014224	000000		.WORD 0		:TEST CASE NUMBER 5
4868	014226	000000		.WORD 0		
4869	014230	000010		.WORD 10		
4870						
4871	014232	000060		.WORD 60		:TEST CASE NUMBER 6
4872	014234	000040		.WORD 40		
4873	014236	000030		.WORD 30		

4874					
4875	014240	000057	.WORD	57	;TEST CASE NUMBER 7
4876	014242	000033	.WORD	33	
4877	014244	000045	.WORD	45	
4878					
4879	014246	000020	.WORD	20	;TEST CASE NUMBER 8
4880	014250	000011	.WORD	11	
4881	014252	000023	.WORD	23	
4882					
4883					
4884	014254	000060	TCMPCI: .WORD	60	;TEST CASE FOR INLINE
4885	014256	063072	.WORD	BUFF1	
4886	014260	000060	.WORD	60	
4887	014262	063172	.WORD	BUFF2	

```

4888      .SBTTL MATC
4889      :*****
4890      :
4891      :   MATC TEST
4892      :
4893      :*****
4894      014264      INLINE <XMATC:>
4895      014264
4896      014264
4897      014264 000004
4898      014266
4899      014266 012737 000020 000404
4900      014274
4901      014274 013737 015170 000470
4902      014302
4903      014302 012737 015172 000544
4904
4905      014310
4906      014310
4907
4908
4909      :+
4910      : GET TEST DATA FROM TABLE
4911      :-
4912      014310
4913      014310 013700 000544
4914      014314
4915      014314 012037 000562
4916      014320
4917      014320 011037 000564
4918
4919
4920      :+
4921      : SET UP THE GPR'S
4922      :-
4923      014324
4924      014324 012700 000070
4925      014330
4926      014330 012701 063072
4927      014334
4928      014334 013702 000562
4929      014340
4930      014340 012703 063172
4931      014344
4932      014344 032737 000001 000470
4933      014352
4934      014352 001003
4935      014354
4936      014354 005004
4937      014356
4938      014356 005005
4939      014360
4940      014360 000404
4941      014362
4942      014362
4943      014362 012704 177777
    
```

```

XMATC:
SCOPE
:ITERATION COUNT
MOV #20,$TIMES
:GET TEST COUNT
MOV TMATC,COUNT
:POINT TO FIRST TEST CASE
MOV #TMATC+2,TSTCAS
$5045:
:
:
LET R0 := TSTCAS
MOV TSTCAS,R0
LET DATA00 := (R0)+
:OBJ.LEN
MOV (R0)+,DATA00
LET DATA01 := (R0)
:OFFSET
MOV (R0),DATA01
:
:
LET R0 := #70
:SRC.LEN
MOV #70,R0
LET R1 := #BUFF1
:SRC.ADR
MOV #BUFF1,R1
LET R2 := DATA00
MOV DATA00,R2
LET R3 := #BUFF2
:OBJ.LEN
MOV #BUFF2,R3
INLINE <BIT #BIT00,COUNT>
BIT #BIT00,COUNT
IFCOND EQ THEN
BNE $50456
LET R4 := #0
CLR R4
LET R5 := #0
CLR R5
ELSE
BR $50457
$50456:
MOV #177777,R4
    
```

4944	014366			LET R5 := #177777		
4945	014366	012705	177777		MOV	#177777,R5
4946	014372			ENDIF		\$50457:
4947	014372					
4948						
4949				:+		
4950				: SET UP THE BUFFERS		
4951				:-		
4952						
4953	014372			LET BFADRS := #BUFF1		
4954	014372	012737	063072		MOV	#BUFF1,BFADRS
4955	014400			CALL SETBUF		
4956	014400	004737	056664		JSR	PC,SETBUF
4957	014404			LET BFADRS := #BUFF2		
4958	014404	012737	063172		MOV	#BUFF2,BFADRS
4959	014412			CALL CLRBUF		
4960	014412	004737	056622		JSR	PC,CLRBUF
4961	014416			IF DATA01 LO #70 THEN		
4962	014416	023727	000564		CMP	DATA01,#70
4963	014424	103015			BHIS	\$50460
4964	014426			LET R1 := R1 + DATA01		
4965	014426	063701	000564		ADD	DATA01,R1
4966	014432			REPEAT		\$50461:
4967	014432					
4968	014432			INLINE <CLRB (R1)+>		
4969	014432	105021			CLRB	(R1)+
4970	014434			LET TEMPO := #BUFF1 + R0		
4971	014434	012737	063072		MOV	#BUFF1,TEMPO
4972	014442	060037	000476		ADD	R0,TEMPO
4973	014446			UNTIL R1 EQ TEMPO		
4974	014446	020137	000476		CMP	R1,TEMPO
4975	014452	001367			BNE	\$50461
4976	014454			LET R1 := #BUFF1 ;RESTORE R1		
4977	014454	012701	063072		MOV	#BUFF1,R1
4978	014460			ENDIF		\$50460:
4979	014460					
4980						
4981				:+		
4982				: SET UP EXPECTED GPR'S AND CONDITION CODES		
4983				:-		
4984						
4985	014460			LET R0.EXP : R0		
4986	014460	010037	000524		MOV	R0,R0.EXP
4987	014464			LET R1.EXP := R1		
4988	014464	010137	000526		MOV	R1,R1.EXP
4989	014470			LET R2.EXP := R2		
4990	014470	010237	000530		MOV	R2,R2.EXP
4991	014474			LET R3.EXP := R3		
4992	014474	010337	000532		MOV	R3,R3.EXP
4993	014500			LET R4.EXP :- R4		
4994	014500	010437	000534		MOV	R4,R4.EXP
4995	014504			LET R5.EXP :- R5		
4996	014504	010537	000536		MOV	R5,R5.EXP
4997	014510			IF DATA00 HI #0 THEN		
4998	014510	005737	000562		TST	DATA00
4999	014514	101432			BLOS	\$50462

5000	014516				LET TEMPO := RO - DATA01		
5001	014516	010037	000476			MOV	RO,TEMPO
5002	014522	163737	000564	000476		SUB	DATA01,TEMPO
5003	014530				IF TEMPO LO DATA00 THEN		
5004	014530	023737	000476	000562		CMP	TEMPO,DATA00
5005	014536	103007				BHIS	\$50463
5006	014540				LET RO.EXP := #0		
5007	014540	005037	000524			CLR	RO.EXP
5008	014544				LET R1.EXP := RO + R1		
5009	014544	010037	000526			MOV	RO,R1.EXP
5010	014550	060137	000526			ADD	R1,R1.EXP
5011	014554				ELSE		
5012	014554	000412				BR	\$50464
5013	014556					\$50463:	
5014	014556				LET RO.EXP := RO - DATA01		
5015	014556	010037	000524			MOV	RO,RO.EXP
5016	014562	163737	000564	000524		SUB	DATA01,RO.EXP
5017	014570				LET R1.EXP := R1 + DATA01		
5018	014570	010137	000526			MOV	R1,R1.EXP
5019	014574	063737	000564	000526		ADD	DATA01,R1.EXP
5020	014602				ENDIF		
5021	014602					\$50464:	
5022	014602				FNDIF		
5023	014602					\$50462:	
5024	014602				INLINE <TST RO.EXP>		
5025	014602	005737	000524			TST	RO.EXP
5026	014606				LET CC.EXP := PSW		
5027	014606	015137	177776	000522		MOV	PSW,CC.EXP
5028	014614				LET CC.EXP := CC.EXP CLR.BY #177760		
5029	014614	042737	177760	000522		BIC	#177760,CC.EXP
5030							
5031					:+		
5032					: DO THE CIS INSTRUCTION		
5033					:-		
5034							
5035	014622				INLINE <MATC>		
5036	014622	076045				MATC	
5037							
5038					:+		
5039					: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS		
5040					:-		
5041							
5042	014624				LET CC := PSW		
5043	014624	013737	177776	000520		MOV	PSW,CC
5044	014632				LET CC := CC CLR.BY #177760		
5045	014632	042737	177760	000520		BIC	#177760,CC
5046	014640				IF CC NE CC.EXP THEN		
5047	014640	023737	000520	000522		CMP	CC,CC.EXP
5048	014646	001402				BEQ	\$50465
5049	014650				CALL ERROR		
5050	014650	004737	060044			JSR	PC,ERROR
5051	014654				ENDIF		
5052	014654					\$50465:	
5053	014654				IF RO NE RO.EXP THEN		
5054	014654	020037	000524			CMP	RO,RO.EXP
5055	014660	001402				BEQ	\$50466

```

5056 014662          CALL ERROR
5057 014662 004737 060044          JSR      PC,ERROR
5058 014666          ENDIF
5059 014666          $50466:
5060 014666          IF R1 NE R1.EXP THEN
5061 014666 020137 000526          CMP      R1,R1.EXP
5062 014672 001402          BEQ      $50467
5063 014674          CALL ERROR
5064 014674 004737 060044          JSR      PC,ERROR
5065 014700          ENDIF
5066 014700          $50467:
5067 014700          IF R2 NE R2.EXP THEN
5068 014700 020237 000530          CMP      R2,R2.EXP
5069 014704 001402          BEQ      $50470
5070 014706          CALL ERROR
5071 014706 004737 060044          JSR      PC,ERROR
5072 014712          ENDIF
5073 014712          $50470:
5074 014712          IF R3 NE R3.EXP THEN
5075 014712 020337 000532          CMP      R3,R3.EXP
5076 014716 001402          BEQ      $50471
5077 014720          CALL ERROR
5078 014720 004737 060044          JSR      PC,ERROR
5079 014724          ENDIF
5080 014724          $50471:
5081 014724          IF R4 NE R4.EXP THEN
5082 014724 020437 000534          CMP      R4,R4.EXP
5083 014730 001402          BEQ      $50472
5084 014732          CALL ERROR
5085 014732 004737 060044          JSR      PC,ERROR
5086 014736          ENDIF
5087 014736          $50472:
5088 014736          IF R5 NE R5.EXP THEN
5089 014736 020537 000536          CMP      R5,R5.EXP
5090 014742 001402          BEQ      $50473
5091 014744          CALL ERROR
5092 014744 004737 060044          JSR      PC,ERROR
5093 014750          ENDIF
5094 014750          $50473:
5095 014750          LET TSTCAS := TSTCAS + #4
5096 014750 062737 000004 000544          ADD      #4,TSTCAS
5097 014756          LET COUNT := COUNT - #1
5098 014756 005337 000470          DEC      COUNT
5099          :UNTIL COUNT EQ #0
5100          IF COUNT NE #0 THEN
5101 014762 005737 000470          TST      COUNT
5102 014766 001402          BEQ      $50474
5103 014770          INLINE <JMP $5045>
5104 014770 000137 014310          JMP      $5045
5105 014774          ENDIF
5106 014774          $50474:
5107
5108
5109          :+
5110          DO ONE INLINE CASE - MATCI
5111          :-
    
```

5112					LET BFADRS := #BUFF1		
5113	014774					MOV	#BUFF1,BFADRS
5114	014774	012737	063072	000542	CALL SETBUF	JSR	PC,SETBUF
5115	015002	004737	056664			MOV	#BUFF2,BFADRS
5116	015002				LET BFADRS := #BUFF2	JSR	PC,SETBUF
5117	015006					MOV	#BUFF2,BFADRS
5118	015006	012737	063172	000542	CALL SETBUF	JSR	PC,SETBUF
5119	015014	004737	056664		LET R0 := #BUFF1	MOV	#BUFF1,R0
5120	015014				REPEAT		
5121	015020				INLINE <CLRB (R0)+>		\$50475:
5122	015020	012700	063072		UNTIL R0 EQ #BUFF1+#40	CLRB	(R0)+
5123	015024					CMP	R0,#BUFF1+#40
5124	015024					BNE	\$50475
5125	015024				CALL CLRREG	JSR	PC,CLRREG
5126	015024	105020			INLINE <MATCI>	MATCI	
5127	015026				INLINE <.WORD TMATCI>	.WORD	TMATCI
5128	015026	020027	063132		INLINE <.WORD TMATCI+4>	.WORD	TMATCI+4
5129	015032	001374			LET CC := PSW	MOV	PSW,CC
5130	015034				INLINE <BIT #17,CC>	BIT	#17,CC
5131	015034	004737	056604		IFCOND NE THEN	BEQ	\$50476
5132	015040				CALL ERROR	JSR	PC,ERROR
5133	015040	076145			ENDIF		
5134	015042				IF R0 NE #30 THEN		\$50476:
5135	015042	015212			;SRC.LEN - 40	CMP	R0,#30
5136	015044					BEQ	\$50477
5137	015044	015216			CALL ERROR	JSR	PC,ERROR
5138	015046				ENDIF		
5139	015046	013737	177776	000520	IF R1 NE #BUFF1+#40 THEN		\$50477:
5140	015054				CALL ERROR	CMP	R1,#BUFF1+#40
5141	015054	032737	000017	000520	ENDIF	BEQ	\$50500
5142	015062				IF R2 NE #0 THEN	JSR	PC,ERROR
5143	015062	001402			CALL ERROR		
5144	015064	004737	060044		ENDIF		
5145	015064				IF R2 NE #0 THEN		\$50500:
5146	015070				CALL ERROR	TST	R2
5147	015070				ENDIF	BEQ	\$50501
5148	015070				IF R2 NE #0 THEN	JSR	PC,ERROR
5149	015070	020027	000030		CALL ERROR		
5150	015074	001402			ENDIF		
5151	015076				IF R2 NE #0 THEN		
5152	015076	004737	060044		CALL ERROR		
5153	015102				ENDIF		
5154	015102				IF R2 NE #0 THEN		
5155	015102				CALL ERROR		
5156	015102	020127	063132		ENDIF		
5157	015106	001402			IF R2 NE #0 THEN		
5158	015110				CALL ERROR		
5159	015110	004737	060044		ENDIF		
5160	015114				IF R2 NE #0 THEN		
5161	015114				CALL ERROR		
5162	015114				ENDIF		
5163	015114	005702			IF R2 NE #0 THEN		
5164	015116	001402			CALL ERROR		
5165	015120				ENDIF		
5166	015120	004737	060044		IF R2 NE #0 THEN		
5167	015124				CALL ERROR		
					ENDIF		

5168	015124				\$50501:
5169	015124			IF R3 NE #0 THEN	TST R3
5170	015124	005703			BEQ \$50502
5171	015126	001402		CALL ERROR	JSR PC,ERROR
5172	015130			ENDIF	\$50502:
5173	015130	004737	060044		TST R4
5174	015134			IF R4 NE #0 THEN	BEQ \$50503
5175	015134			CALL ERROR	JSR PC,ERROR
5176	015134			ENDIF	\$50503:
5177	015134	005704			TST R5
5178	015136	001402		IF R5 NE #0 THEN	BEQ \$50504
5179	015140			CALL ERROR	JSR PC,ERROR
5180	015140	004737	060044	ENDIF	\$50504:
5181	015144				TST R2
5182	015144			IF R2 NE #0 THEN	BEQ \$50505
5183	015144	005705		CALL ERROR	JSR PC,ERROR
5184	015144	001402		ENDIF	\$50505:
5185	015146				TST R2
5186	015150			IF R2 NE #0 THEN	BEQ \$50505
5187	015150	004737	060044	CALL ERROR	JSR PC,ERROR
5188	015154			ENDIF	\$50505:
5189	015154				TST R2
5190	015154			IF R2 NE #0 THEN	BEQ \$50505
5191	015154	005702		CALL ERROR	JSR PC,ERROR
5192	015156	001402		ENDIF	\$50505:
5193	015160				TST R2
5194	015160	004737	060044	CALL ERROR	BEQ \$50505
5195	015164			ENDIF	JSR PC,ERROR
5196	015164				\$50505:
5197	015164			INLINE <JMP INTST>	JMP INTST
5198	015164	000137	015222		

5199
 5200
 5201 :+
 5202 : TABLE OF TEST DATA FOR MATC
 5203 :-

5204	015170	000004		TMATC: .WORD 4	:# OF TEST CASES FOR MATC
5205					
5206	015172	000000		.WORD 0	:OBJ.LEN
5207	015174	000010		.WORD 10	:OFFSET
5208					
5209	015176	000074		.WORD 74	:TEST CASE NUMBER 2
5210	015200	000010		.WORD 10	
5211					
5212	015202	000040		.WORD 40	:TEST CASE NUMBER 3
5213	015204	000020		.WORD 20	
5214					
5215	015206	000040		.WORD 40	:TEST CASE NUMBER 4
5216	015210	000050		.WORD 50	
5217					
5218					
5219	015212	000070		TMATCI: .WORD 70	:TEST CASE FOR INLINE
5220	015214	063072		.WORD BUFF1	
5221	015216	000010		.WORD 10	
5222	015220	063172		.WORD BUFF2	
5223					


```

5224 .....
5225 :SBTTL INTERRUPTABILITY SUBTEST
5226 .....
5227 .....
5228 :+
5229 : THIS SUBTEST VERIFIES THAT A CIS INSTRUCTION CAN BE
5230 : SUSPENDED (USING CONSOLE TERMINAL PRINTER INTERRUPT).
5231 :-
5232 .....
5233 015222 INLINE <INTST:> INTST:
5234 015222 INTST:
5235 015222 INLINE <SCOPE> SCOPE
5236 015222 000004 IFB $ENV EQ #1 THEN ;IF UNDER APT THEN SCOPE
5237 015224 CMPB $ENV,#1
5238 015224 123727 000652 000001 BNE $50506
5239 015232 001005 IF $PASS NE #0 THEN ; IF 1ST PASS COMPLETED THEN
5240 015234 TST $PASS
5241 015234 005737 000640 BEQ $50507
5242 015240 001402 INLINE <JMP SKPTST> ; SKIP THE TEST
5243 015242 JMP SKPTST
5244 015242 000137 016262 ENDIF ;ELSE
5245 015246 ;DO THE TEST
5246 015246 $50507:
5247 015246 $50506:
5248 015246
5249 .....
5250 015246 LET $TIMES := #1 ;ITERATION COUNT
5251 015246 012737 000001 000404 MOV #1,$TIMES
5252 015254 LET ERR := #2 ;SET UP FOR ERROR MSG
5253 015254 012737 000002 000552 MOV #2,ERR
5254 015262 LET SAVPSW := PSW MOV PSW,SAVPSW
5255 015262 013737 177776 000614 MOV PSW,SAVPSW
5256 015270 LET PSW := #PR7 MOV #PR7,PSW
5257 015270 012737 000340 177776 ;SUSPENDED FLAG,1=INST SUSPENDED
5258 015276 LET TEMPO := #0 CLR TEMPO
5259 015276 005037 000476 LET DELAY := #13000 MOV #13000,DELAY
5260 015302
5261 015302 012737 013000 000472
5262 .....
5263 :+
5264 : SET UP TP VECTOR
5265 :-
5266 .....
5267 015310 LET TPS := #0 ;MAKE SURE INTERRUPT ENABLE IS OFF
5268 015310 005037 177564 CLR TPS
5269 015314 LET TPVEC := #TPSRVC MOV #TPSRVC,TPVEC
5270 015314 012737 016264 000064 MOV #TPSRVC,TPVEC
5271 015322 LET TPVEC+2 := #PR7 MOV #PR7,TPVEC+2
5272 015322 012737 000340 000066
5273 .....
5274 :+
5275 : SET UP FOR MOVTC INST
5276 :-
5277 .....
5278 015330 CALL SAVREG ;SAVE R0-R5
5279 015330 004737 057102 JSR PC,SAVREG
    
```

5280	015334				REPEAT		
5281	015334						\$50510:
5282	015334				LET TEMP1 := DELAY		
5283	015334	013737	000472	000500		MOV	DELAY,TEMP1
5284	015342				LET DATA00 := #1700	;SRC.LEN	
5285	015342	012737	001700	000562		MOV	#1700,DATA00
5286	015350				LET DATA01 := #SPACE	;SRC.ADR	
5287	015350	012737	066006	000564		MOV	#SPACE,DATA01
5288	015356				LET DATA02 := #1740	;DST.LEN	
5289	015356	012737	001740	000566		MOV	#1740,DATA02
5290	015364				LET DATA03 := #SPACEY	;DST.ADR	
5291	015364	012737	072006	000570		MOV	#SPACEY,DATA03
5292	015372				LET DATA04 := #0	;FILL CHAR.	
5293	015372	005037	000572			CLR	DATA04
5294							
5295	015376				LET R0 := DATA00		
5296	015376	013700	000562			MOV	DATA00,R0
5297	015402				LET R1 := DATA01		
5298	015402	013701	000564			MOV	DATA01,R1
5299	015406				LET R2 := DATA02		
5300	015406	013702	000566			MOV	DATA02,R2
5301	015412				LET R3 := DATA03		
5302	015412	013703	000570			MOV	DATA03,R3
5303	015416				LET R4 := DATA04		
5304	015416	013704	000572			MOV	DATA04,R4
5305	015422				LET R5 := #TRANS		
5306	015422	012705	063272			MOV	#TRANS,R5
5307							
5308	015426				LET R0.EXP := #0		
5309	015426	005037	000524			CLR	R0.EXP
5310	015432				LET R1.EXP := #0		
5311	015432	005037	000526			CLR	R1.EXP
5312	015436				LET R2.EXP := #0		
5313	015436	005037	000530			CLR	R2.EXP
5314	015442				LET R3.EXP := #0		
5315	015442	005037	000532			CLR	R3.EXP
5316	015446				LET R4.EXP := R4		
5317	015446	010437	000534			MOV	R4,R4.EXP
5318	015452				LET R5.EXP := R5		
5319	015452	010537	000536			MOV	R5,R5.EXP
5320	015456				LET BFADRS := DATA01		
5321	015456	013737	000564	000542		MOV	DATA01,BFADRS
5322							
5323	015464				CALL ALTBUI		
5324	015464	004737	056774			JSR	PC,ALTBUI
5325	015470				LET BFADRS := DATA03		
5326	015470	013737	000570	000542		MOV	DATA03,BFADRS
5327	015476				CALL ALTBUI		
5328	015476	004737	056774			JSR	PC,ALTBUI
5329							
5330					***		
5331					SET UP THE TRANSLATION TABLE		
5332					***		
5333							
5334	015502				DECR TEMPO FROM #200 TO #1 BY #1		
5335	015502	012737	000200	000476		MOV	#200,TEMPO

5336	015510	000402				BR	\$50511
5337	015512					\$50512:	
5338	015512	005337	000476			DEC	TEMPO
5339	015516					\$50511:	
5340	015516	023727	000476	000001		CMP	TEMPO,#1
5341	015524	002402				BLT	\$50513
5342	015526				LET (R5)+ :- #0		
5343	015526	005025				CLR	(R5)+
5344	015530				ENDDEC		
5345	015530	000770				BR	\$50512
5346	015532					\$50513:	
5347	015532				LET R5 := #TRANS		
5348	015532	012705	063272			MOV	#TRANS,R5
5349	015536				LET (R5) :B= #63		
5350	015536	112715	000063			MOVB	#63,(R5)
5351	015542				LET 252(R5) :B= #314		
5352	015542	112765	000314	000252		MOVB	#314,252(R5)
5353	015550				LET 377(R5) :B= #125		
5354	015550	112765	000125	000377		MOVB	#125,377(R5)
5355							
5356	015556				WHILE #BIT07 NOTSET IN TPS DO		
5357	015556					\$50514:	
5358	015556	032737	000200	177564		BIT	#BIT07,TPS
5359	015564	001001				BNE	\$50515
5360	015566				ENDDO		
5361	015566	000773				BR	\$50514
5362	015570					\$50515:	
5363	015570				LET TPB :- #0		
5364	015570	005037	177566			;SEND CHARACTER TO THE XMIT BUFFER	
5365	015574				INLINE<WAIT:>	CLR	TPB
5366	015574					;WAIT FOR DONE TO DOUBLE BUFFER	
5367	015574				INLINE<BIT #BIT07,TPS>	WAIT:	
5368	015574	032737	000200	177564		; XMIT	
5369	015602				INLINE<BEQ WAIT>	BIT #BIT07,TPS	
5370	015602	001774				BEQ WAIT	
5371	015604				LET TPB := #0	;FILL SECOND XMIT BUFFFER	
5372	015604	005037	177566			CLR	TPB
5373	015610				LET TPS := #BIT6	;SET INTERRUPT ENABLE	
5374	015610	012737	000100	177564		MOV	#BIT6,TPS
5375	015616				LET PSW := #PRO	;ALLOW INTERRUPTS	
5376	015616	012737	000000	177776		MOV	#PRO,PSW
5377	015624				INLINE <LOOP2:>	;DELAY...	
5378	015624					LOOP2:	
5379	015624				INLINE <DEC @#TEMP1>	DEC	@#TEMP1
5380	015624	005337	000500			BNE	LOOP2
5381	015630				INLINE <BNE LOOP2>		
5382	015630	001375				MOVTC	
5383	015632				INLINE <MOVTC>		
5384	015632	076032					
5385	015634				UNTIL TEMPO EQ #1 OR DELAY LT #40		
5386	015634	023727	000476	000001		CMP	TEMPO,#1
5387	015642	001404				BEQ	\$50516
5388	015644	023727	000472	000040		CMP	DELAY,#40
5389	015652	002230				BGE	\$50510
5390	015654					\$50516:	
5391	015654				IF TEMPO NE #1 THEN		

5448	016004		
5449	016004		
5450	016004	020537	000536
5451	016010	001402	
5452	016012		
5453	016012	004737	060044
5454	016016		
5455	016016		
5456			
5457			
5458			
5459			
5460			
5461	016016		
5462	016016	013700	000570
5463	016022		
5464	016022	117702	162536
5465	016026		
5466	016026	042702	177400
5467	016032		
5468	016032	116201	063272
5469	016036		
5470	016036	005037	000456
5471	016042		
5472	016042	013703	000562
5473	016046		
5474	016046		
5475	016046	005703	
5476	016050	001411	
5477	016052	005737	000456
5478	016056	001006	
5479	016060		
5480	016060	122001	
5481	016062	001402	
5482	016064		
5483	016064	004737	060044
5484	016070		
5485	016070		
5486	016070		
5487	016070	005303	
5488	016072		
5489	016072	000765	
5490	016074		
5491	016074		
5492	016074	013700	000570
5493	016100	063700	000562
5494	016104		
5495	016104	013703	000566
5496	016110	163703	000562
5497	016114		
5498	016114		
5499	016114		
5500	016114	122037	000572
5501	016120	001402	
5502	016122		
5503	016122	004737	060044

```

IF R5 NE R5.EXP THEN

CALL ERROR

ENDIF

:+
: CHECK ENTIRE DESTINATION BUFFER FOR CORRECT RESULTS
:-

LET R0 := DATA03

LET R2 := @DATA01

LET R2 := R2 CLR.BY #177400

LET R1 := TRANS(R2)

LET ERRFLG := #0

LET R3 := DATA00

WHILE R3 NE #0 AND ERRFLG EQ #0 DO

IFB (R0)+ NE R1 THEN

CALL ERROR

ENDIF

LET R3 := R3 - #1

ENDDO

LET R0 := DATA03 + DATA00

LET R3 := DATA02 - DATA00

REPEAT

IFB (R0)+ NE DATA04 THEN

CALL ERROR
  
```

```

$50526:
CMP R5,R5.EXP
BEQ $50527

JSR PC,ERROR

$50527:

MOV DATA03,R0
MOVB @DATA01,R2
BIC #177400,R2
MOVB TRANS(R2),R1
CLR ERRFLG
MOV DATA00,R3

$50530:
TST R3
BEQ $50531
TST ERRFLG
BNE $50531

CMPB (R0)+,R1
BEQ $50532

JSR PC,ERROR

$50532:
DEC R3

BR $50530

$50531:
MOV DATA03,R0
ADD DATA00,R0

MOV DATA02,R3
SUB DATA00,R3

$50533:
CMPB (R0)+,DATA04
BEQ $50534

JSR PC,ERROR
  
```

5504	016126			ENDIF					
5505	016126								\$50534:
5506	016126			LET R3 := R3 - #1					DEC R3
5507	016126	005303							TST R3
5508	016130			UNTIL R3 EQ #0 OR ERRFLG NE #0					BEQ \$50535
5509	016130	005703							TST ERRFLG
5510	016132	001403							BEQ \$50533
5511	016134	005737	000456						\$50535:
5512	016140	001765							CMP DATA02,#BUFLEN
5513	016142								BHIS \$50536
5514	016142			IF DATA02 LO #BUFLEN THEN					MOV DATA03,TEMPO
5515	016142	023727	000566						ADD #1777,TEMPO
5516	016150	103025							MOV DATA03,RO
5517	016152			LET TEMPO := DATA03 + #1777					ADD DATA02,RO
5518	016152	013737	000570						
5519	016160	062737	001777						
5520	016166			LET RO := DATA03 + DATA02					
5521	016166	013700	000570						
5522	016172	063700	000566						
5523	016176			REPEAT					
5524	016176								\$50537:
5525	016176			IFB (RO)+ NE BF.EXP THEN					CMPB (RO)+,BF.EXP
5526	016176	122037	000540						BEQ \$50540
5527	016202	001402							
5528	016204			CALL ERROR					JSR PC,ERROR
5529	016204	004737	060044						
5530	016210			ENDIF					
5531	016210								\$50540:
5532	016210			UNTIL RO EQ TEMPO OR ERRFLG NE #0					CMP RO,TEMPO
5533	016210	020037	000476						BEQ \$50541
5534	016214	001403							TST ERRFLG
5535	016216	005737	000456						BEQ \$50537
5536	016222	001765							\$50541:
5537	016224								
5538	016224			ENDIF					\$50536:
5539	016224								
5540	016224			ENDIF					\$50520:
5541	016224								
5542	016224			LET PSW := #PR7					MOV #PR7,PSW
5543	016224	012737	000340						
5544	016232			LET TPVEC := #TPVEC+2					MOV #TPVEC+2,TPVEC
5545	016232	012737	000066						MOV #TPVEC+2,TPVEC
5546	016240			LET TPVEC+2 := #0					CLR TPVEC+2
5547	016240	005037	000066						CLR TPS
5548	016244			LET TPS := #0					
5549	016244	005037	177564						JSR PC,RESREG
5550	016250			CALL RESREG					
5551	016250	004737	057134						
5552	016254			LET PSW := SAVPSW					MOV SAVPSW,PSW
5553	016254	013737	000614						
5554	016262			INLINE <SKPTST:>					SKPTST:
5555	016262								
5556	016262			INLINE <BR ILLTST>					;BR AROUND INT SERVICE RTN
5557	016262	000421							BR ILLTST
5558									
5559				;+					

```

5560      :      TP INTERRUPT SERVICE ROUTINE
5561      :-
5562
5563 016264      INLINE <TPSRVC:>
5564 016264
5565 016264      LET TPS := #0      ;TURN OFF INTERRUPT ENABLE
5566 016264 005037 177564      CLR      TPS
5567 016270      LET TEMP5 := 2(SP)      MOV      2(SP),TEMP5
5568 016270 016637 000002 000510
5569 016276      IF #BIT8 SETIN TEMP5 THEN
5570 016276 032737 000400 000510      BIT      #BIT8,TEMP5
5571 016304 001404      BEQ      $50542
5572 016306      LET TEMPO :- #1      ;INST WAS SUSPENDED
5573 016306 012737 000001 000476      MOV      #1,TEMPO
5574 016314      ELSE
5575 016314 000403      BR      $50543
5576 016316      $50542:
5577 016316      LET DELAY := DELAY - #30      ;DECREMENT DELAY COUNTER AND TRY AGAIN
5578 016316 162737 000030 000472      SUB      #30,DELAY
5579 016324      ENDIF
5580 016324
5581 016324      INLINE <RTI>
5582 016324 000002
5583

```

5584
5585
5586
5587
5588
5589
5590
5591
5592
5593
5594
5595
5596
5597
5598
5599
5600
5601
5602
5603
5604
5605
5606
5607
5608
5609
5610
5611
5612
5613
5614
5615
5616
5617
5618
5619
5620
5621
5622
5623
5624
5625
5626
5627
5628
5629
5630
5631
5632
5633
5634
5635
5636
5637
5638
5639

016326
016326
016326
016326 000004
016330
016330 012737 000020 000404
016336
016336 005037 000456
016342
016342 012737 000003 000552
016350
016350 013737 177776 000614
016356
016356 012737 000340 177776
016364
016364 012737 016412 000010
016372
016372 076033
016374
016374 004737 060044
016400
016400 012737 056570 000010
016406
016406 000137 016450

```

:*****
:SBTTL  ILLEGAL OPCODE SUBTEST
:*****

:+
:      THIS SUBTEST VERIFIES THAT AN ILLEGAL CIS CLASS OPCODE
:      (ILLOP=76033) CAUSES A RESERVED INSTRUCTION TRAP (TRAP 10)
:-

INLINE <ILLTST:>
                                ILLTST:
INLINE <SCOPE>
                                SCOPE
LET $TIMES := #20                ;ITERATION COUNT
MOV        #20,$TIMES
LET ERRFLG := #0                ;RESET ERROR FLAG
CLR        ERRFLG
LET ERR := #3                   ;ERROR MSG #
MOV        #3,ERR
LET SAVPSW := PSW              ;LOCK OUT INTERRUPTS
MOV        PSW,SAVPSW
LET PSW := #PR7                ;LOCK OUT INTERRUPTS
MOV        #PR7,PSW
LET RESVEC := #RESRTN          ;DO ILLEGAL CODE
MOV        #RESRTN,RESVEC
INLINE <ILLOP>
                                ILLOP
CALL ERROR                      ;DIDNT TRAP, ERROR
                                JSR    PC,ERROR
LET RESVEC := #ILLRES          ;RESTORE RESERVED INSTRUCTION TRAP VECTOR
MOV        #ILLRES,RESVEC
INLINE <JMP  CIS4END>
                                ;MODULE EXIT
                                JMP    CIS4END

:+
:      TRAP HANDLER
:-

INLINE <RESRTN:>
                                RESRTN:
IF ERRFLG EQ #0 THEN
                                TST    ERRFLG
                                BNE    $50544
                                ;REMOVE PC,PSW PUSHED BY TRAP
                                MOV    (SP)+,TEMPO
                                MOV    (SP)+,TEMPO
                                $50544:
                                CLR    ERRFLG
LET ERRFLG := #0
                                ;ENSURE ERROR FLAG IS RESET
LET PSW : SAVPSW
                                MOV    SAVPSW,PSW
LET RESVEC := #ILLRES          ;RESTORE RESERVED INSTRUCTION TRAP VECTOR
MOV        #ILLRES,RESVEC
    
```


5640
5641 016450
5642 016450
5643 --
5644 --

INLINE <CIS4END:>

CIS4END:

5645
5646
5647
5648
5649
5650
5651
5652
5653
5654
5655
5656
5657
5658
5659
5660
5661
5662
5663
5664
5665
5666
5667
5668
5669
5670
5671
5672
5673
5674
5675
5676
5677
5678
5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5693
5694
5695
5696

```
.....  
.....  
*  
*          C I S S   M O D U L E  
*  
* THIS MODULE TESTS THE ARITHMETIC CONVERT INSTRUCTIONS FOR  
* PACKED, NUMERIC, AND LONG INTEGER DATA TYPES: CVTLP, CVTLN,  
* CVTPL, CVTNL, CVTPN, AND CVTNP.  
*  
*-----  
*.....  
*.....
```

```
..SBTTL  C I S S  I N I T I A L I Z A T I O N  
*.....  
* C I S S  I N I T I A L I Z A T I O N  
*.....  
*.....
```

```
: IF SWR<5:0> ARE CLEARED OR IF SWR<1> IS SET THEN EXECUTE  
: THE C I S S  M O D U L E ; O T H E R W I S E , S K I P T O T H E N E X T C I S  M O D U L E .
```

```
INLINE  <SCOPE>                                SCOPE  
  
LET TEMPO := @SWR CLR.BY #177700                MOV    @SWR,TEMPO  
                                                BIC    #177700,TEMPO  
IF TEMPO NE #0 THEN  
    TST    TEMPO  
    BEQ    $50545  
    IF #BIT1 NOTSET IN TEMPO THEN                BIT    #BIT1,TEMPO  
                                                BNE    $50546  
    INLINE  <JMP  C I S S E N D >                JMP    C I S S E N D  
    ENDF  
$50546:  
ENDIF  
$50545:  
LET CTL := #5                                ;CONTROL CHIP NUMBER  
LET ERRFLG := #0                             ;INITIALIZE ERROR FLAG  
LET PSW := #340  
LET ERR := #1  
MOV    #5,CTL  
CLR    ERRFLG  
MOV    #340,PSW  
MOV    #1,ERR
```

```
5697 .SBTTL CVTLP
5698 :.....
5699 :
5700 : CVTLP TEST
5701 :.....
5702 :
5703 016536 INLINE <XCVTLP:> XCVTLP:
5704 016536
5705 016536
5706 016536 000004 SCOPE
5707 016540 LET $TIMES := #20 ;ITERATION COUNT
5708 016540 012737 000020 000404 MOV #20,$TIMES
5709 016546 LET COUNT := *CVTLP ;GET TEST COUNT
5710 016546 013737 020046 000470 MOV *CVTLP,COUNT
5711 016554 LET TSTCAS := #*CVTLP*2 ;POINT TO FIRST TEST CASE
5712 016554 012737 020050 000544 MOV #*CVTLP*2,TSTCAS
5713 :REPEAT
5714 016562 INLINE <$5077:> $5077:
5715 016562
5716
5717 :
5718 : GET TEST DATA FROM TABLE
5719 :-
5720
5721 016562 LET R0 := TSTCAS
5722 016562 013700 000544 MOV TSTCAS,R0
5723 016566 LET DATA00 := (R0)+ ;DST.DSCR
5724 016566 012037 000562 MOV (R0)+,DATA00
5725 016572 LET DATA01 := (R0)+ ;LONG INTEGER HI
5726 016572 012037 000564 MOV (R0)+,DATA01
5727 016576 LET DATA02 := (R0)+ ;LONG INTEGER LO
5728 016576 012037 000566 MOV (R0)+,DATA02
5729
5730 :
5731 : PUT EXPECTED RESULTS INTO DS.EXP AND CC.EXP
5732 :-
5733
5734 016602 CALL CLRDSB ;CLEAR DECIMAL STRING BUFFER AREAS
5735 016602 004737 057040 JSR PC,CLRDSB
5736 016606 LET R2 := DATA01
5737 016606 013702 000564 MOV DATA01,R2
```

5738	016612			LET R3 := DATA02	
5739	016612	013703	000566		MOV DATA02,R3
5740	016616			CALL CVTLI	;CONVERT LONG INTEGER TO AN INTERMEDIATE FORM
5741	016616	004737	031114		JSR PC,CVTLI
5742	016622			LET CC.EXP := #0	
5743	016622	005037	000522		CLR CC.EXP
5744	016626			LET DSLEN2 := DATA00 CLR.BY #177740	;GET DST.LEN
5745	016626	013737	000562	000516	MOV DATA00,DSLEN2
5746	016634	042737	177740	000516	BIC #177740,DSLEN2
5747	016642			IF DSLEN2 EQ #0 THEN	;DST.LEN = 0
5748	016642	005737	000516		TST DSLEN2
5749	016646	001014			BNE \$50547
5750	016650			LET CC.EXP := CC.EXP SET.BY #ZBIT	
5751	016650	052737	000004	000522	BIS #ZBIT,CC.EXP
5752	016656			LET DS.EXP := #14	
5753	016656	112737	000014	064442	MOVB #14,DS.EXP
5754	016664			IF DSLEN1 HI #0 THEN	
5755	016664	005737	000514		TST DSLEN1
5756	016670	101403			BLOS \$50550
5757	016672			LET CC.EXP := CC.EXP SET.BY #VBIT	
5758	016672	052737	000002	000522	BIS #VBIT,CC.EXP
5759	016700			ENDIF	
5760	016700			ENDIF	\$50550:
5761	016700			ENDIF	\$50547:
5762	016700			IF DSLEN2 NE #0 THEN	;DST.LEN > 0
5763	016700				TST DSLEN2
5764	016700	005737	000516		BEQ \$50551
5765	016704	001577			;SRC.LEN = 0
5766	016706			IF DSLEN1 EQ #0 THEN	TST DSLEN1
5767	016706	005737	000514		BNE \$50552
5768	016712	001017			MOV DSLEN2,TEMPO
5769	016714			LET TEMPO := DSLEN2	ASR TEMPO
5770	016714	013737	000516	000476	MOV #DS.EXP,RO
5771	016722			INLINE <ASR TEMPO>	ADD TEMPO,RO
5772	016722	006237	000476		ASR TEMPO
5773	016726			LET RO := #DS.EXP * TEMPO	MOV #DS.EXP,RO
5774	016726	012700	064442		ADD TEMPO,RO
5775	016732	063700	000476		MOVB #14,(RO)
5776	016736			LET (RO) := #14	BIS #ZBIT,CC.EXP
5777	016736	112710	000014		BIF \$50553
5778	016742			LET CC.EXP := CC.EXP SET.BY #ZBIT	\$50552:
5779	016742	052737	000004	000522	;DST.LEN = SRC.LEN > 0
5780	016750			ELSE	CMP DSLEN2,DSLEN1
5781	016750	000555			BNE \$50554
5782	016752			IF DSLEN2 EQ DSLEN1 THEN	;SRC POINTER
5783	016752				MOV #DSBUF1,R1
5784	016752	023737	000516	000514	ADD #10,R1
5785	016760	001010			SUB DSLEN1,R1
5786	016762			LET R1 := #DSBUF1 * #10. - DSLEN1	;DST POINTER
5787	016762	012701	064274		MOV #DS.EXP,R2
5788	016766	062701	000012		ENDIF
5789	016772	163701	000514		
5790	016776			LET R2 := #DS.EXP	
5791	016776	012702	064442		MOV #DS.EXP,R2
5792	017002			ENDIF	\$50554:
5793	017002				

5794	017002				IF DSLEN2 LT DSLEN1 THEN		:DST.LEN < SRC.LEN
5795	017002	023737	000516	000514			CMP DSLEN2,DSLEN1
5796	017010	002016					BGE \$50555
5797	017012				LET R1 := #DSBUF1 + #10. - DSLEN2		
5798	017012	012701	064274				MOV #DSBUF1,R1
5799	017016	062701	000012				ADD #10.,R1
5800	017022	163701	000516				SUB DSLEN2,R1
5801	017026				LET R2 := #DS.EXP		
5802	017026	012702	064442				MOV #DS.EXP,R2
5803	017032				LET CC.EXP := CC.EXP SET.BY #VBIT		
5804	017032	052737	000002	000522			BIS #VBIT,CC.EXP
5805	017040				LET DSLEN1 := DSLEN2		
5806	017040	013737	000516	000514			MOV DSLEN2,DSLEN1
5807	017046				ENDIF		
5808	017046						\$50555:
5809	017046				IF DSLEN2 GT DSLEN1 THEN		:DST.LEN > SRC.LEN
5810	017046	023737	000516	000514			CMP DSLEN2,DSLEN1
5811	017054	003426					BLE \$50556
5812	017056				LET TEMPO := DSLEN1		
5813	017056	013737	000514	000476			MOV DSLEN1,TEMPO
5814	017064				INLINE <ASR TEMPO>		
5815	017064	006237	000476				ASR TEMPO
5816	017070				LET TEMP1 := DSLEN2		
5817	017070	013737	000516	000500			MOV DSLEN2,TEMP1
5818	017076				INLINE <ASR TEMP1>		
5819	017076	006237	000500				ASR TEMP1
5820	017102				LET R2 := #DS.EXP + TEMP1 - TEMPO		
5821	017102	012702	064442				MOV #DS.EXP,R2
5822	017106	063702	000500				ADD TEMP1,R2
5823	017112	163702	000476				SUB TEMPO,R2
5824	017116				LET R1 := #DSBUF1 + #10. - DSLEN1		:SRC POINTER
5825	017116	012701	064274				MOV #DSBUF1,R1
5826	017122	062701	000012				ADD #10.,R1
5827	017126	163701	000514				SUB DSLEN1,R1
5828	017132				ENDIF		
5829	017132						\$50556:
5830					:LOAD REMAINING DIGITS INTO DS.EXP		
5831	017132				IF DSLEN1 HI #1 THEN		
5832	017132	023727	000514	000001			CMP DSLEN1,#1
5833	017140	101431					BLOS \$50557
5834	017142				IF #1 NOTSET IN DSLEN1 THEN		
5835	017142	032737	000001	000514			BIT #1,DSLEN1
5836	017150	001003					BNE \$50560
5837	017152				LET (R2)+ := (R1)+		:OUTPUT TO DEST. BUFFER
5838	017152	112122					MOVB (R1)+,(R2)+
5839	017154				LET DSLEN1 := DSLEN1 - #1		
5840	017154	005337	000514				DEC DSLEN1
5841	017160				ENDIF		
5842	017160				LET R0 := #0		\$50560:
5843	017160						:USE R0 TO ASSEMBLE DATA
5844	017160	005000					CLR R0
5845	017162				WHILE DSLEN1 HI #1 DO		
5846	017162						\$50561:
5847	017162	023727	000514	000001			CMP DSLEN1,#1
5848	017170	101415					BLOS \$50562
5849	017172				LET R0 := (R1)+		

5850	017172	112100				MOVW	(R1)+,R0	
5851	017174				INLINE <ASH #4,R0>			
5852	017174	072027	000004		LET TEMPO := R0 OR (R1)+	:OUTPUT	ASH #4,R0 TO DST. BUFFER	
5853	017200						MOVW	R0,TEMPO
5854	017200	110037	000476				BISB	(R1)+,TEMPO
5855	017204	152137	000476		LET (R2)+ := TEMPO			
5856	017210						MOVW	TEMPO,(R2)+
5857	017210	113722	000476		LET DSLENT := DSLENT - #2			
5858	017214						SUB	#2,DSLENT
5859	017214	162737	000002	000514				
5860	017222				ENDDO			
5861	017222	000757					BR	\$50561
5862	017224						\$50562:	
5863	017224				ENDIF		\$50557:	
5864	017224				LET R0 := (R1)+			
5865	017224				INLINE <ASH #4,R0>			
5866	017224	112100			IF DATA0 HIS #70000 THEN		MOVW	(R1)+,R0
5867	017226						ASH	#4,R0
5868	017226	072027	000004				:***	UNSIGNED PACKED ***
5869	017232				LET (R2) := R0 OR #17		(MP	DATA0,#70000
5870	017232	023727	000562	070000			BLO	\$50563
5871	017240	103404					MOVW	R0,(R2)
5872	017242				ELSE		BISB	#17,(R2)
5873	017242	110012					BR	\$50564
5874	017244	152712	000017		IF SIGN EQ #0 THEN		\$50563:	:*** SIGNED PACKED ***
5875	017250						TST	SIGN0
5876	017250	000415			LET (R2) := R0 OR #14		BNE	\$50565
5877	017252						:SIGN	IS +
5878	017252				ELSE		MOVW	R0,(R2)
5879	017252	005737	000450				BISB	#14,(R2)
5880	017256	001004			LET (R2) := R0 OR #15		:SIGN	IS -
5881	017260						MOVW	R0,(R2)
5882	017260	110012			LET CC.EXP := CC.EXP SET.BY #NBIT		BISB	#15,(R2)
5883	017262	152712	000014				BIS	#NBIT,CC.EXP
5884	017266				ENDIF			
5885	017266	000406					\$50566:	
5886	017270				ENDIF		\$50564:	
5887	017270						\$50553:	
5888	017270	110012					\$50551:	
5889	017272	152712	000015					
5890	017276							
5891	017276	052737	000010	000522				
5892	017304							
5893	017304							
5894	017304							
5895	017304							
5896	017304							
5897	017304							
5898	017304							
5899	017304							
5900								
5901								
5902								
5903								
5904								
5905	017304							

*** SET UP GPR'S AND EXPECTED GPR'S. ***
 LET R0 := DATA0

5906	017304	013700	000562			MOV	DATA00,R0
5907	017310			LET R1 := #DS.DST	:DST. ADR		
5908	017310	012701	066400			MOV	#DS.DST,R1
5909	017314			LET R2 := DATA01			
5910	017314	013702	000564			MOV	DATA01,R2
5911	017320			LET R3 := DATA02			
5912	017320	013703	000566			MOV	DATA02,R3
5913	017324			IF #1 SET IN COUNT THEN			
5914	017324	032757	000001	000470		BIT	#1,COUNT
5915	017332	001405				BEQ	\$50567
5916	017334			LET R4 := #177777			
5917	017334	012704	177777			MOV	#177777,R4
5918	017340			LET R5 := #177777			
5919	017340	012705	177777			MOV	#177777,R5
5920	017344			ELSE			
5921	017344	000402				BR	\$50570
5922	017346					\$50567:	
5923	017346			LET R4 := #0			
5924	017346	005004				CLR	R4
5925	017350			LET R5 := #0			
5926	017350	005005				CLR	R5
5927	017352			ENDIF			
5928	017352					\$50570:	

5929	017352			LET R0.EXP := R0		
5930	017352	010037	000524		MOV	R0,R0.EXP
5931	017356			LET R1.EXP := R1		
5932	017356	010137	000526		MOV	R1,R1.EXP
5933	017362			LET R2.EXP := #0		
5934	017362	005037	000530		CLR	R2.EXP
5935	017366			LET R3.EXP := #0		
5936	017366	005037	000532		CLR	R3.EXP
5937	017372			LET R4.EXP := R4		
5938	017372	010437	000534		MOV	R4,R4.EXP
5939	017376			LET R5.EXP := R5		
5940	017376	010537	000536		MOV	R5,R5.EXP
5941						


```
5942
5943
5944
5945
5946 017402
5947 017402 076077
5948
5949
5950
5951
5952
5953
5954 017404
5955 017404 013737 177776 000520
5956 017412
5957 017412 042737 177760 000520
5958 017420
5959 017420 023737 000520 000522
5960 017426 001402

: DO THE CIS INSTRUCTION
:
:
:
: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
:
:
:
LET CC := PSW
MOV PSW,CC
LET CC := CC CLR.BY #177760
BIC #177760,CC
IF CC NE CC.EXP THEN
CMP CC,CC.EXP
BEQ $50571
```

5961	017430			CALL ERROR		
5962	017430	004737	060044		JSR	PC,ERROR
5963	017434			ENDIF		
5964	017434				\$50571:	
5965	017434			IF R0 NE R0.EXP THEN		
5966	017434	020037	000524		CMP	R0,R0.EXP
5967	017440	001402			BEQ	\$50572
5968	017442			CALL ERROR		
5969	017442	004737	060044		JSR	PC,ERROR
5970	017446			ENDIF		
5971	017446				\$50572:	
5972	017446			IF R1 NE R1.EXP THEN		
5973	017446	020137	000526		CMP	R1,R1.EXP
5974	017452	001402			BEQ	\$50573
5975	017454			CALL ERROR		
5976	017454	004737	060044		JSR	PC,ERROR
5977	017460			ENDIF		
5978	017460				\$50573:	
5979	017460			IF R2 NE R2.EXP THEN		
5980	017460	020237	000530		CMP	R2,R2.EXP
5981	017464	001402			BEQ	\$50574
5982	017466			CALL ERROR		
5983	017466	004737	060044		JSR	PC,ERROR
5984	017472			ENDIF		
5985	017472				\$50574:	
5986	017472			IF R3 NE R3.EXP THEN		
5987	017472	020337	000532		CMP	R3,R3.EXP
5988	017476	001402			BEQ	\$50575
5989	017500			CALL ERROR		
5990	017500	004737	060044		JSR	PC,ERROR
5991	017504			ENDIF		
5992	017504				\$50575:	
5993	017504			IF R4 NE R4.EXP THEN		
5994	017504	020437	000534		CMP	R4,R4.EXP
5995	017510	001402			BEQ	\$50576
5996	017512			CALL ERROR		
5997	017512	004737	060044		JSR	PC,ERROR
5998	017516			ENDIF		
5999	017516				\$50576:	
6000	017516			IF R5 NE R5.EXP THEN		
6001	017516	020537	000536		CMP	R5,R5.EXP
6002	017522	001402			BEQ	\$50577
6003	017524			CALL ERROR		
6004	017524	004737	060044		JSR	PC,ERROR
6005	017530			ENDIF		
6006	017530				\$50577:	
6007						
6008						
6009				::+ :: COMPARE THE EXPECTED DECIMAL STRING BUFFER :: WITH ACTUAL RESULTS. ::-		
6010						
6011						
6012						
6013	017530			LET R0 := #DS.DST		
6014	017530	012700	064400		MOV	#DS.DST,R0
6015	017534			LET R1 := #DS.EXP		
6016	017534	012701	064442		MOV	#DS.EXP,R1

6073	017662			IF R0 NE #0 THEN					
6074	017662	005700						TST	R0
6075	017664	001402						BEQ	\$50605
6076	017666			CALL ERROR					
6077	017666	004737	060044					JSR	PC,ERROR
6078	017672			ENDIF					
6079	017672							\$50605:	
6080	017672			IF R1 NE #0 THEN					
6081	017672	005701						TST	R1
6082	017674	001402						BEQ	\$50606
6083	017676			CALL ERROR					
6084	017676	004737	060044					JSR	PC,ERROR
6085	017702			ENDIF					
6086	017702							\$50606:	
6087	017702			IF R2 NE #0 THEN					
6088	017702	005702						TST	R2
6089	017704	001402						BEQ	\$50607
6090	017706			CALL ERROR					
6091	017706	004737	060044					JSR	PC,ERROR
6092	017712			ENDIF					
6093	017712							\$50607:	
6094	017712			IF R3 NE #0 THEN					
6095	017712	005703						TST	R3
6096	017714	001402						BEQ	\$50610
6097	017716			CALL ERROR					
6098	017716	004737	060044					JSR	PC,ERROR
6099	017722			ENDIF					
6100	017722							\$50610:	
6101	017722			IF R4 NE #0 THEN					
6102	017722	005704						TST	R4
6103	017724	001402						BEQ	\$50611
6104	017726			CALL ERROR					
6105	017726	004737	060044					JSR	PC,ERROR
6106	017732			ENDIF					
6107	017732							\$50611:	
6108	017732			IF R5 NE #0 THEN					
6109	017732	005705						TST	R5
6110	017734	001402						BEQ	\$50612
6111	017736			CALL ERROR					
6112	017736	004737	060044					JSR	PC,ERROR
6113	017742			ENDIF					
6114	017742							\$50612:	
6115	017742			LET R0 := #DS.DST					
6116	017742	012700	064400					MOV	#DS.DST,R0
6117	017746			IFB (R0)+ NE #1 THEN					
6118	017746	122027	000001					CMPB	(R0)+,#1
6119	017752	001402						BEQ	\$50613
6120	017754			CALL ERROR					
6121	017754	004737	060044					JSR	PC,ERROR
6122	017760			ENDIF					
6123	017760			REPEAT				\$50613:	
6124	017760							\$50614:	
6125	017760								
6126	017760			IFB (R0)+ NE #21 THEN					
6127	017760	122027	000021					CMPB	(R0)+,#21
6128	017764	001402						BEQ	\$50615

6129	017766			CALL ERROR		
6130	017766	004737	060044		JSR	PC,ERROR
6131	017772			ENDIF		
6132	017772				\$50615:	
6133	017772			UNTIL RO EQ #DS.DST+5		
6134	017772	020027	064405		CMP	RO,#DS.DST+5
6135	017776	001370			BNE	\$50614
6136	020000			IFB (RO)+ NE #37 THEN		
6137	020000	122027	000037		CMPB	(RO)+,#37
6138	020004	001402			BEQ	\$50616
6139	020006			CALL ERROR		
6140	020006	004737	060044		JSR	PC,ERROR
6141	020012			ENDIF		
6142	020012				\$50616:	
6143	020012			LET ERRFLG := #0		
6144	020012	005037	000456	REPEAT	CLR	ERRFLG
6145	020016				\$50617:	
6146	020016			IFB (RO)+ NE #0 THEN		
6147	020016				TSTB	(RO)+
6148	020016	105720			BEQ	\$50620
6149	020020	001402				
6150	020022			CALL ERROR		
6151	020022	004737	060044		JSR	PC,ERROR
6152	020026			ENDIF		
6153	020026				\$50620:	
6154	020026			UNTIL RO EQ #DS.DST+40 OR ERRFLG NE #0		
6155	020026	020027	064440		CMP	RO,#DS.DST+40
6156	020032	001403			BEQ	\$50621
6157	020034	005737	000456		TST	ERRFLG
6158	020040	001766			BEQ	\$50617
6159	020042				\$50621:	
6160	020042			INLINE <JMP XCVTLN>		
6161	020042	000137	020170		JMP	XCVTLN

6162						
6163						
6164				:+		
6165				:+ TABLE OF TEST CASES FOR CVTLP		
6166				:+		
6167				:+		
6168	020046	000014		TCVTLP: .WORD	12.	:# OF TEST CASES FOR CVTLP
6169						
6170	020050	060004		.WORD	60004	;DST. DATA TYPE AND LENGTH
6171	020052	000000		.WORD	0	;LONG INTEGER HIGH & SIGN
6172	020054	000000		.WORD	0	;LONG INTEGER LOW
6173						
6174	020056	060020		.WORD	60020	;TEST CASE NUMBER 2
6175	020060	000000		.WORD	0	
6176	020062	001762		.WORD	1762	
6177						
6178	020064	060000		.WORD	60000	;TEST CASE NUMBER 3
6179	020066	000000		.WORD	0	
6180	020070	152361		.WORD	152361	
6181						
6182	020072	060010		.WORD	60010	;TEST CASE NUMBER 4
6183	020074	074377		.WORD	74377	
6184	020076	171700		.WORD	171700	

6185					
6186	020100	060006	.WORD	60006	;TEST CASE NUMBER 5
6187	020102	000000	.WORD	0	
6188	020104	075147	.WORD	75147	
6189					
6190	020106	060020	.WORD	60020	;TEST CASE NUMBER 6
6191	020110	077777	.WORD	77777	
6192	020112	177777	.WORD	177777	
6193					
6194	020114	060015	.WORD	60015	;TEST CASE NUMBER 7
6195	020116	100000	.WORD	100000	
6196	020120	000000	.WORD	0	
6197					
6198	020122	060010	.WORD	60010	;TEST CASE NUMBER 8
6199	020124	177755	.WORD	177755	
6200	020126	130070	.WORD	130070	
6201					
6202	020130	070010	.WORD	70010	;TEST CASE NUMBER 9
6203	020132	177755	.WORD	177755	
6204	020134	130070	.WORD	130070	
6205					
6206	020136	060037	.WORD	60037	;TEST CASE NUMBER 10
6207	020140	177777	.WORD	177777	
6208	020142	177775	.WORD	177775	
6209					
6210	020144	060003	.WORD	60003	;TEST CASE NUMBER 11
6211	020146	000000	.WORD	0	
6212	020150	000257	.WORD	257	
6213					
6214	020152	060000	.WORD	60000	;TEST CASE NUMBER 12
6215	020154	000000	.WORD	0	
6216	020156	000000	.WORD	0	
6217					
6218					
6219	020160				;IN-LINE TEST CASE
6220	020160	070012	.WORD	70012	;DST TYPE = UNSIGNED PACKED, LENGTH = 12
6221	020162	064400	.WORD	DS.DST	;DST ADDRESS
6222	020164	032707	.WORD	32707	;LONG INTEGER LOW WORD +
6223	020166	041072	.WORD	41072	; LONG INTEGER HIGH WORD AND SIGN
6224					; = 1111111111 IN BCD
6225					; - 1,21,21,21,21,21,37 IN PACKED BYTES

```

6226 .SBTTL CVTLN
6227 :*****
6228 :
6229 : CVTLN TEST
6230 :*****
6231 :
6232 020170 INLINE <XCVTLN:>
6233 020170
6234 020170 XCVTLN:
6235 020170 000004 SCOPE
6236 020172 LET $TIMES := #20 ;ITERATION COUNT
6237 020172 012737 000020 000404 MOV #20,$TIMES
6238 020200 LET COUNT := ^CVTLN ;GET TEST COUNT
6239 020200 013737 021604 000470 MOV ^CVTLN,COUNT
6240 020206 LET TSTCAS := #TCVTLN+2 ;POINT TO FIRST TEST CASE
6241 020206 012737 021606 000544 MOV #TCVTLN+2,TSTCAS
6242 :REPEAT
6243 020214 INLINE <$5057:>
6244 020214 $5057:
6245
6246 :+
6247 : GET TEST DATA FROM TABLE
6248 :-
6249
6250 020214 LET R0 := TSTCAS
6251 020214 013700 000544
6252 020220 LET DATA00 := (R0)+ ;DST. DATA TYPE AND LENGTH
6253 020220 012037 000562 MOV TSTCAS,R0
6254 020224 LET DATA01 := (R0)+ ;LONG INTEGER HIGH
6255 020224 012037 000564 MOV (R0)+,DATA00
6256 020230 LET DATA02 := (R0)+ ;LONG INTEGER LOW
6257 020230 012037 000566 MOV (R0)+,DATA01
6258
6259 :+
6260 : PUT EXPECTED RESULTS INTO DS.EXP AND CC.EXP
6261 :-
6262
6263 020234 CALL CLRDSB ;CLEAR DECIMAL STRING BUFFER AREA
6264 020234 004737 057040 JSR PC,CLRDSB
6265 020240 LET R2 := DATA01
6266 020240 013702 000564 MOV DATA01,R2
6267 020244 LET R3 := DATA02
6268 020244 013703 000566 MOV DATA02,R3
6269 020250 CALL CVTLI ;CONVERT LONG INTEGER TO AN INTERMEDIATE FORM
6270 020250 004737 031114 JSR PC,CVTLI
6271 020254 LET CC.EXP := #0
6272 020254 005037 000522 CLR CC.EXP
6273 020260 LET DSLEN2 := DATA00 CLR.BY #177740 ;GET DST.LEN
6274 020260 013737 000562 000516 MOV DATA00,DSLEN2
6275 020266 042737 177740 000516 BIC #177740,DSLEN2
6276 020274 LET DATTYP := DATA00 CLR.BY #107777 ;GET DATA TYPE
6277 020274 013737 000562 000464 MOV DATA00,DATTYP
6278 020302 042737 107777 000464 BIC #107777,DATTYP
6279 020310 IF DSLEN2 EQ #0 THEN ;DST.LEN = 0
6280 020310 005737 000516 TST DSLEN2
6281 020314 001053 BNE $50622
  
```

6282	020316				LET CC.EXP := CC.EXP SET.BY #ZBIT		
6283	020316	052737	000004	000522		BIS	#ZBIT,CC.EXP
6284	020324				IF DSLEN1 HI #0 THEN	; INTERMEDIATE LENGTH > 0	
6285	020324	005737	000514			TST	DSLEN1
6286	020330	101403				BLOS	\$50623
6287	020332				LET CC.EXP := CC.EXP SET.BY #VBIT		
6288	020332	052737	000002	000522		BIS	#VBIT,CC.EXP
6289	020340				ENDIF		
6290	020340					\$50623:	
6291	020340				IF DATTYP EQ #40000 THEN	; TRAILING SEPARATE	
6292	020340	023727	000464	040000		CMP	DATTYP,#40000
6293	020346	001015				BNE	\$50624
6294	020350				IF SIGN0 EQ #0 THEN		
6295	020350	005737	000450			TST	SIGN0
6296	020354	001004				BNE	\$50625
6297	020356				LET DS.EXP :B= #53		
6298	020356	112737	000053	064442		MOVB	#53,DS.EXP
6299	020364				ELSE		
6300	020364	000406				BR	\$50626
6301	020366					\$50625:	
6302	020366				LET DS.EXP :B= #55		
6303	020366	112737	000055	064442		MOVB	#55,DS.EXP
6304	020374				LET CC.EXP := CC.EXP SET.BY #NBIT		
6305	020374	052737	000010	000522		BIS	#NBIT,CC.EXP
6306	020402				ENDIF		
6307	020402					\$50626:	
6308	020402				ENDIF		
6309	020402					\$50624:	
6310	020402				IF DATTYP EQ #50000 THEN	; LEADING SEPARATE	
6311	020402	023727	000464	050000		CMP	DATTYP,#50000
6312	020410	001015				BNE	\$50627
6313	020412				IF SIGN0 EQ #0 THEN		
6314	020412	005737	000450			TST	SIGN0
6315	020416	001004				BNE	\$50630
6316	020420				LET DS.EXP-1 :B= #53		
6317	020420	112737	000053	064441		MOVB	#53,DS.EXP-1
6318	020426				ELSE		
6319	020426	000406				BR	\$50631
6320	020430					\$50630:	
6321	020430				LET DS.EXP-1 :B= #55		
6322	020430	112737	000055	064441		MOVB	#55,DS.EXP-1
6323	020436				LET CC.EXP := CC.EXP SET.BY #NBIT		
6324	020436	052737	000010	000522		BIS	#NBIT,CC.EXP
6325	020444				ENDIF		
6326	020444					\$50631:	
6327	020444				ENDIF		
6328	020444					\$50627:	
6329	020444				ENDIF		
6330	020444					\$50622:	
6331	020444				IF DSLEN2 GT #0 AND DSLEN1 EQ #0 THEN	; DST.LEN > 0, SRC.LEN - 0	
6332	020444	005737	000516			TST	DSLEN2
6333	020450	003420				BLE	\$50632
6334	020452	005737	000514			TST	DSLEN1
6335	020456	001015				BNE	\$50632
6336	020460				LET R1 := #DS.EXP		
6337	020460	012701	064442			MOV	#DS.EXP,R1


```

6338 020464
6339 020464 013702 000516
6340 020470
6341 020470
6342 020470 005702
6343 020472 101404
6344 020474
6345 020474 112721 000060
6346 020500
6347 020500 005302
6348 020502
6349 020502 000772
6350 020504
6351 020504
6352 020504 052737 000004 000522
6353 020512
6354 020512
6355 020512
6356 020512 005737 000516
6357 020516 003565
6358 020520 005737 000514
6359 020524 003562
6360 020526
6361 020526 012701 064274
6362 020532 062701 000012
6363 020536 163701 000514
6364 020542
6365 020542 012702 064442
6366 020546
6367 020546 023737 000516 000514
6368 020554 003413
6369 020556
6370 020556 063702 000516
6371 020562 163702 000514
6372 020566
6373 020566 012100 064442
6374 020572
6375 020572
6376 020572 020002
6377 020574 001403
6378 020576
6379 020576 112720 000060
6380 020602
6381 020602 000773
6382 020604
6383 020604
6384 020604
6385 020604
6386 020604 023737 000516 000514
6387 020612 103007
6388 020614
6389 020614 063701 000514
6390 020620 163701 000516
6391 020624
6392 020624 052737 000002 000522
6393 020632
    
```

```

LET R2 := DSLEN2
WHILE R2 HI #0 DO

    LET (R1)+ :B= #60
    LET R2 := R2 - #1
ENDDO

LET CC.EXP := CC.EXP SET.BY #ZBIT
ENDIF
IF DSLEN2 GT #0 AND DSLEN1 GT #0 THEN

    LET R1 := #DSBUF1 + #10. - DSLEN1

    LET R2 := #DS.EXP
    IF DSLEN2 GT DSLEN1 THEN

        LET R2 := R2 + DSLEN2 - DSLEN1

        LET R0 := #DS.EXP
        WHILE R0 NE R2 DO

            LET (R0)+ :B= #60
        ENDDO
    ENDIF
    IF DSLEN2 LO DSLEN1 THEN

        LET R1 := R1 + DSLEN1 - DSLEN2

        LET CC.EXP := CC.EXP SET.BY #VBIT
    ENDIF
    
```

```

MOV DSLEN2,R2
$50633:
TST R2
BLOS $50634
MOVB #60,(R1)+
DEC R2
BR $50633
$50634:
BIS #ZBIT,CC.EXP
$50632:
;DST.LEN > 0 AND SRC.LEN > 0
TST DSLEN2
BLE $50635
TST DSLEN1
BLE $50635
MOV #DSBUF1,R1
ADD #10.,R1
SUB DSLEN1,R1
MOV #DS.EXP,R2
;DST.LEN > SRC.LEN > 0
CMP DSLEN2,DSLEN1
BLE $50636
ADD DSLEN2,R2
SUB DSLEN1,R2
MOV #DS.EXP,R0
$50637:
CMP R0,R2
BEQ $50640
MOVB #60,(R0)+
BR $50637
$50640:
$50636:
;SRC.LEN > DST.LEN > 0
CMP DSLEN2,DSLEN1
BHS $50641
ADD DSLEN1,R1
SUB DSLEN2,R1
BIS #VBIT,CC.EXP
    
```

6394	020632			
6395	020632			
6396	020632			
6397	020632			
6398	020632	112137	000476	
6399	020636	152737	000060	000476
6400	020644			
6401	020644	113722	000476	
6402	020650			
6403	020650	020127	064306	
6404	020650	001366		
6405	020656			
6406	020656	005737	000464	
6407	020662	001012		
6408	020664	023727	000450	177777
6409	020672	001006		
6410	020674			
6411	020674	052737	000010	000522
6412	020702			
6413	020702	005302		
6414	020704			
6415	020704	152712	000160	
6416	020710			
6417	020710			
6418	020710			
6419	020710	023727	000464	020000
6420	020716	001404		
6421	020720	023727	000464	030000
6422	020726	001026		
6423	020730			
6424	020730			
6425	020730	023727	000464	020000
6426	020736	001002		
6427	020740			
6428	020740	005302		
6429	020742			
6430	020742	000402		
6431	020744			
6432	020744			
6433	020744	012702	064442	
6434	020750			
6435	020750			
6436	020750			
6437	020750	111200		
6438	020752	142700	000060	
6439	020756			
6440	020756	023727	000450	177777
6441	020764	001005		
6442	020766			
6443	020766	062700	000012	
6444	020772			
6445	020772	052737	000010	000522
6446	021000			
6447	021000			
6448	021000			
6449	021000	116012	056530	

```

REPEAT
    LET TEMPO :B= (R1)+ SET.BY #60
    LET (R2)+ :B= TEMPO
UNTIL R1 EQ #DSBUF1+12

IF DATTYP EQ #0 AND SIGN0 EQ #-1 THEN
    LET CC.EXP := CC.EXP SET.BY #NBIT
    LET R2 := R2 - #1
    LET (R2) :B= (R2) SET.BY #160
ENDIF

IF DATTYP EQ #20000 OR DATTYP EQ #30000 THEN
    IF DATTYP EQ #20000 THEN
        LET R2 := R2 - #1
    ELSE
        LET R2 := #DS.EXP
    ENDIF
    LET R0 :B= (R2) CLR.BY #60

    IF SIGN0 EQ #-1 THEN
        LET R0 := R0 + #10.
        LET CC.EXP := CC.EXP SET.BY #NBIT
    ENDIF

    LET (R2) :B= OVPNCH(R0)

```

```

$50641:
$50642:
;TRANSFER THE DIGITS TO DS.EXP
MOVB (R1)+,TEMPO
BISB #60,TEMPO

MOVB TEMPO,(R2)+

CMP R1,#DSBUF1+12
BNE $50642

TST DATTYP
BNE $50643
CMP SIGN0,#-1
BNE $50643

BIS #NBIT,CC.EXP

DEC R2
;*** SIGNED ZONE, NEGATIVE ***
BISB #160,(R2)

$50643:
;*** OVERPUNCH ***
CMP DATTYP,#20000
BEQ $50644
CMP DATTYP,#30000
BNE $50645

$50644:
;TRAILING
CMP DATTYP,#20000
BNE $50646

DEC R2
;LEADING
BR $50647

$50646:
MOV #DS.EXP,R2

$50647:
MOVB (R2),R0
BICB #60,R0
;NEG. SIGN
CMP SIGN0,#-1
BNF $50650

ADD #10.,R0

BIS #NBIT,CC.EXP

$50650:
;GET APPROPRIATE SIGN/DIGIT FROM
MOVB OVPNCH(R0),(R2)

```

```

6450 021004
6451 021004
6452 021004
6453 021004 023727 000464 040000
6454 021012 001404
6455 021014 023727 000464 050000
6456 021022 001023
6457 021024
6458 021024
6459 021024 023727 000464 050000
6460 021032 001003
6461 021034
6462 021034 012702 064442
6463 021040 005302
6464 021042
6465 021042
6466 021042
6467 021042 023727 000450 177777
6468 021050 001006
6469 021052
6470 021052 112712 000055
6471 021056
6472 021056 052737 000010 000522
6473 021064
6474 021064 000402
6475 021066
6476 021066
6477 021066 112712 000053
6478 021072
6479 021072
6480 021072
6481 021072
6482 021072
6483 021072
6484
6485
6486
6487
6488
6489 021072
6490 021072 013700 000562
6491 021076
6492 021076 012701 064400
6493 021102
6494 021102 013702 000564
6495 021106
6496 021106 013703 000566
6497 021112
6498 021112 032737 000001 000470
6499 021120 001405
6500 021122
6501 021122 012704 177777
6502 021126
6503 021126 012705 177777
6504 021132
6505 021132 000402

```

```

ENDIF
IF DATTYP EQ #40000 OR DATTYP EQ #50000 THEN
    IF DATTYP EQ #50000 THEN
        LET R2 := #DS.EXP - #1
    ENDIF
    IF SIGN0 EQ #-1 THEN
        LET (R2) :B= #55
        LET CC.EXP := CC.EXP SET.BY #NBIT
    ELSE
        LET (R2) :B= #53
    ENDIF
ENDIF
ENDIF
ENDIF
::
:: SET UP GPR'S AND EXPECTED GPR'S.
::
LET R0 := DATA00 ;DATA TYPE AND LENGTH
LET R1 := #DS.DST ;DST. ADR
LET R2 := DATA01
LET R3 := DATA02
IF #1 SETIN COUNT THEN
    LET R4 := #177777
    LET R5 := #177777
ELSE

```

```

$50645:
;*** SEPARATE ***
CMP DATTYP,#40000
BEQ $50651
CMP DATTYP,#50000
BNE $50652
$50651:
;LEADING
CMP DATTYP,#50000
BNE $50653
MOV #DS.EXP,R2
DEC R2
$50653:
CMP SIGN0,#-1
BNE $50654
;NEGATIVE
MOV# #55,(R2)
BIS #NBIT,CC.EXP
BR $50655
$50654:
;POSITIVE
MOV# #53,(R2)
$50655:
$50652:
$50635:
MOV DATA00,R0
MOV #DS.DST,R1
MOV DATA01,R2
MOV DATA02,R3
BIT #1,COUNT
BEQ $50656
MOV #177777,R4
MOV #177777,R5
BR $50657

```

6506	021134							\$50656:
6507	021134			LET R4 := #0				CLR R4
6508	021134	005004						CLR R5
6509	021136			LET R5 := #0				
6510	021136	005005						\$50657:
6511	021140			ENDIF				
6512	021140			LET R0.EXP := R0				MOV R0,R0.EXP
6513	021140	010057	000524					MOV R1,R1.EXP
6514	021140			LET R1.EXP := R1				CLR R2.EXP
6515	021144							CLR R3.EXP
6516	021144	010137	000526	LET R2.EXP := #0				MOV R4,R4.EXP
6517	021150			LET R3.EXP := #0				MOV R5,R5.EXP
6518	021150	005037	000530	LET R4.EXP := R4				
6519	021154			LET R5.EXP := R5				
6520	021154	005037	000532					
6521	021160							
6522	021160	010437	000534					
6523	021164							
6524	021164	010537	000536					
6525								
6526				:+				
6527				: DO THE CIS INSTRUCTION				
6528				:-				
6529								
6530	021170			INLINE <CVTLN>				CVTLN
6531	021170	076057						
6532								
6533								
6534				:+				
6535				: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.				
6536				:-				
6537								
6538	021172			LET CC := PSW				MOV PSW,CC
6539	021172	013737	177776	000520				BIC #177760,CC
6540	021200			LET CC := CC CLR.BY #177760				CMP CC,CC.EXP
6541	021200	042137	177760	000520				BEQ \$50660
6542	021206			IF CC NE CC.EXP THEN				JSR PC,ERROR
6543	021206	023737	000520	000522				
6544	021214	001402						
6545	021216			CALL ERROR				
6546	021216	004737	060044					
6547	021222			ENDIF				
6548	021222							\$50660:
6549	021222			IF R0 NE R0.EXP THEN				CMP R0,R0.EXP
6550	021222	020037	000524					BEQ \$50661
6551	021226	001402						JSR PC,ERROR
6552	021230			CALL ERROR				
6553	021230	004737	060044					
6554	021234			ENDIF				
6555	021234							\$50661:
6556	021234			IF R1 NE R1.EXP THEN				CMP R1,R1.EXP
6557	021234	020137	000526					BEQ \$50662
6558	021240	001402						JSR PC,ERROR
6559	021242			CALL ERROR				
6560	021242	004737	060044					
6561	021246			ENDIF				

6562	021246								\$50662:
6563	021246			IF R2 NE R2.EXP THEN					CMP R2,R2.EXP
6564	021246	020237	000530						BEQ \$50663
6565	021252	001402							
6566	021254			CALL ERROR					JSR PC,ERROR
6567	021254	004737	060044						
6568	021260			ENDIF					\$50663:
6569	021260								CMP R3,R3.EXP
6570	021260			IF R3 NE R3.EXP THEN					BEQ \$50664
6571	021260	020337	000532						JSR PC,ERROR
6572	021264	001402							
6573	021266			CALL ERROR					\$50664:
6574	021266	004737	060044						CMP R4,R4.EXP
6575	021272			ENDIF					BEQ \$50665
6576	021272								JSR PC,ERROR
6577	021272			IF R4 NE R4.EXP THEN					
6578	021272	020437	000534						\$50665:
6579	021276	001402							CMP R5,R5.EXP
6580	021300			CALL ERROR					BEQ \$50666
6581	021300	004737	060044						JSR PC,ERROR
6582	021304			ENDIF					
6583	021304			IF R5 NE R5.EXP THEN					\$50665:
6584	021304								CMP R5,R5.EXP
6585	021304	020537	000536						BEQ \$50666
6586	021310	001402							JSR PC,ERROR
6587	021312			CALL ERROR					
6588	021312	004737	060044						\$50666:
6589	021316			ENDIF					
6590	021316								
6591									
6592				:+					
6593				: COMPARE THE EXPECTED DECIMAL STRING BUFFER					
6594				: WITH ACTUAL RESULTS.					
6595				:-					
6596									
6597	021316			LET R0 := #DS.DST					MOV #DS.DST,R0
6598	021316	012700	064400						
6599	021322			LET R1 := #DS.EXP					MOV #DS.EXP,R1
6600	021322	012701	064442						
6601	021326			LET ERRFLG := #0					CLR ERRFLG
6602	021326	005037	000456						
6603	021332			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO					\$50667:
6604	021332								CMP R0,#DS.DST+40
6605	021332	020027	064440						BEQ \$50670
6606	021336	001410							TST ERRFLG
6607	021340	005737	000456						BNE \$50670
6608	021344	001005							
6609	021346			IFB (R0)+ NE (R1)+ THEN					CMPB (R0)+,(R1)+
6610	021346	122021							BEQ \$50671
6611	021350	001402							JSR PC,ERROR
6612	021352			CALL ERROR					
6613	021352	004737	060044						\$50671:
6614	021356			ENDIF					BR \$50667
6615	021356								
6616	021356			ENDDO					
6617	021356	000765							

6618	021360				LET TSTCAS := TSTCAS + #6	\$50670:
6619	021360					ADD #6,TSTCAS
6620	021360	062737	000006	000544	LET COUNT := COUNT - #1	DEC COUNT
6621	021366					
6622	021366	005337	000470		:UNTIL COUNT EQ #0	TST COUNT
6623					IF COUNT NE #0 THEN	BEQ \$50672
6624	021372					JMP \$5057
6625	021372	005737	000470		INLINE <JMP \$5057>	
6626	021376	001402			ENDIF	\$50672:
6627	021400					
6628	021400	000137	020214			
6629	021404					
6630	021404					
6631						
6632					:+ DO ONE IN-LINE CASE - CVTLNI	
6633					:-	
6634						
6635					CALL CLRREG	JSR PC,CLRREG
6636	021404					
6637	021404	004737	056604		CALL CLRDSB	JSR PC,CLRDSB
6638	021410					
6639	021410	004737	057040		INLINE <CVTLNI>	CVTLNI
6640	021414				INLINE <.WORD TCVLNI>	.WORD TCVLNI
6641	021414	076157			INLINE <.WORD TCVLNI+4>	.WORD TCVLNI+4
6642	021416				LET CC := PSW	MOV PSW,CC
6643	021416	021716			LET CC := CC CLR.BY #177760	BIC #177760,CC
6644	021420				IF CC NE #0 THEN	TST CC
6645	021420	021722			CALL ERROR	BEQ \$50673
6646	021422				ENDIF	JSR PC,ERROR
6647	021422	013737	177776	000520	IF R0 NE #0 THEN	\$50673:
6648	021430					TST R0
6649	021430	042737	177760	000520	CALL ERROR	BEQ \$50674
6650	021436				ENDIF	JSR PC,ERROR
6651	021436	005737	000520		IF R1 NE #0 THEN	\$50674:
6652	021442	001402				TST R1
6653	021444				CALL ERROR	BEQ \$50675
6654	021444	004737	060044		ENDIF	JSR PC,ERROR
6655	021450				IF R2 NE #0 THEN	\$50675:
6656	021450					TST R2
6657	021450				CALL ERROR	BEQ \$50676
6658	021450	005700			ENDIF	
6659	021452	001402			IF R2 NE #0 THEN	
6660	021454					
6661	021454	004737	060044		CALL ERROR	
6662	021460				ENDIF	
6663	021460					
6664	021460					
6665	021460	005701				
6666	021462	001402				
6667	021464					
6668	021464	004737	060044			
6669	021470					
6670	021470					
6671	021470					
6672	021470	005702				
6673	021472	001402				

6674	021474			CALL ERROR					
6675	021474	004737	060044					JSR	PC,ERROR
6676	021500			ENDIF					
6677	021500							\$50676:	
6678	021500			IF R3 NE #0 THEN				TST	R3
6679	021500	005703						BEQ	\$50677
6680	021502	001402							
6681	021504			CALL ERROR				JSR	PC,ERROR
6682	021504	004737	060044						
6683	021510			ENDIF				\$50677:	
6684	021510							TST	R4
6685	021510			IF R4 NE #0 THEN				BEQ	\$50700
6686	021510	005704							
6687	021512	001402						JSR	PC,ERROR
6688	021514			CALL ERROR				\$50700:	
6689	021514	004737	060044					TST	R5
6690	021520			ENDIF				BEQ	\$50701
6691	021520							JSR	PC,ERROR
6692	021520			IF R5 NE #0 THEN				\$50701:	
6693	021520	005705						MOV	#DS.DST-1,R0
6694	021522	001402						CMPB	(R0)+,#53
6695	021524			CALL ERROR				BEQ	\$50702
6696	021524	004737	060044					JSR	PC,ERROR
6697	021530			ENDIF				\$50702:	
6698	021530			LET R0 := #DS.DST-1				CLR	ERRFLG
6699	021530							\$50703:	
6700	021530	012700	064377	IFB (R0)+ NE #53 THEN				CMPB	(R0)+,#61
6701	021534							BEQ	\$50704
6702	021534	122027	000053					JSR	PC,ERROR
6703	021540	001402		CALL ERROR				\$50704:	
6704	021542			ENDIF				CMP	R0,#DS.DST+12
6705	021542	004737	060044					BEQ	\$50705
6706	021546			LET ERRFLG := #0				TST	ERRFLG
6707	021546							BEQ	\$50703
6708	021546			REPEAT				\$50705:	
6709	021546	005037	000456					JMP	XCVTPL
6710	021552			IFB (R0)+ NE #61 THEN					
6711	021552								
6712	021552								
6713	021552	122027	000061						
6714	021556	001402		CALL ERROR					
6715	021560			ENDIF					
6716	021560	004737	060044						
6717	021564			UNTIL R0 EQ #DS.DST+12 OR ERRFLG NE #0					
6718	021564								
6719	021564								
6720	021564	020027	064412						
6721	021570	001403		INLINE <JMP XCVTPL>					
6722	021572	005737	000456						
6723	021576	001765							
6724	021600								
6725	021600								
6726	021600	000137	021726						
6727									
6728									
6729									

:+

```

6730          ; TABLE OF TEST CASES FOR CVTLN
6731          ; -
6732
6733 021604 000014      TCVTLN: .WORD 12.          ;# OF TEST CASES FOR CVTLN
6734
6735 021606 000005      .WORD 5          ;DST. DATA TYPE AND LENGTH
6736 021610 000000      .WORD 0          ;LONG INTEGER HIGH & SIGN
6737 021612 000000      .WORD 0          ;LONG INTEGER LOW
6738
6739 021614 050006      .WORD 50006       ;TEST CASE NUMBER 2
6740 021616 000000      .WORD 0
6741 021620 000000      .WORD 0
6742
6743 021622 020010      .WORD 20010       ;TEST CASE NUMBER 3
6744 021624 000000      .WORD 0
6745 021626 001751      .WORD 1751
6746
6747 021630 030000      .WORD 30000       ;TEST CASE NUMBER 4
6748 021632 000000      .WORD 0
6749 021634 152361      .WORD 152361
6750
6751 021636 040000      .WORD 40000       ;TEST CASE NUMBER 5
6752 021640 000000      .WORD 0
6753 021642 024137      .WORD 24137
6754
6755 021644 050003      .WORD 50003       ;TEST CASE NUMBER 6
6756 021646 000000      .WORD 0
6757 021650 075147      .WORD 75147
6758
6759 021652 020004      .WORD 20004       ;TEST CASE NUMBER 7
6760 021654 000000      .WORD 0
6761 021656 075147      .WORD 75147
6762
6763 021660 000012      .WORD 12          ;TEST CASE NUMBER 8
6764 021662 077777      .WORD 77777
6765 021664 177177      .WORD 177777
6766
6767 021666 000037      .WORD 37          ;TEST CASE NUMBER 9
6768 021670 100000      .WORD 100000
6769 021672 000000      .WORD 0
6770
6771 021674 010021      .WORD 10021       ;TEST CASE NUMBER 10
6772 021676 100000      .WORD 100000
6773 021700 000000      .WORD 0
6774
6775 021702 040012      .WORD 40012       ;TEST CASE NUMBER 11
6776 021704 074377      .WORD 74377
6777 021706 171700      .WORD 171700
6778
6779 021710 030000      .WORD 30000       ;TEST CASE NUMBER 12
6780 021712 000000      .WORD 0
6781 021714 000000      .WORD 0
6782
6783
6784 021716          TCVLNI: .WORD 50012       ;IN-LINE TEST CASE
6785 021716 050012      .WORD 50012       ;DST TYPE = LEADING SEPARATE, LENGTH = 12
    
```


6786 021720 064400
6787 021722 032707
6788 021724 041072
6789

.WORD DS.DST
.WORD 32707
.WORD 41072

:DST.ADDRESS
:LONG INTEGER LOW WORD +
: LONG INTEGER HIGH WORD AND SIGN
: = 1111111111 IN BCD

```

6790          .SBTTL CVTPL
6791          :*****
6792          :
6793          :   CVTPL TEST
6794          :
6795          :*****
6796 021726    INLINE <XCVTPL:>
6797 021726                                XCVTPL:
6798 021726    INLINE <SCOPE>
6799 021726    000004                                SCOPE
6800 021730    LET $TIMES := #20                                ;ITERATION COUNT
6801 021730    012737 000020 000404                        MOV      #20,$TIMES
6802 021736    LET COUNT := TCVTPL                                ;GET TEST COUNT
6803 021736    013737 022750 000470                        MOV      TCVTPL,COUNT
6804 021744    LET TSTCAS := #TCVTPL+2                        ;POINT TO FIRST TEST CASE
6805 021744    012737 022752 000544                        MOV      #TCVTPL+2,TSTCAS
6806          ;REPEAT
6807 021752    INLINE <$5073:>
6808 021752                                $5073:
6809
6810          ;+
6811          ; GET TEST DATA FROM TABLE
6812          ;-
6813
6814 021752    LET R0 := TSTCAS
6815 021752    013700 000544                                MOV      TSTCAS,R0
6816 021756    LET DATA00 := (R0)+                                ;SRC. DATA TYPE AND LENGTH
6817 021756    012037 000562                                MOV      (R0)+,DATA00
6818 021762    LET DATA01 :- (R0)                                ;SRC. POINTER
6819 021762    011037 000564                                MOV      (R0),DATA01
6820
6821          ;+
6822          ; SET UP R0.EXP, R1.EXP, R4.EXP, R5.EXP, R4, AND R5.
6823          ;-
6824
6825 021766    LET R0.EXP := #0
6826 021766    005037 000524                                CLR      R0.EXP
6827 021772    LET R1.EXP := #0
6828 021772    005037 000526                                CLR      R1.EXP
6829 021776    IF #1 SETIN COUNT THEN
6830 021776    032737 000001 000470                                BIT      #1,COUNT
6831 022004    001405                                BEQ      $50706
6832          LET R4 := #177777
6833 022006    012704 177777                                MOV      #177777,R4
6834 022012    LET R5 : #177777
6835 022012    012705 177777                                MOV      #177777,R5
6836 022016    ELSE
6837 022016    000402                                BR       $50707
6838 022020    $50706:
6839 022020    LET R4 := #0
6840 022020    005004                                CLR      R4
6841 022022    LET R5 := #0
6842 022022    005005                                CLR      R5
6843 022024    ENDIF
6844 022024                                $50707:
6845 022024    LET R4.EXP :- R4
    
```

6846	022024	010437	000534			MOV	R4,R4.EXP
6847	022030				LET R5.EXP := R5		
6848	022030	010537	000536			MOV	R5,R5.EXP
6849							
6850					:+		
6851					:-		
6852					CONVERT PACKED NUMBER AND LOAD INTO R2.EXP AND R3.EXP		
6853							
6854	022034				CALL CLRDSB		
6855	022034	004737	057040			JSR	PC,CLRDSB
6856	022040				LET CC.EXP := #0		
6857	022040	005037	000522			CLR	CC.EXP
6858	022044				LET R0 := #DSBUF1	;DST.POINTER	
6859	022044	012700	064274			MOV	#DSBUF1,R0
6860	022050				LET R1 := DATA01	;SRC.POINTER	
6861	022050	013701	000564			MOV	DATA01,R1
6862	022054				LET DSLEN1 := DATA00 CLR.BY #177740	;GET SRC.LEN	
6863	022054	013737	000562	000514		MOV	DATA00,DSLEN1
6864	022062	042737	177740	000514		BIC	#177740,DSLEN1
6865	022070				IF DSLEN1 EQ #0 THEN	;SRC.LEN=0	
6866	022070	005737	000514			TST	DSLEN1
6867	022074	001010				BNE	\$50710
6868	022076				LET CC.EXP := CC.EXP SET.BY #ZBIT		
6869	022076	052737	000004	000522		BIS	#ZBIT,CC.EXP
6870	022104				LET R2.EXP := #0		
6871	022104	005037	000530			CLR	R2.EXP
6872	022110				LET R3.EXP := #0		
6873	022110	005037	000532			CLR	R3.EXP
6874	022114				ELSE		
6875	022114	000532				BR	\$50711
6876	022116					\$50710:	
6877	022116				LET R0 := #DSBUF1 + #10. - DSLEN1 ;DEST. FOR INTERMEDIATE RESULTS	MOV	#DSBUF1,R0
6878	022116	012700	064274			ADD	#10.,R0
6879	022122	062700	000012			SUB	DSLEN1,R0
6880	022126	163700	000514				
6881					;LOAD UP DSBUF1 BUFFER WITH INTERMEDIATE RESULTS		
6882	022132				CALL CLRDSB	;CLEAR DECIMAL STRING BUFFERS	
6883	022132	004737	057040			JSR	PC,CLRDSB
6884	022136				IF #1 NOTSET IN DSLEN1 THEN		
6885	022136	032737	000001	000514		BIT	#1,DSLEN1
6886	022144	001005				BNE	\$50712
6887	022146				LET (R1) :B= (R1) CLR.BY #360 ;CLEAR UPPER NIBBLE		
6888	022146	142711	000360			BICB	#360,(R1)
6889	022152				LET (R0)+ :B= (R1)+		
6890	022152	112120				MOVB	(R1)+,(R0)+
6891	022154				LET DSLEN1 := DSLEN1 - #1		
6892	022154	005337	000514			DEC	DSLEN1
6893	022160				ENDIF		
6894	022160					\$50712:	
6895	022160				WHILE DSLEN1 HI #1 DO		
6896	022160					\$50713:	
6897	022160	023727	000514	000001		CMP	DSLEN1,#1
6898	022166	101420				BLOS	\$50714
6899	022170				LET (R0) :B= (R1)		
6900	022170	111110				MOVB	(R1),(R0)
6901	022172				INLINE <ASRB (R0)>		

6902	022172	106210					ASRB (R0)
6903	022174					INLINE <ASRB (R0)>	
6904	022174	106210					ASRB (R0)
6905	022176					INLINE <ASRB (R0)>	
6906	022176	106210					ASRB (R0)
6907	022200					INLINE <ASRB (R0)>	
6908	022200	106210					ASRB (R0)
6909	022202					LET (R0) :B= (R0) CLR.BY #360	; CLEAR UPPER NIBBLE
6910	022202	142710	000360				BICB #360,(R0)
6911	022206					LET R0 := R0 + #1	
6912	022206	005200					INC R0
6913	022210					LET (R0) :B= (R1)+ CLR.BY #360	
6914	022210	112110					MOVB (R1)+,(R0)
6915	022212	142710	000360				BICB #360,(R0)
6916	022216					LET R0 := R0 + #1	
6917	022216	005200					INC R0
6918	022220					LET DSLEN1 := DSLEN1 - #2	
6919	022220	162737	000002	000514			SUB #2,DSLEN1
6920	022226					ENDDO	
6921	022226	000754					BR \$50713
6922	022230						\$50714:
6923	022230					LET (R0) :B= (R1)	; GET THE LSD
6924	022230	111110					MOVB (R1),(R0)
6925	022232					INLINE <ASRB (R0)>	
6926	022232	106210					ASRB (R0)
6927	022234					INLINE <ASRB (R0)>	
6928	022234	106210					ASRB (R0)
6929	022236					INLINE <ASRB (R0)>	
6930	022236	106210					ASRB (R0)
6931	022240					INLINE <ASRB (R0)>	
6932	022240	106210					ASRB (R0)
6933	022242					LET (R0) :B= (R0) CLR.BY #360	
6934	022242	142710	000360				BICB #360,(R0)
6935						; LOAD UP R2.EXP AND R3.EXP WITH FINAL RESULT	
6936	022246					CALL CVTIL	
6937	022246	004737	032020				JSR PC,CVTIL
6938	022252					LET DATTYP := DATA00 CLR.BY #107777	
6939	022252	013737	000562	000464			MOV DATA00,DATTYP
6940	022260	042737	107777	000464			BIC #107777,DATTYP
6941	022266					IF DATTYP EQ #60000 THEN	; SIGNED PACKED
6942	022266	023727	000464	060000			CMP DATTYP,#60000
6943	022274	001042					BNE \$50715
6944	022276					LET TEMP5 :B= (R1) CLR.BY #360	; ISOLATE THE SIGN
6945	022276	111137	000510				MOVB (R1),TEMP5
6946	022302	142737	000360	000510			BICB #360,TEMP5
6947	022310					IFB TEMP5 EQ #15 OR TEMP5 EQ #13 THEN	
6948	022310	123727	000510	000015			CMPB TEMP5,#15
6949	022316	001404					BEQ \$50716
6950	022320	023727	000510	000013			CMP TEMP5,#13
6951	022326	001025					BNE \$50717
6952	022330						\$50716:
6953	022330					LET CC.EXP :- CC.EXP SET.BY #NBIT	
6954	022330	052737	000010	000522			BIS #NBIT,CC.EXP
6955	022336					LET R3.EXP := NEGATE R3.EXP	
6956	022336	005437	000532				NEG R3.EXP
6957	022342					LET R2.EXP :- COMP R2.EXP	

6958	022342	005137	000530		COM	R2.EXP
6959	022346			IF R3.EXP EQ #0 THEN	TST	R3.EXP
6960	022346	005737	000532		BNE	\$50720
6961	022352	001002		LET R2.EXP := R2.EXP + #1	INC	R2.EXP
6962	022354			ENDIF	\$50720:	
6963	022354	005237	000530		TST	R2.EXP
6964	022360			IF R2.EXP NE #0 OR R3.EXP NE #0 THEN	BNE	\$50721
6965	022360				TST	R3.EXP
6966	022360	005737	000530		BEQ	\$50722
6967	022360	001003			\$50721:	
6968	022364	005737	000532	LET CC.EXP := CC.EXP SET.BY #CBIT	BIS	#CBIT,CC.EXP
6969	022366	001403		ENDIF	\$50722:	
6970	022372			ENDIF	\$50717:	
6971	022374			ENDIF	\$50715:	
6972	022374	052737	000001 000522	ENDIF	\$50711:	
6973	022402			ENDIF		
6974	022402			ENDIF		
6975	022402			ENDIF		
6976	022402			ENDIF		
6977	022402			ENDIF		
6978	022402			ENDIF		
6979	022402			ENDIF		
6980	022402			ENDIF		
6981	022402			ENDIF		
6982						
6983				:+		
6984				: SET UP R0 THRU R3		
6985				:-		
6986						
6987	022402			LET R0 := DATA00 ;SRC.DSCR	MOV	DATA00,R0
6988	022402	013700	000562		MOV	DATA01,R1
6989	022406			LET R1 := DATA01 ;SRC.DSCR	MOV	DATA01,R1
6990	022406	013701	000564		CLR	R2
6991	022412			LET R2 := #0	CLR	R3
6992	022412	005002				
6993	022414			LET R3 := #0		
6994	022414	005003				
6995						
6996				:+		
6997				: DO THE CIS INSTRUCTION		
6998				:-		
6999						
7000	022416			INLINE <CVTPL>	CVTPL	
7001	022416	076073				
7002						
7003						
7004				:+		
7005				: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
7006				:-		
7007						
7008	022420			LET CC := PSW	MOV	PSW,CC
7009	022420	013737	177776 000520		BIC	#177760,CC
7010	022426			LET CC := CC CLR.BY #177760		
7011	022426	042737	177760 000520		CMP	CC,CC.EXP
7012	022434			IF CC NE CC.EXP THEN		
7013	022434	023737	000520 000522			

7014	022442	001402			BEQ	\$50723
7015	022444			CALL ERROR	JSR	PC,ERROR
7016	022444	004737	060044			
7017	022450			ENDIF		\$50723:
7018	022450					
7019	022450			IF R0 NE R0.EXP THEN		
7020	022450	020037	000524		CMP	R0,R0.EXP
7021	022454	001402			BEQ	\$50724
7022	022456			CALL ERROR		
7023	022456	004737	060044		JSR	PC,ERROR
7024	022462			ENDIF		\$50724:
7025	022462					
7026	022462			IF R1 NE R1.EXP THEN		
7027	022462	020137	000526		CMP	R1,R1.EXP
7028	022466	001402			BEQ	\$50725
7029	022470			CALL ERROR		
7030	022470	004737	060044		JSR	PC,ERROR
7031	022474			ENDIF		\$50725:
7032	022474					
7033	022474			IF R2 NE R2.EXP THEN		
7034	022474	020237	000530		CMP	R2,R2.EXP
7035	022500	001402			BEQ	\$50726
7036	022502			CALL ERROR		
7037	022502	004737	060044		JSR	PC,ERROR
7038	022506			ENDIF		\$50726:
7039	022506					
7040	022506			IF R3 NE R3.EXP THEN		
7041	022506	020337	000532		CMP	R3,R3.EXP
7042	022512	001402			BEQ	\$50727
7043	022514			CALL ERROR		
7044	022514	004737	060044		JSR	PC,ERROR
7045	022520			ENDIF		\$50727:
7046	022520					
7047	022520			IF R4 NE R4.EXP THEN		
7048	022520	020437	000534		CMP	R4,R4.EXP
7049	022524	001402			BEQ	\$50730
7050	022526			CALL ERROR		
7051	022526	004737	060044		JSR	PC,ERROR
7052	022532			ENDIF		\$50730:
7053	022532					
7054	022532			IF R5 NE R5.EXP THEN		
7055	022532	020537	000536		CMP	R5,R5.EXP
7056	022536	001402			BEQ	\$50731
7057	022540			CALL ERROR		
7058	022540	004737	060044		JSR	PC,ERROR
7059	022544			ENDIF		\$50731:
7060	022544					
7061						
7062	022544			LET TSTCAS := TSTCAS + #4		
7063	022544	062737	000064 000544		ADD	#4,TSTCAS
7064	022552			LET COUNT := COUNT - #1		
7065	022552	005337	000470		DEC	COUNT
7066				:UNTIL COUNT EQ #0		
7067	022556			IF COUNT NE #0 THEN		
7068	022556	005737	000470		TST	COUNT
7069	022562	001402			BEQ	\$50732

7070	022564				INLINE <JMP \$5073>		
7071	022564	000137	021752				JMP \$5073
7072	022570				ENDIF		
7073	022570						\$50732:
7074					::+		
7075					:: DO ONE INLINE CASE - CVTPLI		
7076					::-		
7077							
7078	022570				CALL CLRREG		
7079	022570	004737	056604				JSR PC,CLRREG
7080	022574				CALL CLRDSB		
7081	022574	004737	057040				JSR PC,CLRDSB
7082	022600				INLINE <CVTPLI>		
7083	022600	076173					CVTPLI
7084	022602				INLINE <.WORD TCVPLI>		
7085	022602	023026					.WORD TCVPLI
7086	022604				INLINE <.WORD TCVPLI+4>		
7087	022604	023032					.WORD TCVPLI+4
7088	022606				LET CC := PSW		
7089	022606	013737	177776	000520			MOV PSW,CC
7090	022614				LET CC := CC CLR.BY #177760		
7091	022614	042737	177760	000520			BIC #177760,CC
7092	022622				IF CC NE #0 THEN		
7093	022622	005737	000520				TST CC
7094	022626	001402					BEQ \$50733
7095	022630				CALL ERROR		
7096	022630	004737	060044				JSR PC,ERROR
7097	022634				ENDIF		
7098	022634						\$50733:
7099	022634				IF R0 NE #0 THEN		
7100	022634	005700					TST R0
7101	022636	001402					BEQ \$50734
7102	022640				CALL ERROR		
7103	022640	004737	060044				JSR PC,ERROR
7104	022644				ENDIF		
7105	022644						\$50734:
7106	022644				IF R1 NE #0 THEN		
7107	022644	005701					TST R1
7108	022646	001402					BEQ \$50735
7109	022650				CALL ERROR		
7110	022650	004737	060044				JSR PC,ERROR
7111	022654				ENDIF		
7112	022654						\$50735:
7113	022654				IF R2 NE #0 THEN		
7114	022654	005702					TST R2
7115	022656	001402					BEQ \$50736
7116	022660				CALL ERROR		
7117	022660	004737	060044				JSR PC,ERROR
7118	022664				ENDIF		
7119	022664						\$50736:
7120	022664				IF R3 NE #0 THEN		
7121	022664	005703					TST R3
7122	022666	001402					BEQ \$50737
7123	022670				CALL ERROR		
7124	022670	004737	060044				JSR PC,ERROR
7125	022674				ENDIF		

7126 022674
 7127 022674
 7128 022674 005704
 7129 022676 001402
 7130 022700
 7131 022700 004737 060044
 7132 022704
 7133 022704
 7134 022704
 7135 022704 005705
 7136 022706 001402
 7137 022710
 7138 022710 004737 060044
 7139 022714
 7140 022714
 7141 022714
 7142 022714 023727 023032 032707
 7143 022722 001402
 7144 022724
 7145 022724 004737 060044
 7146 022730
 7147 022730
 7148 022730
 7149 022730 023727 023034 041072
 7150 022736 001402
 7151 022740
 7152 022740 004737 060044
 7153 022744
 7154 022744
 7155 022744
 7156 022744 000137 023046
 7157
 7158
 7159
 7160
 7161
 7162
 7163
 7164 022750 000006
 7165
 7166 022752 060000
 7167 022754 023002
 7168
 7169 022756 060004
 7170 022760 023003
 7171
 7172 022762 060004
 7173 022764 023006
 7174
 7175 022766 060012
 7176 022770 023011
 7177
 7178 022772 070007
 7179 022774 023017
 7180
 7181 022776 060005

```

IF R4 NE #0 THEN
    CALL ERROR
ENDIF
IF R5 NE #0 THEN
    CALL ERROR
ENDIF
IF TCVPLI+4 NE #32707 THEN
    CALL ERROR
ENDIF
IF TCVPLI+6 NE #41072 THEN
    CALL ERROR
ENDIF
INLINE <JMP XCVTNL>
  
```

```

$50737:
TST R4
BEQ $50740
JSR PC,ERROR

$50740:
TST R5
BEQ $50741
JSR PC,ERROR

$50741:
CMP TCVPLI+4,#32707
BEQ $50742
JSR PC,ERROR

$50742:
CMP TCVPLI+6,#41072
BEQ $50743
JSR PC,ERROR

$50743:
JMP XCVTNL
  
```

::+
 :: TABLE OF TEST CASES FOR CVTPL
 ::-

```

TCVTPL: .WORD 6 ;# OF TEST CASES FOR CVTPL
        .WORD 60000 ;SRC. DATA TYPE AND LENGTH
        .WORD CVTPL1 ;POINTER TO PACKED DECIMAL STRING
        .WORD 60004 ;TEST CASE NUMBER 2
        .WORD CVTPL2
        .WORD 60004 ;TEST CASE NUMBER 3
        .WORD CVTPL3
        .WORD 60012 ;TEST CASE NUMBER 4
        .WORD CVTPL4
        .WORD 70007 ;TEST CASE NUMBER 5
        .WORD CVTPL5
        .WORD 60005 ;TEST CASE NUMBER 6
  
```



```

7182 023000 023023 .WORD CVTPL6
7183
7184
7185 023002 014 CVTPL1: .BYTE 014 ;PACKED DECIMAL STRING 1
7186
7187 023003 001 043 114 CVTPL2: .BYTE 001,043,114 ;PACKED DECIMAL STRING 2
7188
7189 023006 001 043 115 CVTPL3: .BYTE 001,043,115 ;PACKED DECIMAL STRING 3
7190
7191 023011 002 024 164 CVTPL4: .BYTE 002,024,164,203,144,174 ;PACKED DECIMAL STRING 4
7192 023014 203 144 174
7193
7194 023017 022 000 007 CVTPL5: .BYTE 022,000,007,055 ;PACKED DECIMAL STRING 5
7195 023022 055
7196
7197 023023 061 040 174 CVTPL6: .BYTE 061,040,174 ;PACKED DECIMAL STRING 6
7198
7199
7200
7201 .EVEN
7202
7203 023026 TCVFLI: ;INLINE TEST CASE
7204 023026 060012 .WORD 60012 ;SRC. DATA TYPE = SIGNED PACKED, LENGTH 12
7205 023030 023036 .WORD PLI ;POINTER TO SRC. STRING
7206 023032 000000 .WORD 0 ;LONG INTEGER RESULT, LOW
7207 023034 000000 .WORD 0 ;LONG INTEGER RESULT, HI, AND SIGN
7208
7209 023036 001 021 021 PLI: .BYTE 1,21,21,21,21,21,34 ;INLINE PACKED DECIMAL SRC. STRING
7210 023041 021 021 021
7211 023044 034 ; = -1111111111 IN BCD
7212
7213
7214 023046 .EVEN
    
```

```

7215 .SBTTL CVTNL
7216 :*****
7217 :
7218 : CVTNL TEST
7219 :
7220 :*****
7221 023046 INLINE <XCVTNL:>
7222 023046 XCVTNL:
7223 023046 INLINE <SCOPE> SCOPE
7224 023046 000004
7225 023050 LET $TIMES := #20 ;ITERATION COUNT
7226 023050 012737 000020 000404 MOV #20,$TIMES
7227 023056 LET COUNT := ^CVTNL ;GET TEST COUNT
7228 023056 013737 024330 000470 MOV TCVTNL,COUNT
7229 023064 LET TSTCAS := #TCVTNL+2 ;POINT TO FIRST TEST CASE
7230 023064 012737 024332 000544 MOV #TCVTNL+2,TSTCAS
7231 :REPEAT
7232 023072 INLINE <$5053:>
7233 023072 $5053:
7234
7235 :+
7236 : GET TEST DATA FROM TABLE
7237 :-
7238
7239 023072 LET R0 := TSTCAS
7240 023072 013700 000544 MOV TSTCAS,R0
7241 023076 LET DATA00 := (R0)+ ;SRC. DATA TYPE AND LENGTH
7242 023076 012037 000562 MOV (R0)+,DATA00
7243 023102 LET DATA01 := (R0) ;SRC. POINTER
7244 023102 011037 000564 MOV (R0),DATA01
7245
7246 :+
7247 : SET UP R0.EXP, R1.EXP, R4.EXP, R5.EXP, R4, AND R5.
7248 :-
7249
7250 023106 LET R0.EXP := #0
7251 023106 005037 000524 CLR R0.EXP
7252 023112 LET R1.EXP := #0
7253 023112 005037 000526 CLR R1.EXP
7254 023116 IF #1 SETIN COUNT THEN
7255 023116 032737 000001 000470 BIT #1,COUNT
7256 023124 001405 BEQ $50744
7257 023126 LET R4 := #177777
7258 023126 012704 177777 MOV #177777,R4
7259 023132 LET R5 := #177777
7260 023132 012705 177777 MOV #177777,R5
7261 023136 ELSE
7262 023136 000402 BR $50745
7263 023140 $50744:
7264 023140 LET R4 := #0
7265 023140 005004 CLR R4
7266 023142 LET R5 := #0
7267 023142 005005 CLR R5
7268 023144 ENDIF
7269 023144 $50745:
7270 023144 LET R4.EXP := R4
    
```

```

7271 023144 010437 000534          MOV      R4,R4.EXP
7272 023150          LET R5.EXP := R5
7273 023150 010537 000536          MOV      R5,R5.EXP
7274
7275
7276          ;+
          ; CONVERT NUMERIC DECIMAL STRING AND LOAD INTO R2.EXP AND R3.EXP
          ; -
7277
7278
7279 023154          LET CC.EXP := #0
7280 023154 005037 000522          CLR      CC.EXP
7281 023160          LET R0 := #DSBUF1          ;DST. POINTER
7282 023160 012700 064274          MOV      #DSBUF1,R0
7283 023164          LET R1 := DATA01          ;SRC. POINTER
7284 023164 013701 000564          MOV      DATA01,R1
7285 023170          LET SIGNC := #0
7286 023170 005037 000450          CLR      SIGNO
7287 023174          LET DSLEN1 := DATA00 CLR.BY #177740          ;GET SRC.LEN
7288 023174 013737 000562 000514          MOV      DATA00,DSLEN1
7289 023202 042737 177740 000514          BIC      #177740,DSLEN1
7290 023210          IF DSLEN1 EQ #0 THEN          ;SRC.LEN = 0
7291 023210 005737 000514          TST      DSLEN1
7292 023214 001010          BNE      $50746
7293 023216          LET CC.EXP := CC.EXP SET.BY #ZBIT
7294 023216 052737 000004 000522          BIS      #ZBIT,CC.EXP
7295 023224          LET R2.EXP := #0
7296 023224 005037 000530          CLR      R2.EXP
7297 023230          LET R3.EXP := #0
7298 023230 005037 000532          CLR      R3.EXP
7299 023234          ELSE          ;SRC.LEN > 0
7300 023234 000417          BR      $50746
7301 023236          LET R0 := #DSBUF1 + #10. - DSLEN1
7302 023236
7303 023236 012700 064274          MOV      #DSBUF1,R0
7304 023242 062700 000012          ADD      #10.,R0
7305 023246 163700 000514          SUB      DSLEN1,R0
7306 023252          LET TEMPO := DSLEN1
7307 023252 013737 000514 000476          MOV      DSLEN1,TEMPO
7308 023260          REPEAT
7309 023260          LET (R0)+ := B- (R1)+          ;XFER TO INTERMEDIATE
7310 023260          MOV      (R1)+,(R0)+
7311 023260 112120          LET TEMPO := TEMPO - #1
7312 023262          DEC      TEMPO
7313 023262 005337 000476          UNTIL TEMPO EQ #0
7314 023266          TST      TEMPO
7315 023266 005737 000476          BNE      $50750
7316 023272 001372          ENDIF
7317 023274          LET DATTYP := DATA00 CLR.BY #107777
7318 023274
7319 023274          ;GET THE SIGN OUT OF THE INTERMEDIATE RESULTS
7320 023274 013737 000562 000464          IF DSLEN1 NE #0 THEN          ;SRC.LEN > 0
7321 023302 042737 107777 000464          TST      DSLEN1
7322          BEQ      $50751
7323 023310          IF DATTYP EQ #0 OR DATTYP EQ #10000 THEN
7324 023310 005737 000514          ;*** ZONED ***
7325 023314 001533
7326 023316
    
```

```

7327 023316 005737 000464
7328 023322 001404
7329 023324 023727 000464 010000
7330 023332 001020
7331 023334
7332 023334
7333 023334 005300
7334 023336
7335 023336 111037 000476
7336 023342 142737 000017 000476
7337 023350
7338 023350 123727 000476 000160
7339 023356 001006
7340 023360
7341 023360 012737 177777 000450
7342 023366
7343 023366 052737 000010 000522
7344 023374
7345 023374
7346 023374
7347 023374
7348 023374
7349 023374 023727 000464 020000
7350 023402 001404
7351 023404 023727 000464 030000
7352 023412 001044
7353 023414
7354 023414
7355 023414 023727 000464 020000
7356 023422 001002
7357 023424
7358 023424 005300
7359 023426
7360 023426 000406
7361 023430
7362 023430
7363 023430 012700 064274
7364 023434 062700 000012
7365 023440 163700 000514
7366 023444
7367 023444
7368 023444
7369 023444 111037 000476
7370 023450
7371 023450 012701 024222
7372 023454
7373 023454
7374 023454
7375 023454 112137 000500
7376 023460
7377 023460 005201
7378 023462
7379 023462 023737 000476 000500
7380 023470 001371
7381 023472
7382 023472 005301

```

```

LET R0 := R0 - #1
LET TEMPO :B= (R0) CLR.BY #17
IFB TEMPO EQ #160 THEN
LET SIGNO := #-1
LET CC.EXP := CC.EXP SET.BY #NBIT
ENDIF
ENDIF
IF DATTYP EQ #20000 OR DATTYP EQ #30000 THEN
IF DATTYP EQ #20000 THEN
LET R0 := R0 - #1
ELSE
LET R0 := #DSBUF1 + #10. - DSLEN1
ENDIF
LET TEMPO :B= (R0)
LET R1 := #SGNDGT
REPEAT
LET TEMP1 :B= (R1)+
LET R1 := R1 + #1
UNTIL TEMPO EQ TEMP1
LET R1 := R1 - #1

```

```

TST DATTYP
BEQ $50752
CMP DATTYP,#10000
BNE $50753
$50752:
;POINT TO LAST BYTE
DEC R0
MOVB (R0),TEMPO
BICB #17,TEMPO
;NEGATIVE
CMPB TEMPO,#160
BNE $50754
MOV #-1,SIGNO
BIS #NBIT,CC.EXP
$50754:
$50753:
;*** OVERPUNCH ***
CMP DATTYP,#20000
BEQ $50755
CMP DATTYP,#30000
BNE $50756
$50755:
;TRAILING
CMP DATTYP,#20000
BNE $50757
DEC R0
;LEADING
BR $50760
$50757:
MOV #DSBUF1,R0
ADD #10.,R0
SUB DSLEN1,R0
$50760:
;GET SIGN/DIGIT
MOVB (R0),TEMPO
;TABLE ADDRESS
MOV #SGNDGT,R1
$50761:
MOVB (R1)+,TEMP1
;SKIP OVER ONE ENTRY
INC R1
CMP TEMPO,TEMP1
BNE $50761
;BACK UP TO CORRECT ENTRY
DEC R1

```

7383	023474				IFB #BIT07 SETIN (R1) THEN		:NEGATIVE
7384	023474	132711	000200				BITB #BIT07,(R1)
7385	023500	001406					BEQ \$50762
7386	023502				LET CC.EXP := CC.EXP SET.BY #NBIT		
7387	023502	052737	000010	000522			BIS #NBIT,CC.EXP
7388	023510				LET SIGNO := #-1		
7389	023510	012737	177777	000450			MOV #-1,SIGNO
7390	023516				ENDIF		
7391	023516						\$50762:
7392	023516				LET (R0) :B= (R1) CLR.BY #BIT07		:PUT CORRECT DIGIT INTO INTERMEDIATE RES
7393	023516	111110					MOVB (R1),(R0)
7394	023520	142710	000200				BICB #BIT07,(R0)
7395	023524				ENDIF		
7396	023524						\$50756:
7397	023524				IF DATTYP EQ #50000 THEN		:LEADING SEPARATE
7398	023524	023727	000464	050000			CMP DATTYP,#50000
7399	023532	001003					BNE \$50763
7400	023534				LET R1 := DATA01 - #1		:POINT TO SIGN
7401	023534	013701	000564				MOV DATA01,R1
7402	023540	005301					DEC R1
7403	023542				ENDIF		
7404	023542						\$50763:
7405	023542				IF DATTYP EQ #50000 OR DATTYP EQ #40000 THEN		
7406	023542	023727	000464	050000			CMP DATTYP,#50000
7407	023550	001404					BEQ \$50764
7408	023552	023727	000464	040000			CMP DATTYP,#40000
7409	023560	001011					BNE \$50765
7410	023562						\$50764:
7411	023562				IFB (R1) EQ #55 THEN		:NEGATIVE
7412	023562	121127	000055				CMPB (R1),#55
7413	023566	001006					BNE \$50766
7414	023570				LET SIGNO : #-1		
7415	023570	012737	177777	000450			MOV #-1,SIGNO
7416	023576				LET CC.EXP := CC.EXP SET.BY #NBIT		
7417	023576	052737	000010	000522			BIS #NBIT,CC.EXP
7418	023604				ENDIF		
7419	023604						\$50766:
7420	023604				ENDIF		
7421	023604						\$50765:
7422	023604				ENDIF		
7423	023604						\$50751:
7424					:CLEAR OUT UPPER NIBBLE OF INTERMEDIATE RESULTS		
7425	023604				INCR R0 FROM #DSBUF1 TO #DSBUF1+37 BY #1		
7426	023604	012700	064274				MOV #DSBUF1,R0
7427	023610	000401					BR \$50767
7428	023612						\$50770:
7429	023612	005200					INC R0
7430	023614						\$50767:
7431	023614	020027	064333				CMP R0,#DSBUF1+37
7432	023620	003003					BGT \$50771
7433	023622				LET (R0) :B (R0) CLR.BY #360		
7434	023622	142710	000360				BICB #360,(R0)
7435	023626				ENDINC		
7436	023626	000771					BR \$50770
7437	023630						\$50771:
7438					:LOAD UP R2.EXP AND R3.EXP WITH FINAL RESULT		

7439	023630				CALL CVTIL		
7440	023630	004737	032020			JSR	PC, CVTIL
7441	023634				IF SIGNO EQ #-1 THEN		
7442	023634	023727	000450	177777		CMP	SIGNO, #-1
7443	023642	001014				BNE	\$50772
7444	023644				LET CC.EXP := CC.EXP SET.BY #NBIT		
7445	023644	052737	000010	000522		BIS	#NBIT, CC.EXP
7446	023652				IF R2.EXP NE #0 OR R3.EXP NE #0 THEN		
7447	023652	005737	000530			TST	R2.EXP
7448	023656	001003				BNE	\$50773
7449	023660	005737	000532			TST	R3.EXP
7450	023664	001403				BEQ	\$50774
7451	023666						\$50773:
7452	023666				LET CC.EXP := CC.EXP SET.BY #CBIT		
7453	023666	052737	000001	000522		BIS	#CBIT, CC.EXP
7454	023674				ENDIF		
7455	023674						\$50774:
7456	023674				ENDIF		
7457	023674						\$50772:
7458							
7459					::+ :: SET UP R0 THRU R3 ::-		
7460							
7461							
7462							
7463	023674				LET R0 := DATA00 ;SRC.DSCR		
7464	023674	013700	000562			MOV	DATA00, R0
7465	023700				LET R1 := DATA01 ;SRC.DSCR		
7466	023700	013701	000564			MOV	DATA01, R1
7467	023704				LET R2 := #0		
7468	023704	005002				CLR	R2
7469	023706				LET R3 := #0		
7470	023706	005003				CLR	R3
7471							
7472					::+ :: DO THE CIS INSTRUCTION ::-		
7473							
7474							
7475							
7476	023710				INLINE <CVTNL>		
7477	023710	076053					CVTNL
7478							
7479							
7480					::+ :: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS. ::-		
7481							
7482							
7483							
7484	023712				LET CC := PSW		
7485	023712	013737	177776	000520		MOV	PSW, CC
7486	023720				LET CC := CC CLR.BY #177760		
7487	023720	042737	177760	000520		BIC	#177760, CC
7488	023726				IF CC NE CC.EXP THEN		
7489	023726	023737	000520	000522		CMP	CC, CC.EXP
7490	023734	001402				BEQ	\$50775
7491	023736				CALL ERROR		
7492	023736	004737	060044			JSR	PC, ERROR
7493	023742				ENDIF		
7494	023742						\$50775:

7495	023742			IF R0 NE R0.EXP THEN		
7496	023742	020037	000524		CMP	R0,R0.EXP
7497	023746	001402			BEQ	\$50776
7498	023750			CALL ERROR		
7499	023750	004737	060044		JSR	PC,ERROR
7500	023754			ENDIF		
7501	023754					\$50776:
7502	023754			IF R1 NE R1.EXP THEN		
7503	023754	020137	000526		CMP	R1,R1.EXP
7504	023760	001402			BEQ	\$50777
7505	023762			CALL ERROR		
7506	023762	004737	060044		JSR	PC,ERROR
7507	023766			ENDIF		
7508	023766					\$50777:
7509	023766			IF R2 NE R2.EXP THEN		
7510	023766	020237	000530		CMP	R2,R2.EXP
7511	023772	001402			BEQ	\$51000
7512	023774			CALL ERROR		
7513	023774	004737	060044		JSR	PC,ERROR
7514	024000			ENDIF		
7515	024000					\$51000:
7516	024000			IF R3 NE R3.EXP THEN		
7517	024000	020337	000532		CMP	R3,R3.EXP
7518	024004	001402			BEQ	\$51001
7519	024006			CALL ERROR		
7520	024006	004737	060044		JSR	PC,ERROR
7521	024012			ENDIF		
7522	024012					\$51001:
7523	024012			IF R4 NE R4.EXP THEN		
7524	024012	020437	000534		CMP	R4,R4.EXP
7525	024016	001402			BEQ	\$51002
7526	024020			CALL ERROR		
7527	024020	004737	060044		JSR	PC,ERROR
7528	024024			ENDIF		
7529	024024					\$51002:
7530	024024			IF R5 NE R5.EXP THEN		
7531	024024	020537	000536		CMP	R5,R5.EXP
7532	024030	001402			BEQ	\$51003
7533	024032			CALL ERROR		
7534	024032	004737	060044		JSR	PC,ERROR
7535	024036			ENDIF		
7536	024036					\$51003:
7537						
7538	024036			LET TSTCAS := TSTCAS + #4		
7539	024036	062737	000004 000544		ADD	#4,TSTCAS
7540	024044			LET COUNT := COUNT - #1		
7541	024044	005337	000470		DEC	COUNT
7542				;UNTIL COUNT EQ #0		
7543	024050			IF COUNT NE #0 THEN		
7544	024050	005737	000470		TST	COUNT
7545	024054	001402			BEQ	\$51004
7546	024056			INLINE <JMP \$5053>		
7547	024056	000137	023072		JMP	\$5053
7548	024062			ENDIF		
7549	024062					\$51004:
7550				::		

```
7551 ; DO ONE INLINE CASE - CVTNLI
7552 ; -
7553
7554 024062 CALL CLRREG JSR PC,CLRREG
7555 024062 004737 056604 CALL CLRDSB JSR PC,CLRDSB
7556 024066 004737 057040 INLINE <CVTNLI> CVTNLI
7557 024072 076153 INLINE <.WORD TCVNLI> .WORD TCVNLI
7558 024072 025056 INLINE <.WORD TCVNLI+4> .WORD TCVNLI+4
7559 024074 025062 LET CC := PSW MOV PSW,CC
7560 024100 013737 177776 000520 LET CC := CC CLR.BY #177760 BIC #177760,CC
7561 024106 042737 177760 000520 IF CC NE #0 THEN TST CC
7562 024114 005737 000520 CALL ERROR BEQ $51005
7563 024122 004737 060044 JSR PC,ERROR
7564 024122 060044 ENDIF $51005:
7565 024126 IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN TST R0
7566 024126 005700 BNE $51006
7567 024130 001004 TST R1
7568 024132 005701 BNE $51006
7569 024134 001002 TST R2
7570 024136 005702 BEQ $51007
7571 024140 001402 $51006:
7572 024142 CALL ERROR JSR PC,ERROR
7573 024142 004737 060044 ENDIF $51007:
7574 024146 IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN TST R3
7575 024146 005703 BNE $51010
7576 024150 001004 TST R4
7577 024152 005704 BNE $51010
7578 024154 001002 TST R5
7579 024156 005705 BEQ $51011
7580 024160 001402 $51010:
7581 024162 CALL ERROR JSR PC,ERROR
7582 024162 004737 060044 ENDIF $51011:
7583 024166 IF TCVNLI+4 NE #32707 THEN CMP TCVNLI+4,#32707
7584 024166 023727 025062 032707 BEQ $51012
7585 024174 001402 CALL ERROR JSR PC,ERROR
7586 024176 004737 060044 ENDIF $51012:
7587 024176 060044 IF TCVPLI+6 NE #41072 THEN
```


7607 024202 023727 023034 041072
 7608 024210 001402
 7609 024212
 7610 024212 004737 060044
 7611 024216
 7612 024216
 7613 024216
 7614 024216 000137 025100
 7615
 7616
 7617
 7618
 7619
 7620
 7621
 7622

CALL ERROR
 ENDIF
 INLINE <JMP XCVTPN>

CMP TCVPLI+6,#41072
 BEQ \$51013
 JSR PC,ERROR
 \$51013:
 JMP XCVTPN

::+
 :: TABLE FOR INTERPRETING ACCEPTABLE SIGN/DIGIT COMBINATIONS FOR OVERPUNCH
 ::-

7623	024222	173	000	060	SGNDGT: .BYTE	173,0,60,0,133,0,77,0	;+0
7624	024225	000	133	000			
7625	024230	077	000				
7626	024232	101	001	061	.BYTE	101,1,61,1	;+1
7627	024235	001					
7628	024236	102	002	062	.BYTE	102,2,62,2	;+2
7629	024241	002					
7630	024242	103	003	063	.BYTE	103,3,63,3	;+3
7631	024245	003					
7632	024246	104	004	064	.BYTE	104,4,64,4	;+4
7633	024251	004					
7634	024252	105	005	065	.BYTE	105,5,65,5	;+5
7635	024255	005					
7636	024256	106	006	066	.BYTE	106,6,66,6	;+6
7637	024261	006					
7638	024262	107	007	067	.BYTE	107,7,67,7	;+7
7639	024265	007					
7640	024266	110	010	070	.BYTE	110,10,70,10	;+8
7641	024271	010					
7642	024272	111	011	071	.BYTE	111,11,71,11	;+9
7643	024275	011					
7644	024276	175	200	135	.BYTE	175,200,135,200,41,200,72,200	; -0
7645	024301	200	041	200			
7646	024304	072	200				
7647	024306	112	201		.BYTE	112,201	; -1
7648	024310	113	202		.BYTE	113,202	; -2
7649	024312	114	203		.BYTE	114,203	; -3
7650	024314	115	204		.BYTE	115,204	; -4
7651	024316	116	205		.BYTE	116,205	; -5
7652	024320	117	206		.BYTE	117,206	; -6
7653	024322	120	207		.BYTE	120,207	; -7
7654	024324	121	210		.BYTE	121,210	; -8
7655	024326	122	211		.BYTE	122,211	; -9

::+
 :: TABLE OF TEST CASES FOR CVTNL

7656
 7657
 7658
 7659
 7660
 7661
 7662

```

7663      :-
7664
7665      024330 000055      TCVTNL: .WORD 45.          ;# OF TEST CASES FOR CVTNL
7666
7667      024332 000000      .WORD 0                ;SRC. DATA TYPE AND LENGTH
7668      024334 024617      .WORD CVTNL1          ;POINTER TO SRC. STRING
7669
7670      024336 050000      .WORD 50000           ;TEST CASE NUMBER 2
7671      024340 024621      .WORD CVTNL2
7672
7673      024342 040004      .WORD 40004           ;TEST CASE NUMBER 3
7674      024344 024623      .WORD CVTNL3
7675
7676      024346 050004      .WORD 50004           ;TEST CASE NUMBER 4
7677      024350 024631      .WORD CVTNL4
7678
7679      024352 030007      .WORD 30007           ;TEST CASE NUMBER 5
7680      024354 024636      .WORD CVTNL5
7681
7682      024356 000002      .WORD 2                ;TEST CASE NUMBER 6
7683      024360 024646      .WORD CVTNL6
7684
7685      024362 020005      .WORD 20005           ;TEST CASE NUMBER 7
7686      024364 024651      .WORD CVTNL7
7687
7688      024366 010005      .WORD 10005           ;TEST CASE NUMBER 8
7689      024370 024657      .WORD CVTNL8
7690
7691      024372 040004      .WORD 40004           ;TEST CASE NUMBER 9
7692      024374 024665      .WORD CVTNL9
7693
7694      024376 000012      .WORD 12                ;TEST CASE NUMBER 10
7695      024400 024673      .WORD CVTNLA
7696
7697      024402 020002      .WORD 20002           ;TEST CASE NUMBER 11
7698      024404 024706      .WORD CVTNLB
7699
7700      024406 020002      .WORD 20002           ;TEST CASE NUMBER 12
7701      024410 024711      .WORD CVTNLC
7702
7703      024412 020002      .WORD 20002           ;TEST CASE NUMBER 13
7704      024414 024714      .WORD CVTNLD
7705
7706      024416 020002      .WORD 20002           ;TEST CASE NUMBER 14
7707      024420 024717      .WORD CVTNLE
7708
7709      024422 020002      .WORD 20002           ;TEST CASE NUMBER 15
7710      024424 024722      .WORD CVTNLF
7711
7712      024426 020002      .WORD 20002           ;TEST CASE NUMBER 16
7713      024430 024725      .WORD CVTNLG
7714
7715      024432 020002      .WORD 20002           ;TEST CASE NUMBER 17
7716      024434 024730      .WORD CVTNLH
7717
7718      024436 020002      .WORD 20002           ;TEST CASE NUMBER 18
    
```

7719	024440	024733	.WORD	CVTNLS	
7720					
7721	024442	020002	.WORD	20002	;TEST CASE NUMBER 19
7722	024444	024736	.WORD	CVTNLJ	
7723					
7724	024446	020002	.WORD	20002	;TEST CASE NUMBER 20
7725	024450	024741	.WORD	CVTNLK	
7726					
7727	024452	020002	.WORD	20002	;TEST CASE NUMBER 21
7728	024454	024744	.WORD	CVTNLL	
7729					
7730	024456	020002	.WORD	20002	;TEST CASE NUMBER 22
7731	024460	024747	.WORD	CVTNLM	
7732					
7733	024462	020002	.WORD	20002	;TEST CASE NUMBER 23
7734	024464	024752	.WORD	CVTNLN	
7735					
7736	024466	020002	.WORD	20002	;TEST CASE NUMBER 24
7737	024470	024755	.WORD	CVTNLO	
7738					
7739	024472	020002	.WORD	20002	;TEST CASE NUMBER 25
7740	024474	024760	.WORD	CVTNLP	
7741					

7742	024476	020002	.WORD	20002	;TEST CASE NUMBER 26
7743	024500	024763	.WORD	CVTNLQ	
7744					
7745	024502	020002	.WORD	20002	;TEST CASE NUMBER 27
7746	024504	024766	.WORD	CVTNLR	
7747					
7748	024506	020002	.WORD	20002	;TEST CASE NUMBER 28
7749	024510	024771	.WORD	CVTNLS	
7750					
7751	024512	020002	.WORD	20002	;TEST CASE NUMBER 29
7752	024514	024774	.WORD	CVTNLT	
7753					
7754	024516	020002	.WORD	20002	;TEST CASE NUMBER 30
7755	024520	024777	.WORD	CVTNLU	
7756					
7757	024522	020002	.WORD	20002	;TEST CASE NUMBER 31
7758	024524	025002	.WORD	CVTNLV	
7759					
7760	024526	020002	.WORD	20002	;TEST CASE NUMBER 32
7761	024530	025005	.WORD	CVTNLW	
7762					
7763	024532	020002	.WORD	20002	;TEST CASE NUMBER 33
7764	024534	025010	.WORD	CVTNLX	
7765					
7766	024536	020002	.WORD	20002	;TEST CASE NUMBER 34
7767	024540	025013	.WORD	CVTNLY	
7768					
7769	024542	020002	.WORD	20002	;TEST CASE NUMBER 35
7770	024544	025016	.WORD	CVTNLZ	
7771					
7772	024546	020002	.WORD	20002	;TEST CASE NUMBER 36
7773	024550	025021	.WORD	CVTNAA	
7774					
7775	024552	020002	.WORD	20002	;TEST CASE NUMBER 37
7776	024554	025024	.WORD	CVTNBB	
7777					
7778	024556	020002	.WORD	20002	;TEST CASE NUMBER 38
7779	024560	025027	.WORD	CVTNCC	
7780					
7781	024562	020002	.WORD	20002	;TEST CASE NUMBER 39
7782	024564	025032	.WORD	CVTNDD	
7783					
7784	024566	020002	.WORD	20002	;TEST CASE NUMBER 40
7785	024570	025035	.WORD	CVTNEE	
7786					
7787	024572	020002	.WORD	20002	;TEST CASE NUMBER 41
7788	024574	025040	.WORD	CVTNFF	
7789					
7790	024576	020002	.WORD	20002	;TEST CASE NUMBER 42
7791	024600	025043	.WORD	CVTNHG	
7792					
7793	024602	020002	.WORD	20002	;TEST CASE NUMBER 43
7794	024604	025046	.WORD	CVTNHH	
7795					
7796	024606	020002	.WORD	20002	;TEST CASE NUMBER 44
7797	024610	025051	.WORD	CVTNII	

7854	024721	000			.BYTE		
7855	024722	061	101	CVTNLF:	.BYTE	61,101	:DECIMAL NUMERIC SRC STRING 15
7856							
7857	024724	000			.BYTE		
7858	024725	061	061	CVTNLG:	.BYTE	61,61	:DECIMAL NUMERIC SRC STRING 16
7859							
7860	024727	000			.BYTE		
7861	024730	061	102	CVTNLH:	.BYTE	61,102	:DECIMAL NUMERIC SRC STRING 17
7862							
7863	024732	000			.BYTE		
7864	024733	061	062	CVTNLS:	.BYTE	61,62	:DECIMAL NUMERIC SRC STRING 18
7865							
7866	024735	000			.BYTE		
7867	024736	061	103	CVTNLJ:	.BYTE	61,103	:DECIMAL NUMERIC SRC STRING 19
7868							
7869	024740	000			.BYTE		
7870	024741	061	063	CVTNLK:	.BYTE	61,63	:DECIMAL NUMERIC SRC STRING 20
7871							
7872	024743	000			.BYTE		
7873	024744	061	104	CVTNLL:	.BYTE	61,104	:DECIMAL NUMERIC SRC STRING 21
7874							
7875	024746	000			.BYTE		
7876	024747	061	064	CVTNLM:	.BYTE	61,64	:DECIMAL NUMERIC SRC STRING 22
7877							
7878	024751	000			.BYTE		
7879	024752	061	105	CVTNLN:	.BYTE	61,105	:DECIMAL NUMERIC SRC STRING 23
7880							
7881	024754	000			.BYTE		
7882	024755	061	065	CVTNLO:	.BYTE	61,65	:DECIMAL NUMERIC SRC STRING 24
7883							
7884	024757	000			.BYTE		
7885	024760	061	106	CVTNLP:	.BYTE	61,106	:DECIMAL NUMERIC SRC STRING 25
7886							
7887	024762	000			.BYTE		
7888	024763	061	066	CVTNLQ:	.BYTE	61,66	:DECIMAL NUMERIC SRC STRING 26
7889							
7890	024765	000			.BYTE		
7891	024766	061	107	CVTNLR:	.BYTE	61,107	:DECIMAL NUMERIC SRC STRING 27
7892							
7893	024770	000			.BYTE		
7894	024771	061	067	CVTNLS:	.BYTE	61,67	:DECIMAL NUMERIC SRC STRING 28
7895							
7896	024773	000			.BYTE		
7897	024774	061	110	CVTNLT:	.BYTE	61,110	:DECIMAL NUMERIC SRC STRING 29
7898							
7899	024776	000			.BYTE		
7900	024777	061	070	CVTNLU:	.BYTE	61,70	:DECIMAL NUMERIC SRC STRING 30
7901							
7902	025001	000			.BYTE		
7903	025002	061	111	CVTNLV:	.BYTE	61,111	:DECIMAL NUMERIC SRC STRING 31
7904							
7905	025004	000			.BYTE		
7906	025005	061	071	CVTNLW:	.BYTE	61,71	:DECIMAL NUMERIC SRC STRING 32
7907							
7908	025007	000			.BYTE		
7909	025010	061	175	CVTNLX:	.BYTE	61,175	:DECIMAL NUMERIC SRC STRING 33

```

7910
7911 025012 000
7912 025013 061 135 CVTNLY: .BYTE 61,135 ;DECIMAL NUMERIC SRC STRING 34
7913
7914 025015 000
7915 025016 061 041 CVTNLZ: .BYTE 61,41 ;DECIMAL NUMERIC SRC STRING 35
7916
7917 025020 000
7918 025021 061 072 CVTNAA: .BYTE 61,72 ;DECIMAL NUMERIC SRC STRING 36
7919
7920 025023 000
7921 025024 061 112 CVTNBB: .BYTE 61,112 ;DECIMAL NUMERIC SRC STRING 37
7922
7923 025026 000
7924 025027 061 113 CVTNCC: .BYTE 61,113 ;DECIMAL NUMERIC SRC STRING 38
7925
7926 025031 000
7927 025032 061 114 CVTNDD: .BYTE 61,114 ;DECIMAL NUMERIC SRC STRING 39
7928
7929 025034 000
7930 025035 061 115 CVTNEE: .BYTE 61,115 ;DECIMAL NUMERIC SRC STRING 40
7931
7932 025037 000
7933 025040 061 116 CVTNFF: .BYTE 61,116 ;DECIMAL NUMERIC SRC STRING 41
7934
7935 025042 000
7936 025043 061 117 CVTNGG: .BYTE 61,117 ;DECIMAL NUMERIC SRC STRING 42
7937
7938 025045 000
7939 025046 061 120 CVTNHH: .BYTE 61,120 ;DECIMAL NUMERIC SRC STRING 43
7940
7941 025050 000
7942 025051 061 121 CVTNII: .BYTE 61,121 ;DECIMAL NUMERIC SRC STRING 44
7943
7944 025053 000
7945 025054 061 122 CVTNJJ: .BYTE 61,122 ;DECIMAL NUMERIC SRC STRING 45
7946
7947
7948 .EVEN
7949
7950 025056 TCVNLI: ;INLINE TEST CASE
7951 025056 000012 .WORD 000012 ;DATA TYPE = SIGNED ZONED, LENGTH = 12
7952 025060 025066 .WORD NL1 ;ADDRESS OF DECIMAL SRC STRING
7953 025062 000000 .WORD 0 ;LONG INTEGER RESULT, LOW
7954 025064 000000 .WORD 0 ;LONG INTEGER RESULT, HI, AND SIGN
7955
7956 025066 061 061 061 NL1: .BYTE 61,61,61,61,61,61,61,61,61 ; = +1111111111 IN BCD
7957 025071 061 061 061
7958 025074 061 061 061
7959 025077 061
7960 .EVEN
7961
    
```

```
7962 .SBTTL CVTPN
7963 :*****
7964 :
7965 : CVTPN TEST
7966 :
7967 :*****
7968 025100 INLINE <XCVTPN:>
7969 025100 XCVTPN:
7970 025100 INLINE <SCOPE> SCOPE
7971 025100 000004 ;ITERATION COUNT
7972 025102 LET $TIMES := #20 ;GET TEST COUNT MOV #20,$TIMES
7973 025102 012737 000020 000404 ;POINT TO FIRST TEST CASE MOV TCVTPN,COUNT
7974 025110 LET COUNT := TCVTPN MOV TCVTPN,COUNT
7975 025110 013737 027014 000470 LET TSTCAS := #TCVTPN+2 MOV #TCVTPN+2,TSTCAS
7976 025116 012737 027016 000544 ;REPEAT
7977 025116 012737 027016 000544 INLINE <$5054:> $5054:
7978
7979 025124
7980 025124
7981
7982 :+
7983 : GET TEST DATA FROM TABLE
7984 :-
7985
7986 025124 LET R0 := TSTCAS
7987 025124 013700 000544 MOV TSTCAS,R0
7988 025130 LET DATA00 := (R0)+ ;SRC DATA TYPE AND LENGTH
7989 025130 012037 000562 MOV (R0)+,DATA00
7990 025134 LET DATA01 := (R0)+ ;POINTER TO PACKED SRC STRING
7991 025134 012037 000564 MOV (R0)+,DATA01
7992 025140 LET DATA02 := (R0) ;DST.LEN
7993 025140 011037 000566 MOV (R0),DATA02
7994
7995
7996 :+
7997 : SET UP R0.EXP, R1.EXP, R3.EXP, R4.EXP, R5.EXP, R4, AND R5.
7998 :-
7999
8000 025144 LET R0.EXP := #0 CLR R0.EXP
8001 025150 LET R1.EXP := #0 CLR R1.EXP
8002 025150 005037 000526 LET R3.EXP := #DS.DST MOV #DS.DST,R3.EXP
8003 025154 012737 064400 000532 IF #1 SETIN COUNT THEN
8004 025154 012737 064400 000532
8005 025162
8006 025162 032737 000001 000470 BIT #1,COUNT
8007 025170 001405 BEQ $51014
8008 025172 LET R4 := #-1 MOV #-1,R4
8009 025172 012704 177777 LET R5 := #-1 MOV #-1,R5
8010 025176 012705 177777
8011 025176 012705 177777
8012 025202 ELSE BR $51015
8013 025202 000402 $51014:
8014 025204
8015 025204 LET R4 : #0 CLR R4
8016 025204 005004
8017 025206 LET R5 := #0
```



```

8074 025316 020127 000014
8075 025322 001003
8076 025324
8077 025324 005037 000450
8078 025330
8079 025330 000403
8080 025332
8081 025332
8082 025332 012737 177777 000450
8083 025340
8084 025340
8085 025340
8086 025340
8087
8088
8089
8090
8091
8092
8093
8094 025340
8095 025340 023737 000516 000514
8096 025346 103403
8097 025350
8098 025350 013700 000514
8099 025354
8100 025354 000402
8101 025356
8102 025356
8103 025356 013700 000516
8104 025362
8105 025362
8106
8107
8108
8109
8110
8111
8112
8113
8114 025362
8115 025362 012701 064442
8116 025366 063701 000566
8117 025372 005301
8118 025374
8119 025374 013702 000514
8120 025400
8121 025400 006202
8122 025402
8123 025402 063702 000564
8124
8125
8126
8127
8128
8129

```

```

                LET SIGNO := #0
            ELSE
                LET SIGNO := #-1
            ENDIF
        ENDIF

;+
; CALCULATE THE NUMBER OF CHARACTERS TO BE MOVED INTO DS.EXP;
; USE R0.
;-

IF DSLEN2 HIS DSLEN1 THEN                ;DST.LEN >= SRC.LEN
    CMP DSLEN2,DSLEN1
    BLO $51023
    LET R0 := DSLEN1
ELSE                                        ;SRC.LEN > DST.LEN
    BR $51024
    LET R0 := DSLEN2
ENDIF

;+
; MAKE R1 POINT TO THE END OF THE DESTINATION IN DS.EXP;
; MAKE R2 POINT TO THE END OF THE SOURCE STRING IN THE
; TABLE OF TEST CASES.
;-

LET R1 := #DS.EXP + DATA02 - #1
    MOV #DS.EXP,R1
    ADD DATA02,R1
    DEC R1
LET R2 := DSLEN1
    MOV DSLEN1,R2
    INLINE <ASR R2>
    ASR R2
LET R2 := R2 + DATA01
    ADD DATA01,R2

;+
; MOVE THE LEAST SIGNIFICANT BCD FROM THE SOURCE STRING
; TO THE EXPECTED DESTINATION BUFFER, IN UNPACKED FORMAT.
;-

```

```

8130
8131 025406          LET R3 :B= (R2)
8132 025406 111203          MOVB      (R2),R3
8133 025410          INLINE  <ASH #-4,R3>
8134 025410 072327 177774          ASH      #-4,R3
8135 025414          LET (R1) :B= R3 SET.BY #60 CLR.BY #300
8136 025414 110311          MOVB      R3,(R1)
8137 025416 152711 000060          BISB      #60,(R1)
8138 025422 142711 000300          BICB      #300,(R1)
8139 025426          LET R0 := R0 - #1          ;DECREMENT COUNT
8140 025426 005300          DEC      R0
8141
8142
8143
8144
8145 ;+
8146 ; MOVE THE REST OF THE SOURCE TO THE EXPECTED DESTINATION
8147 ; BUFFER, IN UNPACKED FORMAT, USING R0 AS A COUNTER FOR
8148 ; THE NUMBER OF CHARACTERS TO MOVE.
8149 ;-
8150
8151 025430          WHILE R0 GT #0 DO
8152 025430 005700          $51025:
8153 025432 003425          TST      R0
8154 025434          LET R1 := R1 - #1          ;DECREMENT DEST. POINTER
8155 025434 005301          BLE      $51026
8156 025436          LET R2 := R2 - #1          ;DECREMENT SRC. POINTER
8157 025436 005302          DEC      R1
8158 025440          LET (R1) :B= (R2) CLR.BY #300 SET.BY #60          DEC      R2
8159 025440 111211          MOVB      (R2),(R1)
8160 025442 142711 000300          BICB      #300,(R1)
8161 025446 152711 000060          BISB      #60,(R1)
8162 025452          LET R0 := R0 - #1          ;DECREMENT COUNT
8163 025452 005300          DEC      R0
8164 025454          IF R0 HI #0 THEN          ;COUNT > 0
8165 025454 005700          TST      R0
8166 025456 101412          BLOS      $51027
8167 025460          LET R1 := R1 - #1          ;DECREMENT DEST. POINTER
8168 025460 005301          DEC      R1
8169 025462          LET R3 :B= (R2)
8170 025462 111203          MOVB      (R2),R3
8171 025464          INLINE  <ASH      #-4,R3>
8172 025464 072327 177774          ASH      #-4,R3
8173 025470          LET (R1) :B= R3 CLR.BY #300 SET.BY #60
8174 025470 110311          MOVB      R3,(R1)
8175 025472 142711 000300          BICB      #300,(R1)
8176 025476 152711 000060          BISB      #60,(R1)
8177 025502          LET R0 := R0 - #1          ;DECREMENT COUNT
8178 025502 005300          DEC      R0
8179 025504          ENDIF
8180 025504          ENDDO
8181 025504 000751          $51027:
8182 025506          BR      $51025
8183          $51026:
8184
8185 ;+
    
```

```

8186          ; FILL THE REST OF THE DESTINATION WITH LEADING ZEROES (60'S).
8187          :-
8188
8189 025506    WHILE R1 GT #DS.EXP DO
8190 025506
8191 025506    020127 064442          $51030:
8192 025512    003403          CMP      R1,#DS.EXP
8193 025514          LET -(R1) :B= #60          BLE      $51031
8194 025514    112741 000060          MOVB     #60,-(R1)
8195 025520    ENDDO
8196 025520    000772          BR       $51030
8197 025522          $51031:
8198
8199
8200          ;+
8201          ; PUT THE EXPECTED VALUE OF THE NBIT INTO CC.EXP
8202          :-
8203
8204 025522    IF DSLEN2 HI #0 AND SIGNO EQ #-1 THEN
8205 025522    005737 000516          TST     DSLEN2
8206 025526    101407          BLOS    $51032
8207 025530    023727 000450 177777    CMP     SIGNO,#-1
8208 025536    001003          BNE     $51032
8209 025540          LET CC.EXP := CC.EXP SET.BY #NBIT
8210 025540    052737 000010 000522    BIS     #NBIT,CC.EXP
8211 025546    ENDF
8212 025546          $51032:
8213
8214
8215          ;+
8216          ; PUT THE EXPECTED VALUE OF THE ZBIT INTO CC.EXP
8217          :-
8218
8219
8220 025546    IF DSLEN1 EQ #0 OR DSLEN2 EQ #0 THEN
8221 025546    005737 000514          TST     DSLEN1
8222 025552    001403          BEQ     $51033
8223 025554    005737 000516          TST     DSLEN2
8224 025560    001003          BNE     $51034
8225 025562          $51033:
8226 025562          LET CC.EXP :- CC.EXP SET.BY #ZBIT
8227 025562    052737 000004 000522    BIS     #ZBIT,CC.EXP
8228 025570    ENDF
8229 025570          $51034:
8230
8231
8232          ;+
8233          ; PUT THE EXPECTED VALUE OF THE VBIT INTO CC.EXP
8234          :-
8235
8236 025570    IF DSLEN1 HI DSLEN2 THEN          ;SRC.LEN > DST.LEN
8237 025570    023737 000514 000516    CMP     DSLEN1,DSLEN2
8238 025576    101403          BLOS    $51035
8239 025600          LET CC.EXP := CC.EXP SET.BY #VBIT
8240 025600    052737 000002 000522    BIS     #VBIT,CC.EXP
8241 025606    ENDF
    
```


8298	025674	000137	026576			JMP	\$5403
8299	025700			ENDIF			
8300	025700					\$51040:	
8301							
8302				;RESTORE FIRST AND LAST BYTES OF DESTINATION			
8303							
8304	025700			LET DS.EXP :B= DSTMP0			
8305	025700	113737	000554	064442		MOVB	DSTMP0,DS.EXP
8306	025706			IF DATA02 NE #0 THEN			
8307	025706	005737	000566			TST	DATA02
8308	025712	001406				BEQ	\$51041
8309	025714			LET RO :- #DS.EXP-1 + DATA02			
8310	025714	012700	064441			MOV	#DS.EXP-1,RO
8311	025720	063700	000566			ADD	DATA02,RO
8312	025724			LET (RO) :B- DSTMP1			
8313	025724	113710	000555			MOVB	DSTMP1,(RO)
8314	025730			ENDIF			
8315	025730					\$51041:	
8316							
8317				;*** SIGNED ZONED ***			
8318	025730			IF DATTYP EQ #0 THEN			
8319	025730	005737	000464			TST	DATTYP
8320	025734	001022				BNE	\$51042
8321	025736			LET DS.EXP-1 :B= #0		;RESTORE AFTER LEADING SEPARATE	
8322	025736	105037	064441			CLRB	DS.EXP-1
8323	025742			IF DATA02 HI #0 THEN		;DST.LEN > 0	
8324	025742	005737	000566			TST	DATA02
8325	025746	101415				BLOS	\$51043
8326	025750			LET RO :- #DS.EXP-1 + DATA02			
8327	025750	012700	064441			MOV	#DS.EXP-1,RO
8328	025754	063700	000566			ADD	DATA02,RO
8329	025760			IF #NBIT NOTSET IN CC.EXP THEN		;SIGN IS +	
8330	025760	032737	000010	000522		BIT	#NBIT,CC.EXP
8331	025766	001003				BNE	\$51044
8332	025770			LET (RO) :B- (RO) SET.BY #60			
8333	025770	152710	000060			BISB	#60,(RO)
8334	025774			ELSE		;SIGN IS -	
8335	025774	000402				BR	\$51045
8336	025776					\$51044:	
8337	025776			LET (RO) :B (RO) SET.BY #160			
8338	025776	152710	000160			BISB	#160,(RO)
8339	026002			ENDIF			
8340	026002					\$51045:	
8341	026002			ENDIF			
8342	026002					\$51043:	
8343	026002			ENDIF			
8344	026002					\$51042:	
8345							
8346				;*** UNSIGNED ZONED ***			
8347	026002			IF DATTYP EQ #10000 THEN			
8348	026002	023727	000464	010000		CMP	DATTYP,#10000
8349	026010	001014				BNE	\$51046
8350	026012			IF DATA02 HI #0 THEN		;DST.LEN > 0	
8351	026012	005737	000566			TST	DATA02
8352	026016	101406				BLOS	\$51047
8353	026020			LET RO :- #DS.EXP-1 + DATA02			

8410					:*** LEADING OVERPUNCH ***		
8411	026170				IF DATTYP EQ #30000 THEN		
8412	026170	023727	000464	030000		CMP	DATTYP,#30000
8413	026176	001020				BNE	\$51055
8414	026200				IF DATA02 HI #0 THEN	:DST.LEN > 0	
8415	026200	005737	000566			TST	DATA02
8416	026204	101415				BLOS	\$51056
8417	026206				LET R1 :B- DSTMPO CLR.BY #60	:GET MSD	
8418	026206	113701	000554			MOVB	DSTMPO,R1
8419	026212	142701	000060			BICB	#60,R1
8420	026216				IF #NBIT SETIN CC.EXP THEN		
8421	026216	032737	000010	000522		BIT	#NBIT,CC.EXP
8422	026224	001402				BEQ	\$51057
8423	026226				LET R1 := R1 + #10.	:FORM OVPNCH TABLE OFFSET	
8424	026226	062701	000012			ADD	#10.,R1
8425	026232				ENDIF		
8426	026232					\$51057:	
8427	026232				LET DS.EXP :B= OVPNCH(R1)	:GET CORRECT SIGN/DIGIT	
8428	026232	116137	056530	064442		MOVB	OVPNCH(R1),DS.EXP
8429	026240				ENDIF		
8430	026240					\$51056:	
8431	026240				ENDIF		
8432	026240					\$51055:	
8433							
8434					:*** TRAILING SEPARATE ***		
8435	026240				IF DATTYP EQ #40000 THEN		
8436	026240	023727	000464	040000		CMP	DATTYP,#40000
8437	026246	001015				BNE	\$51060
8438	026250				LET R0 := #DS.EXP + DATA02		
8439	026250	012700	064442			MOV	#DS.EXP,R0
8440	026254	063700	000566			ADD	DATA02,R0
8441	026260				IF #NBIT NOTSETIN CC.EXP THEN	:SIGN	
8442	026260	032737	000010	000522		IS +	
8443	026266	001003				BIT	#NBIT,CC.EXP
8444	026270				LET (R0) :B- #53	BNE	\$51061
8445	026270	112710	000053			MOVB	#53,(R0)
8446	026274				ELSE	:SIGN IS -	
8447	026274	000402				BR	\$51062
8448	026276					\$51061:	
8449	026276				LET (R0) :B- #55		
8450	026276	112710	000055			MOVB	#55,(R0)
8451	026302				ENDIF		
8452	026302					\$51062:	
8453	026302				ENDIF		
8454	026302					\$51060:	
8455							
8456					:*** LEADING SEPARATE ***		
8457	026302				IF DATTYP EQ #50000 THEN		
8458	026302	023727	000464	050000		CMP	DATTYP,#50000
8459	026310	001020				BNE	\$51063
8460	026312				LET R0 := #DS.EXP + DATA02		
8461	026312	012700	064442			MOV	#DS.EXP,R0
8462	026316	063700	000566			ADD	DATA02,R0
8463	026322				LET (R0) :B= #0	:RESTORE AFTER TRAILING SEPARATE	
8464	026322	105010				CLRB	(R0)
8465	026324				IF #NBIT NOTSETIN CC.EXP THEN	:SIGN = +	

8466	026324	032737	000010	000522		BIT	#NBIT,CC.EXP
8467	026332	001004				BNE	\$51064
8468	026334				LET DS.EXP-1 :B= #53		
8469	026334	112737	000053	064441		MOVB	#53,DS.EXP-1
8470	026342				ELSE		:SIGN = -
8471	026342	000403				BR	\$51065
8472	026344						\$51064:
8473	026344				LET DS.EXP-1 :B= #55		
8474	026344	112737	000055	064441		MOVB	#55,DS.EXP-1
8475	026352				ENDIF		\$51065:
8476	026352						
8477	026352				ENDIF		\$51063:
8478	026352						
8479							
8480					:+		
8481					:+ SET UP R2.EXP, R0, R1, R2, AND R3; DO THE INSTRUCTION.		
8482					:-		
8483							
8484	026352				LET R0 := DATA00	:SRC DATA TYPE AND LENGTH	
8485	026352	013700	000562			MOV	DATA00,R0
8486	026356				LET R1 := DATA01	:SRC POINTER	
8487	026356	013701	000564			MOV	DATA01,R1
8488	026362				LET R2 := DATA02 + DATTYP	:DST DATA TYPE AND LENGTH	
8489	026362	013702	000566			MOV	DATA02,R2
8490	026366	063702	000464			ADD	DATTYP,R2
8491	026372				LET R3 := #DS.DST	:DST POINTER	
8492	026372	012703	064400			MOV	#DS.DST,R3
8493	026376				LET R2.EXP := R2		
8494	026376	010237	000530			MOV	R2,R2.EXP
8495	026402				INLINE <CVTPN>		
8496	026402	076054					CVTPN
8497							
8498							
8499					:+		
8500					:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
8501					:-		
8502							
8503	026404				LET CC := PSW		
8504	026404	013737	177776	000520		MOV	PSW,CC
8505	026412				LET CC := CC CLR.BY #177760		
8506	026412	042737	177760	000520		BIC	#177760,CC
8507	026420				IF CC NE CC.EXP THEN		
8508	026420	023737	000520	000522		CMP	CC,CC.EXP
8509	026426	001402				BEQ	\$51066
8510	026430				CALL ERROR		
8511	026430	004737	060044			JSR	PC,ERROR
8512	026434				ENDIF		\$51066:
8513	026434						
8514	026434				IF R0 NE R0.EXP THEN		
8515	026434	020037	000524			CMP	R0,R0.EXP
8516	026440	001402				BEQ	\$51067
8517	026442				CALL ERROR		
8518	026442	004737	060044			JSR	PC,ERROR
8519	026446				ENDIF		\$51067:
8520	026446						
8521	026446				IF R1 NE R1.EXP THEN		

8522	026446	020137	000526						
8523	026452	001402							
8524	026454			CALL ERROR					
8525	026454	004737	060044						
8526	026460			ENDIF					
8527	026460								
8528	026460			IF R2 NE R2.EXP THEN					
8529	026460	020237	000530						
8530	026464	001402							
8531	026466			CALL ERROR					
8532	026466	004737	060044						
8533	026472			ENDIF					
8534	026472								
8535	026472			IF R3 NE R3.EXP THEN					
8536	026472	020337	000532						
8537	026476	001402							
8538	026500			CALL ERROR					
8539	026500	004737	060044						
8540	026504			ENDIF					
8541	026504								
8542	026504			IF R4 NE R4.EXP THEN					
8543	026504	020437	000534						
8544	026510	001402							
8545	026512			CALL ERROR					
8546	026512	004737	060044						
8547	026516			ENDIF					
8548	026516								
8549	026516			IF R5 NE R5.EXP THEN					
8550	026516	020537	000536						
8551	026522	001402							
8552	026524			CALL ERROR					
8553	026524	004737	060044						
8554	026530			ENDIF					
8555	026530								
8556									
8557				;COMPARE DS.DST WITH DS.EXP					
8558	026530			LET R0 := #DS.EXP-1					
8559	026530	012700	064441						
8560	026534			LET R1 := #DS.DST-1					
8561	026534	012701	064377						
8562	026540			LET ERRFLG := #0					
8563	026540	005037	000456						
8564	026544			WHILE R0 NE #DS.EXP+40 AND ERRFLG EQ #0 DO					
8565	026544								
8566	026544	020027	064502						
8567	026550	001410							
8568	026552	005737	000456						
8569	026556	001005							
8570	026560			IFB (R0)+ NE (R1)+ THEN					
8571	026560	122021							
8572	026562	001402							
8573	026564			CALL ERROR					
8574	026564	004737	060044						
8575	026570			ENDIF					
8576	026570								
8577	026570			ENDDO					

;CLEAR THE ERROR FLAG
 CLR ERRFLG

CMP R1,R1,EXP
 BEQ \$51070
 JSR PC,ERROR
 \$51070:
 CMP R2,R2,EXP
 BEQ \$51071
 JSR PC,ERROR
 \$51071:
 CMP R3,R3,EXP
 BEQ \$51072
 JSR PC,ERROR
 \$51072:
 CMP R4,R4,EXP
 BEQ \$51073
 JSR PC,ERROR
 \$51073:
 CMP R5,R5,EXP
 BEQ \$51074
 JSR PC,ERROR
 \$51074:
 MOV #DS.EXP-1,R0
 MOV #DS.DST-1,R1
 \$51075:
 CMP R0,#DS.EXP+40
 BEQ \$51076
 TST ERRFLG
 BNE \$51076
 CMPB (R0)+,(R1)+
 BEQ \$51077
 JSR PC,ERROR
 \$51077:

8578	026570	000765					BR	\$51075
8579	026572						\$51076:	
8580						:ENDINC		
8581	026572					INLINE <JMP \$5401>		
8582	026572	000137	025640			INLINE <\$5403:>	JMP	\$5401
8583	026576						\$5403:	
8584	026576							
8585								
8586	026576					LET TSTCAS := TSTCAS + #6	ADD	#6,TSTCAS
8587	026576	062737	000006	000544		LET COUNT := COUNT - #1	DEC	COUNT
8588	026604							
8589	026604	005337	000470			:UNTIL COUNT EQ #0		
8590						IF COUNT NE #0 THEN		
8591	026610						TST	COUNT
8592	026610	005737	000470				BEQ	\$51100
8593	026614	001402						
8594	026616					INLINE <JMP \$5054>	JMP	\$5054
8595	026616	000137	025124					
8596	026622					ENDIF		
8597	026622						\$51100:	
8598								
8599						:+ : DC ONE INLINE CASE - CVTPNI :-		
8600								
8601								
8602	026622					CALL CLRREG	JSR	PC,CLRREG
8603	026622	004737	056604			CALL CLRDSB	JSR	PC,CLRDSB
8604	026626							
8605	026626	004737	057040			INLINE <CVTPNI>		
8606	026632						CVTPNI	
8607	026632	076154				INLINE <.WORD TCVPNI>	.WORD	TCVPNI
8608	026634							
8609	026634	027150				INLINE <.WORD TCVPNI+4>	.WORD	TCVPNI+4
8610	026636							
8611	026636	027154				LET CC := PSW	MOV	PSW,CC
8612	026640							
8613	026640	013737	177776	000520		LET CC := CC CLR.BY #177760	BIC	#177760,CC
8614	026646							
8615	026646	042737	177760	000520		IF CC NE #10 THEN		
8616	026654							
8617	026654	023727	000520	000010		CALL ERROR	CMP	CC,#10
8618	026662	001402					BEQ	\$51101
8619	026664							
8620	026664	004737	060044			ENDIF	JSR	PC,ERROR
8621	026670							
8622	026670						\$51101:	
8623	026670					IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN		
8624	026670	005700					TST	R0
8625	026672	001004					BNE	\$51102
8626	026674	005701					TST	R1
8627	026676	001002					BNE	\$51102
8628	026700	005702					TST	R2
8629	026702	001402					BEQ	\$51103
8630	026704						\$51102:	
8631	026704					CALL ERROR		
8632	026704	004737	060044			ENDIF	JSR	PC,ERROR
8633	026710							

8634	026710								\$51103:
8635	026710								TST R3
8636	026710	005703							BNE \$51104
8637	026712	001004							TST R4
8638	026714	005704							BNE \$51104
8639	026716	001002							TST R5
8640	026720	005705							BEQ \$51105
8641	026722	001402							\$51104:
8642	026724								JSR PC,ERROR
8643	026724								
8644	026724	004737	060044						\$51105:
8645	026730								CMPB DS.DST,#61
8646	026730								BEQ \$51106
8647	026730								JSR PC,ERROR
8648	026730	123727	064400	000061					\$51106:
8649	026736	001402							CMPB DS.DST+1,#62
8650	026740								BEQ \$51107
8651	026740	004737	060044						JSR PC,ERROR
8652	026744								\$51107:
8653	026744								CMPB DS.DST+2,#63
8654	026744								BEQ \$51110
8655	026744	123727	064401	000062					JSR PC,ERROR
8656	026752	001402							\$51110:
8657	026754								CMPB DS.DST+3,#164
8658	026754	004737	060044						BEQ \$51111
8659	026760								JSR PC,ERROR
8660	026760								\$51111:
8661	026760								JMP XCVTPN
8662	026760	123727	064402	000063					
8663	026766	001402							
8664	026770								
8665	026770	004737	060044						
8666	026774								
8667	026774								
8668	026774								
8669	026774	123727	064403	000164					
8670	027002	001402							
8671	027004								
8672	027004	004737	060044						
8673	027010								
8674	027010								
8675	027010								
8676	027010	000137	027164						
8677									
8678									
8679									
8680									
8681									
8682									
8683									
8684	027014	000011							
8685									
8686	027016	060000							
8687	027020	027104							
8688	027022	000000							
8689									

```

: +
: TABLE OF TEST CASES FOR CVTPN
: -

```

```

TCVTPN: .WORD 9. ;# OF TEST CASES FOR CVTPN
        .WORD 60000 ;SRC DATA TYPE AND LENGTH
        .WORD CVTPN1 ;POINTER TO PACKED SRC STRING
        .WORD 0 ;DST.LEN

```

8690	027024	060000				.WORD	60000		;TEST CASE NUMBER 2
8691	027026	027105				.WORD	CVTPN2		
8692	027030	000005				.WORD	5		
8693									
8694	027032	060004				.WORD	60004		;TEST CASE NUMBER 3
8695	027034	027106				.WORD	CVTPN3		
8696	027036	000000				.WORD	0		
8697									
8698									
8699	027040	060025				.WORD	60025		;TEST CASE NUMBER 4
8700	027042	027111				.WORD	CVTPN4		
8701	027044	000014				.WORD	14		
8702									
8703	027046	060007				.WORD	60007		;TEST CASE NUMBER 5
8704	027050	027124				.WORD	CVTPN5		
8705	027052	000003				.WORD	3		
8706									
8707	027054	070007				.WORD	70007		;TEST CASE NUMBER 6
8708	027056	027130				.WORD	CVTPN6		
8709	027060	000003				.WORD	3		
8710									
8711	027062	060004				.WORD	60004		;TEST CASE NUMBER 7
8712	027064	027134				.WORD	CVTPN7		
8713	027066	000037				.WORD	37		
8714									
8715	027070	060012				.WORD	60012		;TEST CASE NUMBER 8
8716	027072	027137				.WORD	CVTPN8		
8717	027074	000012				.WORD	12		
8718									
8719	027076	060005				.WORD	60005		;TEST CASE NUMBER 9
8720	027100	027145				.WORD	CVTPN9		
8721	027102	000005				.WORD	5		
8722									
8723									
8724									
8725	027104	014				CVTPN1: .BYTE	014		;PACKED DECIMAL STRING 1
8726									
8727	027105	015				CVTPN2: .BYTE	015		;PACKED DECIMAL STRING 2
8728									
8729	027106	001	043	114		CVTPN3: .BYTE	001,043,114		;PACKED DECIMAL STRING 3
8730									
8731	027111	020	043	127		CVTPN4: .BYTE	020,043,127,206,224,022,003,000,100,000,014		;PACKED DECIMAL STRING 4
8732	027114	206	224	022					
8733	027117	003	000	100					
8734	027122	000	014						
8735									
8736	027124	022	000	007		CVTPN5: .BYTE	022,000,007,055		;PACKED DECIMAL STRING 5
8737	027127	055							
8738									
8739	027130	022	000	007		CVTPN6: .BYTE	022,000,007,055		;PACKED DECIMAL STRING 6
8740	027133	055							
8741									
8742	027134	001	043	114		CVTPN7: .BYTE	001,043,114		;PACKED DECIMAL STRING 7
8743									
8744	027137	002	024	164		CVTPN8: .BYTE	002,024,164,203,144,215		;PACKED DECIMAL STRING 8
8745	027142	203	144	215					

8746										
8747	027145	061	040	174	CVTPN9:	.BYTE	061,040,174			;PACKED DECIMAL STRING 9
8748										
8749										
8750										
8751					.EVEN					
8752										
8753	027150				TCVPNI:					; INLINE TEST CASE
8754	027150	060004				.WORD	060004			; SIGNED PACKED, LENGTH = 4
8755	027152	027160				.WORD	PN1			; POINTER TO PACKED SRC STRING
8756	027154	000004				.WORD	000004			; SIGNED ZONED, LENGTH = 4
8757	027156	064400				.WORD	DS.DST			; POINTER TO DST
8758										
8759	027160	001	043	113	PN1:	.BYTE	1,43,113			;-1234
8760		027164				.EVEN				

```
8761 .SBTTL CVTNP
8762 :*****
8763 :
8764 :   CVTNP TEST
8765 :*****
8766 :
8767 027164 INLINE <XCVTNP:>                                XCVTNP:
8768 027164
8769 027164                                     SCOPE
8770 027164 000004                               ;ITERATION COUNT
8771 027166 LET $TIMES := #20                               MOV      #20,$TIMES
8772 027166 012737 000020 000404                   ;GET TEST COUNT
8773 027174 LET COUNT := TCVTNP                               MOV      TCVTNP,COUNT
8774 027174 013737 030660 000470                   ;POINT TO FIRST TEST CASE
8775 027202 LET TSTCAS := #TCVTNP+2                               MOV      #TCVTNP+2,TSTCAS
8776 027202 012737 030662 000544
8777 :REPEAT
8778 027210 INLINE <$5055:>                                $5055:
8779 027210
8780
8781 :+
8782 : GET TEST DATA FROM TABLE
8783 :-
8784
8785 027210 LET RO := TSTCAS
8786 027210 013700 000544                               MOV      TSTCAS,RO
8787 027214 LET DATA00 := (RO)+                               ;SRC DATA TYPE AND LENGTH
8788 027214 012037 000562                               MOV      (RO)+,DATA00
8789 027220 LET DATA01 := (RO)+                               ;POINTER TO INTERMEDIATE SRC STRING
8790 027220 012037 000564                               MOV      (RO)+,DATA01
8791 027224 LET DATA02 := (RO)                               ;DST DATA TYPE AND LENGTH
8792 027224 011037 000566                               MOV      (RO),DATA02
8793
8794
8795
8796 :+
8797 : INITIALIZE THE DECIMAL BUFFERS AND CC.EXP.
8798 :-
8799 027230 CALL CLRDSB
8800 027230 004737 057040                               JSR      PC,CLRDSB
8801 027234 LET CC.EXP := #0
8802 027234 005037 000522                               CLR      CC.EXP
8803
8804
8805
8806 :+
8807 : LET SIGN0 REFLECT THE SIGN OF THE SOURCE STRING.
8808 :-
8809 027240 LET SIGN0 := @DATA01
8810 027240 017737 151320 000450                               MOV      @DATA01,SIGN0
8811
8812
8813 :+
8814 : PUT SRC MAGNITUDE INTO DSBUF1. DSBUF1 WILL BE THE SOURCE FOR THE
8815 : CVTNP INSTRUCTION.
8816 :-
```

```

8817
8818 027246          CALL CLRDSB
8819 027246 004737 057040
8820 027252          LET R0 := DATA01          ;GET ADDRESS OF INTERMEDIATE SRC. STRING
8821 027252 013700 000564          MOV          DATA01,R0
8822 027256          LET R0 := R0 + #2          ;GET PAST THE SIGN WORD
8823 027256 062700 000002          ADD          #2,R0
8824 027262          LET R1 := #DSBUF1
8825 027262 012701 064274          MOV          #DSBUF1,R1
8826 027266          LET DSLEN1 := DATA00 CLR.BY #70000 ;GET SRC.LEN
8827 027266 013737 000562 000514          MOV          DATA00,DSLEN1
8828 027274 042737 070000 000514          BIC          #70000,DSLEN1
8829 027302          INCR R2 FROM #1 TO DSLEN1 BY #1 ;TRANSFER THE MAGNITUDE
8830 027302 012702 000001          MOV          #1,R2
8831 027306 000401          BR          $51112
8832 027310          $51113:
8833 027310 005202          INC          R2
8834 027312          $51112:
8835 027312 020237 000514          CMP          R2,DSLEN1
8836 027316 003002          BGT          $51114
8837 027320          LET (R1)+ :B= (R0)+
8838 027320 112021          MOVB         (R0)+,(R1)+
8839 027322          ENDINC
8840 027322 000772
8841 027324
8842
8843
8844
8845          +-
8846          . PUT THE PROPER SIGN INTO DSBUF1, ACCORDING TO THE SOURCE DATA TYPE.
8847          :-
8848 027324          LET DATTYP := DATA00 CLR.BY #107777 ;GET SRC DATA TYPE
8849 027324 013737 000562 000464          MOV          DATA00,DATTYP
8850 027332 042737 107777 000464          BIC          #107777,DATTYP
8851
8852
8853          ;*** SIGNED ZONED ***
8854 027340          IF DATTYP EQ #0 THEN
8855 027340 005737 000464          TST          DATTYP
8856 027344 001014          BNE          $51115
8857 027346          IF DSLEN1 HI #0 THEN          ;SRC.LEN > 0
8858 027346 005737 000514          TST          DSLEN1
8859 027352 101411          BLOS         $51116
8860 027354          LET R1 := R1 - #1          ;POINT TO SIGN BYTE
8861 027354 005301          DEC          R1
8862 027356          IF SIGN0 EQ #0 THEN          ;SIGN IS +
8863 027356 005737 000450          TST          SIGN0
8864 027362 001003          BNE          $51117
8865 027364          LET (R1) :B= (R1) SET.BY #60
8866 027364 152711 000060          BISB         #60,(R1)
8867 027370          ELSE          ;SIGN IS -
8868 027370 000402          BR          $51120
8869 027372          $51117:
8870 027372          LET (R1) :B= (R1) SET.BY #160
8871 027372 152711 000160          BISB         #160,(R1)
8872 027376          ENDIF
    
```


8873	027376								\$51120:
8874	027376				ENDIF				\$51116:
8875	027376								\$51115:
8876	027376				ENDIF				
8877	027376								
8878									
8879									
8880					:*** UNSIGNED ZONED ***				
8881	027376				IF DATTYP EQ #10000 THEN				
8882	027376	023727	000464	010000				CMP	DATTYP,#10000
8883	027404	001006						BNE	\$51121
8884	027406				IF DSLEN1 HI #0 THEN				
8885	027406	005737	000514					TST	DSLEN1
8886	027412	101403						BLOS	\$51122
8887	027414				LET R1 := R1 - #1				
8888	027414	005301							;POINT TO SIGN BYTE
8889	027416				LET (R1) :B= (R1) SET.BY #60			DEC	R1
8890	027416	152711	000060					BISB	#60,(R1)
8891	027422				ENDIF				\$51122:
8892	027422								\$51121:
8893	027422				ENDIF				
8894	027422								
8895									
8896									
8897					:*** TRAILING OVERPUNCH ***				
8898	027422				IF DATTYP EQ #20000 THEN				
8899	027422	023727	000464	020000				CMP	DATTYP,#20000
8900	027430	001015						BNE	\$51123
8901	027432				IF DSLEN1 HI #0 THEN				
8902	027432	005737	000514					TST	DSLEN1
8903	027436	101412						BLOS	\$51124
8904	027440				LET R1 := R1 - #1				
8905	027440	005301							;POINT TO SIGN BYTE
8906	027442				LET R0 := (R1)			DEC	R1
8907	027442	011100						MOV	(R1),R0
8908	027444				IF SIGN0 EQ #-1 THEN				
8909	027444	023727	000450	177777				CMP	SIGN0,#-1
8910	027452	001002						BNE	\$51125
8911	027454				LET R0 := R0 + #10.				
8912	027454	062700	000012					ADD	#10.,R0
8913	027460				ENDIF				\$51125:
8914	027460								;PUT IN SIGN/DIGIT COMBINATION
8915	027460				LET (R1) :B= OVPNCH(R0)			MOV	OVPNCH(R0),(R1)
8916	027460	116011	056530						\$51124:
8917	027464				ENDIF				\$51123:
8918	027464								
8919	027464				ENDIF				
8920	027464								
8921									
8922									
8923					:*** LEADING OVERPUNCH ***				
8924	027464				IF DATTYP EQ #30000 THEN				
8925	027464	023727	000464	030000				CMP	DATTYP,#30000
8926	027472	001016						BNE	\$51126
8927	027474				IF DSLEN1 HI #0 THEN				
8928	027474	005737	000514					TST	DSLEN1

8985
 8986
 8987
 8988
 8989
 8990
 8991
 8992 027614
 8993 027614 013702 000566
 8994 027620 042702 070000
 8995 027624
 8996 027624 006202
 8997 027626
 8998 027626 012700 064442
 8999 027632 060200
 9000 027634
 9001 027634 013737 000566 000500
 9002 027642 042737 107777 000500
 9003
 9004
 9005
 9006
 9007
 9008
 9009 027650
 9010 027650 013737 000562 000514
 9011 027656 042737 070000 000514
 9012 027664
 9013 027664 013737 000566 000516
 9014 027672 042737 070000 000516
 9015 027700
 9016 027700 005737 000514
 9017 027704 001003
 9018 027706
 9019 027706 112710 000014
 9020 027712
 9021 027712 000414
 9022 027714
 9023 027714
 9024 027714 005777 150644
 9025 027720 001404
 9026 027722 023727 000464 010000
 9027 027730 001003
 9028 027732
 9029 027732
 9030 027732 112710 000014
 9031 027736
 9032 027736 000402
 9033 027740
 9034 027740
 9035 027740 112710 000015
 9036 027744
 9037 027744
 9038 027744
 9039 027744
 9040 027744

```

: +
: LOCATE THE END OF THE EXPECTED DESTINATION, WHERE THE SIGN
: GOES. GET THE DESTINATION DATA TYPE.
: -
    
```

```

LET R2 := DATA02 CLR.BY #70000 ;GET DST.LEN
MOV DATA02,R2
BIC #70000,R2
INLINE <ASR R2>
ASR R2
LET R0 := #DS.EXP + R2
MOV #DS.EXP,R0
ADD R2,R0
LET TEMP1 = DATA02 CLR.BY #107777
MOV DATA02,TEMP1
BIC #107777,TEMP1
    
```

```

: +
: PUT THE SIGN INTO THE EXPECTED DESTINATION.
: -
    
```

```

LET DSLEN1 := DATA00 CLR.BY #70000 ;GET SRC.LEN
MOV DATA00,DSLEN1
BIC #70000,DSLEN1
LET DSLEN2 := DATA02 CLR.BY #70000 ;GET DST.LEN
MOV DATA02,DSLEN2
BIC #70000,DSLEN2
IF DSLEN1 EQ #0 THEN ;SRC.LEN OR DST.LEN = 0
TST DSLEN1
BNE $51137
LET (R0) :B= #14
MOV #14,(R0)
ELSE ;SRC.LEN & DST.LEN > 0
BR $51140
$51137:
IF @DATA01 EQ #0 OR DATTYP EQ #10000 THEN
TST @DATA01
BEQ $51141
CMP DATTYP,#10000
BNE $51142
$51141:
;SIGN IS +
MOV #14,(R0)
$51142:
;SIGN IS -
MOV #15,(R0)
$51143:
$51140:
ENDIF
ENDIF
IF TEMP1 EQ #70000 THEN
    
```

9041	027744	023727	000500	070000					
9042	027752	001002							
9043	027754				LET (R0) :B= #17				
9044	027754	112710	000017						
9045	027760				ENDIF				
9046	027760								\$51144:
9047									
9048									
9049									
9050					:+				
9051					:-				
9052					PUT THE EXPECTED VALUE OF THE NBIT INTO CC.EXP				
9053	027760				IF DATTYP NE #10000 AND TEMP1 NE #70000 THEN				
9054	027760	023727	000464	010000					
9055	027766	001415							
9056	027770	023727	000500	070000					
9057	027776	001411							
9058	030000				IFB (R0) EQ #15 AND DSLEN2 GT #0 THEN				
9059	030000	121027	000015						
9060	030004	001006							
9061	030006	005737	000516						
9062	030012	003403							
9063	030014				LET CC.EXP := CC.EXP SET.BY #NBIT				
9064	030014	052737	000010	000522					
9065	030022				ENDIF				
9066	030022								\$51146:
9067	030022				ENDIF				
9068	030022								\$51145:
9069									
9070									
9071									
9072					:+				
9073					:-				
9074					PUT THE EXPECTED VALUE OF THE ZBIT INTO CC.EXP				
9075	030022				IF DSLEN1 EQ #0 OR DSLEN2 EQ #0 THEN				
9076	030022	005737	000514						
9077	030026	001403							
9078	030030	005737	000516						
9079	030034	001003							
9080	030036								
9081	030036				LET CC.EXP := CC.EXP SET.BY #ZBIT				
9082	030036	052737	000004	000522					
9083	030044				ENDIF				
9084	030044								\$51150:
9085									
9086									
9087									
9088					:+				
9089					:-				
9090					PUT THE EXPECTED VALUE OF THE VBIT INTO CC.EXP				
9091	030044				IF DSLEN1 HI DSLEN2 THEN				
9092	030044	023737	000514	000516					
9093	030052	101403							
9094	030054				LET CC.EXP := CC.EXP SET.BY #VBIT				
9095	030054	052737	000002	000522					
9096	030062				ENDIF				

```

9097 030062                                $51151:
9098
9099
9100
9101      ;+
9102      ; CALCULATE THE NUMBER OF CHARACTERS TO BE MOVED INTO DS.EXP;
9103      ; USE R0.
9104      ;-
9105 030062                                IF DSLEN2 HIS DSLEN1 THEN                ;DST.LEN >- SRC.LEN
9106 030062 023737 000516 000514          ;DST.LEN >- SRC.LEN          CMP      DSLEN2,DSLEN1
9107 030070 103403                          ;DST.LEN >- SRC.LEN          BLO     $51152
9108 030072                                LET R0 := DSLEN1
9109 030072 013700 000514
9110 030076                                ELSE                                     ;SRC.LEN > DST.LEN
9111 030076 000402                          ;SRC.LEN > DST.LEN          MOV     DSLEN1,R0
9112 030100                                BR      $51153
9113 030100                                LET R0 := DSLEN2
9114 030100 013700 000516                                MOV     DSLEN2,R0
9115 030104                                ENDIF
9116 030104                                $51153:
9117
9118
9119
9120      ;+
9121      ; MAKE R1 POINT TO THE END OF THE DESTINATION IN DS.EXP; MAKE
9122      ; R2 POINT TO THE END OF THE SOURCE IN THE TABLE OF TEST CASES.
9123      ;-
9124 030104                                LET R1 := DSLEN2
9125 030104 013701 000516                                MOV     DSLEN2,R1
9126 030110                                INLINE <ASR R1>
9127 030110 006201                                ASR     R1
9128 030112                                LET R1 := R1 + #DS.EXP
9129 030112 062701 064442                                ADD     #DS.EXP,R1
9130 030116                                LET R2 := DATA01 + DSLEN1 + #1
9131 030116 013702 000564                                MOV     DATA01,R2
9132 030122 063702 000514                                ADD     DSLEN1,R2
9133 030126 005202                                INC     R2
9134
9135
9136      ;+
9137      ; MOVE THE LEAST SIGNIFICANT BCD FROM THE SOURCE TO THE
9138      ; EXPECTED DESTINATION BUFFER, IN PACKED FORMAT.
9139      ;-
9140
9141 030130                                IF R0 GT #0 THEN
9142 030130 005700                                TST     R0
9143 030132 003405                                BLE     $51154
9144 030134                                LET R3 := (R2)
9145 030134 111203                                MOVB   (R2),R3
9146 030136                                INLINE <ASH #4,R3>
9147 030136 072327 000004                                ASH     #4,R3
9148 030142                                LET (R1) := (R1) SET.BY R3
9149 030142 150311                                BISB   R3,(R1)
9150 030144                                LET R0 := R0 - #1
9151 030144 005300                                ;DECREMENT COUNT          DEC     R0
9152 030146                                ENDIF
    
```

```

9153 030146                                     $51154:
9154
9155
9156
9157
9158
9159
9160
9161
9162 030146                                     WHILE RO GT #0 DO
9163 030146
9164 030146 005700
9165 030150 003417
9166
9167 030152                                     LET R1 := R1 - #1           ;DECREMENT DEST. POINTER
9168 030152 005301                                     LET R2 := R2 - #1           ;DECREMENT SRC. POINTER
9169 030154
9170 030154 005302                                     LET (R1) :B= (R2) CLR.BY #360 ;PUT BCD INTO LOWER NIBBLE
9171 030156
9172 030156 111211                                     MOV B (R2), (R1)
9173 030160 142711 000360                             BIC B #360, (R1)
9174 030164                                     LET RO := RO - #1           ;DECREMENT COUNT
9175 030164 005300
9176 030166                                     IF RO HI #0 THEN
9177 030166 005700
9178 030170 101406
9179 030172                                     LET R2 := R2 - #1           ;DECREMENT SRC. POINTER
9180 030172 005302
9181 030174                                     LET R3 :B= (R2)
9182 030174 111203                                     MOV B (R2), R3
9183 030176                                     INLINE <ASH #4, R3>
9184 030176 072327 000004                             ASH #4, R3
9185 030202                                     LET (R1) :B= (R1) SET.BY R3 ;PUT BCD INTO UPPER NIBBLE
9186 030202 150311                                     BIS B R3, (R1)
9187 030204                                     LET RO := RO - #1           ;DECREMENT COUNT
9188 030204 005300                                     DEC RO
9189 030206                                     ENDIF
9190 030206
9191 030206
9192 030206 000757                                     ENDDO
9193 030210
9194
9195
9196
9197
9198
9199
9200
9201 030210                                     LET RO : DATA00           ;SRC DATA TYPE AND LENGTH
9202 030210 013700 000562                             MOV DATA00, RO          ;SRC ADDRESS
9203 030214                                     LET R1 := #DSBUF1         ;SRC ADDRESS
9204 030214 012701 064274                             MOV #DSBUF1, R1          ;DST DATA TYPE AND LENGTH
9205 030220                                     LET R2 := DATA02         ;DST DATA TYPE AND LENGTH
9206 030220 013702 000566                             MOV DATA02, R2          ;DST ADDRESS
9207 030224                                     LET R3 := #DS.DST         ;DST ADDRESS
9208 030224 012703 064400                             MOV #DS.DST, R3
    
```

9209	030230				LET R4 := #0		
9210	030230	005004				CLR	R4
9211	030232				LET R5 := #-1		
9212	030232	012705	177777			MOV	#-1,R5
9213	030236				LET R0.EXP := #0		
9214	030236	005037	000524			CLR	R0.EXP
9215	030242				LET R1.EXP := #0		
9216	030242	005037	000526			CLR	R1.EXP
9217	030246				LET R2.EXP := R2		
9218	030246	010237	000530			MOV	R2,R2.EXP
9219	030252				LET R3.EXP := R3		
9220	030252	010337	000532			MOV	R3,R3.EXP
9221	030256				LET R4.EXP := #0		
9222	030256	005037	000534			CLR	R4.EXP
9223	030262				LET R5.EXP := #-1		
9224	030262	012737	177777	000536		MOV	#-1,R5.EXP
9225	030270				INLINE <CVTNP>		
9226	030270	076055				CVTNP	
9227							
9228					;;		
9229					;; CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
9230					;;		
9231					;;		
9232	030272				LET CC := PSW		
9233	030272	013737	177776	000520		MOV	PSW,CC
9234	030300				LET CC := CC CLR.BY #177760		
9235	030300	042737	177760	000520		BIC	#177760,CC
9236	030306				IF CC NE CC.EXP THEN		
9237	030306	023737	000520	000522		CMP	CC,CC.EXP
9238	030314	001402				BEQ	\$51160
9239	030316				CALL ERROR		
9240	030316	004737	060044			JSR	PC,ERROR
9241	030322				ENDIF		
9242	030322					\$51160:	
9243	030322				IF R0 NE R0.EXP THEN		
9244	030322	020037	000524			CMP	R0,R0.EXP
9245	030326	001402				BEQ	\$51161
9246	030330				CALL ERROR		
9247	030330	004737	060044			JSR	PC,ERROR
9248	030334				ENDIF		
9249	030334					\$51161:	
9250	030334				IF R1 NE R1.EXP THEN		
9251	030334	020137	000526			CMP	R1,R1.EXP
9252	030340	001402				BEQ	\$51162
9253	030342				CALL ERROR		
9254	030342	004737	060044			JSR	PC,ERROR
9255	030346				ENDIF		
9256	030346					\$51162:	
9257	030346				IF R2 NE R2.EXP THEN		
9258	030346	020237	000530			CMP	R2,R2.EXP
9259	030352	001402				BEQ	\$51163
9260	030354				CALL ERROR		
9261	030354	004737	060044			JSR	PC,ERROR
9262	030360				ENDIF		
9263	030360					\$51163:	
9264	030360				IF R3 NE R3.EXP THEN		

9321						
9322						
9323						
9324					:+ : DO ONE INLINE CASE - CVTNPI	
9325						
9326	030504				CALL CLRREG	
9327	030504	004737	056604			JSR PC,CLRREG
9328	030510				CALL CLRDSB	
9329	030510	004737	057040			JSR PC,CLRDSB
9330	030514				INLINE <CVTNPI>	
9331	030514	076155				CVTNPI
9332	030516				INLINE <.WORD TCVNPI>	
9333	030516	031076				.WORD TCVNPI
9334	030520				INLINE <.WORD TCVNPI+4>	
9335	030520	031102				.WORD TCVNPI+4
9336	030522				LET CC := PSW	
9337	030522	013737	177776	000520		MOV PSW,CC
9338	030530				LET CC := CC CLR.BY #177760	
9339	030530	042737	177760	000520		BIC #177760,CC
9340	030536				IF CC NE #0 THEN	
9341	030536	005737	000520			TST CC
9342	030542	001402				BEQ \$51173
9343	030544				CALL ERROR	
9344	030544	004737	060044			JSR PC,ERROR
9345	030550				ENDIF	
9346	030550					\$51173:
9347	030550				IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN	
9348	030550	005700				TST R0
9349	030552	001004				BNE \$51174
9350	030554	005701				TST R1
9351	030556	001002				BNE \$51174
9352	030560	005702				TST R2
9353	030562	001402				BEQ \$51175
9354	030564					\$51174:
9355	030564				CALL ERROR	
9356	030564	004737	060044			JSR PC,ERROR
9357	030570				ENDIF	
9358	030570					\$51175:
9359	030570				IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN	
9360	030570	005703				TST R3
9361	030572	001004				BNE \$51176
9362	030574	005704				TST R4
9363	030576	001002				BNE \$51176
9364	030600	005705				TST R5
9365	030602	001402				BEQ \$51177
9366	030604					\$51176:
9367	030604				CALL ERROR	
9368	030604	004737	060044			JSR PC,ERROR
9369	030610				ENDIF	
9370	030610					\$51177:
9371	030610				IFB DS.DST NE #124 THEN	
9372	030610	123727	064400	000124		CMPB DS.DST,#124
9373	030616	001402				BEQ \$51200
9374	030620				CALL ERROR	
9375	030620	004737	060044			JSR PC,ERROR
9376	030624				ENDIF	

9377	030624					
9378	030624					
9379	030624	123727	064401	000062		IFB DS.DST+1 NE #62 THEN
9380	030632	001402				
9381	030634					CALL ERROR
9382	030634	004737	060044			

\$51200:
CMPB DS.DST+1,#62
BEQ \$51201
JSR PC,ERROR

9383 030640
9384 030640
9385 030640
9386 030640 123727 064402 000037
9387 030646 001402
9388 030650
9389 030650 004737 060044
9390 030654
9391 030654
9392 030654
9393 030654 000137 032432
9394
9395
9396
9397
9398
9399
9400

ENDIF
IFB DS.DST+2 NE #37 THEN
CALL ERROR
ENDIF
INLINE <JMP CISSEND>

\$51201:
CMPB DS.DST+2,#37
BEQ \$51202
JSR PC,ERROR
\$51202:
JMP CISSEND

:+
: TABLE OF TEST CASES FOR CVTNP
:-

9401 030660 000010
9402
9403 030662 000000
9404 030664 030750
9405 030666 060000
9406
9407 030670 050000
9408 030672 030754
9409 030674 060000
9410
9411 030676 030000
9412 030700 030760
9413 030702 060037
9414
9415 030704 020023
9416 030706 030764
9417 030710 060000
9418
9419 030712 040010
9420 030714 031012
9421 030716 060020
9422
9423 030720 010011
9424 030722 031024
9425 030724 060006
9426
9427 030726 000017
9428 030730 031040
9429 030732 060017
9430
9431 030734 000006
9432 030736 031062
9433 030740 070006
9434
9435 030742 050000
9436 030744 031072
9437 030746 060002
9438

TCVTNP: .WORD 8. ;# OF TEST CASES FOR CVTNP
.WORD 0 ;SRC DATA TYPE AND LENGTH
.WORD CVTNP1 ;POINTER TO INTERMEDIATE SRC STRING
.WORD 60000 ;DST DATA TYPE AND LENGTH
.WORD 50000 ;TEST CASE NUMBER 2
.WORD CVTNP2
.WORD 60000
.WORD 30000 ;TEST CASE NUMBER 3
.WORD CVTNP3
.WORD 60037
.WORD 20023 ;TEST CASE NUMBER 4
.WORD CVTNP4
.WORD 60000
.WORD 40010 ;TEST CASE NUMBER 5
.WORD CVTNP5
.WORD 60020
.WORD 10011 ;TEST CASE NUMBER 6
.WORD CVTNP6
.WORD 60006
.WORD 17 ;TEST CASE NUMBER 7
.WORD CVTNP7
.WORD 60017
.WORD 6 ;TEST CASE NUMBER 8
.WORD CVTNP8
.WORD 70006
.WORD 50000 ;TEST CASE NUMBER 9
.WORD CVTNP9
.WORD 60002

```

9439
9440 030750 000000 CVTNP1: .WORD 0 ;SIGN WORD: 0 = +, -1 = -
9441 030752 000 .BYTE ;MAGNITUDE, ONE BCD PER BYTE
9442 030754 .EVEN
9443
9444 030754 177777 CVTNP2: .WORD -1 ;INTERMEDIATE SRC STRING 2
9445 030756 000 .BYTE
9446 030760 .EVEN
9447
9448 030760 177777 CVTNP3: .WORD -1 ;INTERMEDIATE SRC STRING 3
9449 030762 000 .BYTE
9450 030764 .EVEN
9451
9452 030764 177777 CVTNP4: .WORD -1 ;INTERMEDIATE SRC STRING 4
9453 030766 011 010 007 .BYTE 011,010,007,006,005,004,003,002,001,000,001,002,003,004,005,006
9454 030771 006 005 004
9455 030774 003 002 001
9456 030777 000 001 002
9457 031002 003 004 005
9458 031005 006
9459 031006 007 010 011 .BYTE 007,010,011
9460 031012 .EVEN
9461
9462 031012 000000 CVTNP5: .WORD 0 ;INTERMEDIATE SRC STRING 5
9463 031014 007 000 006 .BYTE 007,000,006,001,005,002,004,003
9464 031017 001 005 002
9465 031022 004 003
9466 .EVEN
9467
9468 031024 177777 CVTNP6: .WORD -1 ;INTERMEDIATE SRC STRING 6
9469 031026 001 003 005 .BYTE 001,003,005,007,011,002,004,006,010
9470 031031 007 011 002
9471 031034 004 006 010
9472 031040 .EVEN
9473
9474 031040 177777 CVTNP7: .WORD -1 ;INTERMEDIATE SRC STRING 7
9475 031042 005 005 007 .BYTE 005,005,007,003,010,002,001,011,004,007,006,000,000,007,001
9476 031045 003 010 002
9477 031050 001 011 004
9478 031053 007 006 000
9479 031056 000 007 001
9480 031062 .EVEN
9481
9482 031062 177777 CVTNP8: .WORD -1 ;INTERMEDIATE SRC STRING 8
9483 031064 007 005 003 .BYTE 007,005,003,001,004,002
9484 031067 001 004 002
9485 .EVEN
9486
9487 031072 000000 CVTNP9: .WORD 0 ;INTERMEDIATE SRC STRING 9
9488 031074 007 002 .BYTE 007,002
9489 .EVEN
9490
9491
9492
9493 031076 TCVNPI: .WORD 000005 ;INLINE TEST CASE
9494 031076 000005 .WORD 000005 ;SIGNED ZONED, LENGTH = 5
    
```

9495	031100	031106				.WORD	NP1		: POINTER TO SRC
9496	031102	070005				.WORD	070005		: UNSIGNED PACKED, LENGTH = 5
9497	031104	064400				.WORD	DS.DST		: POINTER TO DST
9498									
9499	031106	005	004	003	NP1:	.BYTE	5,4,3,2,161		:-54321
9500	031111	002	161						
9501		031114				.EVEN			

9502
9503
9504
9505
9506
9507
9508
9509
9510
9511
9512
9513
9514
9515
9516
9517
9518
9519
9520
9521
9522
9523
9524
9525
9526
9527
9528
9529
9530
9531
9532
9533
9534
9535
9536
9537
9538
9539
9540
9541
9542
9543
9544
9545
9546
9547
9548
9549
9550
9551
9552
9553
9554
9555
9556
9557

031114
031114

031114
031114 004737 057102
031120
031120 005037 000450
031124 005037 000514

031130
031130 005702
031132
031132 100003
031134
031134 012737 177777 000450
031142
031142

```

:*****
:SBTTL CONVERT LONG INTEGER TO INTERMEDIATE FORM ROUTINE
:*****
:* THIS ROUTINE CONVERTS A LONG INTEGER INTO
:* AN INTERMEDIATE FORM.
:* THE INPUT IS THE 31 BIT LONG INTEGER (PLUS THE
:* SIGN BIT) IN REGISTERS R2 AND R3:
:*
:*
:   R2  ?SIGN?-----HIGH?
:-----
:
:   R3  ?-----LOW?
:-----
:*
:* THE OUTPUT IS:
:* SIGN0 SET TO SIGN OF THE LONG INTEGER
:* (0 FOR +, 177777 FOR -); AND 10 BINARY
:* CODED DECIMAL (BCD) DIGITS PUT INTO MEMORY,
:* ONE DIGIT PER BYTE, STARTING AT DSBUF1,
:* MOST SIGNIFICANT DIGIT FIRST. DSLEN1 WILL
:* HAVE THE LENGTH (EXCLUDING LEADING ZEROES).
:* THE GENERAL PURPOSE REGISTERS ARE SAVED
:* AT THE BEGINNING OF THE ROUTINE AND RESTORED
:* AT THE END.
:* DSBUF1 DECIMAL STRING BUFFER SHOULD BE CLEAR BEFORE
:* ENTERING THIS ROUTINE.
    
```

```

ROUTINE CVTLI
CVTLI:
:
: +
: ROUTINE INITIALIZATION
: -
CALL SAVREG          ;SAVE THE GPR'S      JSR    PC,SAVREG
LET SIGN0 := #0
LET DSLEN1 := #0
:
: +
: TEST FOR SPECIAL CASES OF 0 VALUE OR MOST NEGATIVE VALUE
: -
INLINE <TST R2>
IFCOND MI THEN
    LET SIGN0 := #-1
ENDIF
$51205:
    
```

```
9558 031142          LET TEMPO := R2 CLR.BY #100000
9559 031142 010237 000476          MOV R2,TEMPO
9560 031146 042737 100000 000476          BIC #100000,TEMPO
9561 031154          IF R3 EQ #0 AND TEMPO EQ #0 THEN
9562 031154 005703          TST R3
9563 031156 001044          BNE $51206
9564 031160 005737 000476          TST TEMPO
9565 031164 001041          BNE $51206
9566 031166          IF SIGN0 EQ #0 THEN
9567 031166 005737 000450          TST SIGN0
9568 031172 001003          BNE $51207
9569          ;RETURN
9570 031174          INLINE <JMP XCVTLI>
9571 031174 000137 032016          JMP XCVTLI
9572 031200          ELSE ;RESULT = -2,147,483,648
9573 031200 000433          BR $51210
9574 031202          ;$51207:
9575 031202          LET R0 := #DSBUF1
9576 031202 012700 064274          MOV #DSBUF1,R0
9577 031206          LET (R0)+ :B= #2
9578 031206 112720 000002          MOV3 #2,(R0)+
9579 031212          LET (R0)+ :B= #1
9580 031212 112720 000001          MOV8 #1,(R0)+
9581 031216          LET (R0)+ :B= #4
9582 031216 112720 000004          MOV8 #4,(R0)+
9583 031222          LET (R0)+ :B= #7
9584 031222 112720 000007          MOV8 #7,(R0)+
9585 031226          LET (R0)+ :B= #4
9586 031226 112720 000004          MOV8 #4,(R0)+
9587 031232          LET (R0)+ :B= #8.
9588 031232 112720 000010          MOV8 #8.,(R0)+
9589 031236          LET (R0)+ :B= #3
9590 031236 112720 000003          MOV8 #3,(R0)+
9591 031242          LET (R0)+ :B= #6
9592 031242 112720 000006          MOV8 #6,(R0)+
9593 031246          LET (R0)+ :B= #4
9594 031246 112720 000004          MOV8 #4,(R0)+
9595 031252          LET (R0)+ :B= #8.
9596 031252 112720 000010          MOV8 #8.,(R0)+
9597 031256          LET DSLEN1 := #10.
9598 031256 012737 000012 000514          MOV #10.,DSLEN1
9599          ;RETURN
9600 031264          INLINE <JMP XCVTLI>
9601 031264 000137 032016          JMP XCVTLI
9602 031270          ENDIF
9603 031270          ;$51210:
9604 031270          ENDIF ;$51206:
9605 031270
9606
9607          ;+
9608          ; LONG INTEGER SOURCE WAS NOT ZERO OR MOST NEGATIVE NUMBER,
9609          ; SO PREPARE FOR CONVERSION ALGORITHM.
9610          ;-
9611
9612 031270          IF SIGN0 EQ #-1 THEN
9613 031270 023727 000450 177777          CMP SIGN0,#-1
```

9614	031276	001005			BNE	\$51211
9615	031300		LET R3 := NEGATE R3		NEG	R3
9616	031300	005403			COM	R2
9617	031302		LET R2 := COMP R2		TST	R3
9618	031302	005102			BNE	\$51212
9619	031304		IF R3 EQ #0 THEN		INC	R2
9620	031304	005703				
9621	031306	001001		LET R2 := R2 + #1		
9622	031310			ENDIF		\$51212.
9623	031310	005202				
9624	031312			ENDIF		\$51211:
9625	031312					
9626	031312			LET R0 := #DSBUF1		
9627	031312					
9628	031312				MOV	#DSBUF1,R0
9629	031312	012700	064274			
9630						
9631				:+		
9632				: GET TENTH BCD DIGIT (MOST SIGNIFICANT DIGIT)		
9633				:-		
9634						
9635	031316			WHILE R2 HI #35632 DO		\$51213:
9636	031316				CMP	R2,#35632
9637	031316	020227	035632		BLOS	\$51214
9638	031322	101407			SUB	#145000,R3
9639	031324			LEI R3 := R3 - #145000		
9640	031324	162703	145000		SBC	R2
9641	031330			INLINE <SBC R2>		
9642	031330	005602			SUB	#35632,R2
9643	031332			LET R2 := R2 - #35632		
9644	031332	162702	035632		INCB	(R0)
9645	031336			LET (R0) :=B= (R0) + #1		
9646	031336	105210			BR	\$51213
9647	031340			ENDDO		\$51214:
9648	031340	000766			CMP	R2,#35632
9649	031342			IF R2 EQ #35632 AND R3 HIS #145000 THEN		\$51215
9650	031342				BNE	\$51215
9651	031342	020227	035632		CMP	R3,#145000
9652	031346	001007			BLO	\$51215
9653	031350	020327	145000		SUB	#145000,R3
9654	031354	103404			CLR	R2
9655	031356			LET R3 := R3 - #145000		
9656	031356	162703	145000		INCB	(R0)
9657	031362			LET R2 := #0		
9658	031362	005002				
9659	031364			LET (R0) :=B= (R0) + #1		
9660	031364	105210				\$51215:
9661	031366			ENDIF		
9662	031366					
9663						
9664				:+		
9665				: GET NINTH BCD DIGIT		
9666				:-		
9667						
9668	031366			LET R0 := R0 + #1		
9669	031366	005200			INC	R0


```

9670 031370
9671 031370
9672 031370 020227 002765
9673 031374 101407
9674 031376
9675 031376 162703 160400
9676 031402
9677 031402 005602
9678 031404
9679 031404 162702 002765
9680 031410
9681 031410 105210
9682 031412
9683 031412 000766
9684 031414
9685 031414
9686 031414 020227 002765
9687 031420 001007
9688 031422 020327 160400
9689 031426 103404
9690 031430
9691 031430 162703 160400
9692 031434
9693 031434 005002
9694 031436
9695 031436 105210
9696 031440
9697 031440
9698
9699
9700
9701
9702
9703 031440
9704 031440 005200
9705 031442
9706 031442
9707 031442 020227 000230
9708 031446 101407
9709 031450
9710 031450 162703 113200
9711 031454
9712 031454 005602
9713 031456
9714 031456 162702 000230
9715 031462
9716 031462 105210
9717 031464
9718 031464 000766
9719 031466
9720 031466
9721 031466 020227 000230
9722 031472 001007
9723 031474 020327 113200
9724 031500 103404
9725 031502

```

```

WHILE R2 HI #2765 DO
    LET R3 := R3 - #160400
    INLINE <SBC R2>
    LET R2 := R2 - #2765
    LET (R0) :B= (R0) + #1
ENDDO
IF R2 EQ #2765 AND R3 HIS #160400 THEN
    LET R3 := R3 - #160400
    LET R2 := #0
    LET (R0) :B= (R0) + #1
ENDIF
: +
: GET EIGHTH BCD DIGIT
: -
LET R0 := R0 + #1
WHILE R2 HI #230 DO
    LET R3 := R3 - #113200
    INLINE <SBC R2>
    LET R2 := R2 - #230
    LET (R0) :B= (R0) + #1
ENDDO
IF R2 EQ #230 AND R3 HIS #113200 THEN
    LET R3 := R3 - #113200

```

```

$51216:
CMP R2,#2765
BLOS $51217
SUB #160400,R3
SBC R2
SUB #2765,R2
INCB (R0)
BR $51216
$51217:
CMP R2,#2765
BNE $51220
CMP R3,#160400
BLO $51220
SUB #160400,R3
CLR R2
INCB (R0)
$51220:
INC R0
$51221:
CMP R2,#230
BLOS $51222
SUB #113200,R3
SBC R2
SUB #230,R2
INCB (R0)
BR $51221
$51222:
CMP R2,#230
BNE $51223
CMP R3,#113200
BLO $51223

```

9726	031502	162703	113200					SUB	#113200,R3
9727	031506			LET R2 := #0				CLR	R2
9728	031506	005002		LET (R0) := (R0) + #1				INCB	(R0)
9729	031510			ENDIF				\$51223:	
9730	031510	105210							
9731	031512								
9732	031512								
9733									
9734				:+					
9735				:+ GET SEVENTH BCD DIGIT					
9736				:-					
9737									
9738	031512			LET R0 := R0 + #1				INC	R0
9739	031512	005200		WHILE R2 HI #17 DO				\$51224:	
9740	031514							CMP	R2,#17
9741	031514							BLOS	\$51225
9742	031514	020227	000017	LET R3 := R3 - #41100				SUB	#41100,R3
9743	031520	101407		INLINE <SBC R2>				SBC	R2
9744	031522			LET R2 := R2 - #17				SUB	#17,R2
9745	031522	162703	041100	LET (R0) := (R0) + #1				INCB	(R0)
9746	031526			ENDDO				BR	\$51224
9747	031526	005602						\$51225:	
9748	031530			IF R2 EQ #17 AND R3 HIS #41100 THEN				CMP	R2,#17
9749	031530	162702	000017					BNE	\$51226
9750	031534							CMP	R3,#41100
9751	031534	105210		LET R3 := R3 - #41100				BLO	\$51226
9752	031536			LET R2 := #0				SUB	#41100,R3
9753	031536	000766		LET (R0) := (R0) + #1				CLR	R2
9754	031540			ENDIF				INCB	(R0)
9755	031540							\$51226:	
9756	031540	020227	000017						
9757	031544	001007		:+					
9758	031546	020327	041100	:+ GET SIXTH BCD DIGIT					
9759	031552	103404		:-					
9760	031554			LET R0 := R0 + #1				INC	R0
9761	031554	162703	041100	WHILE R2 HI #1 DO				\$51227:	
9762	031560							CMP	R2,#1
9763	031560	005002		LET R3 := R3 - #103240				BLOS	\$51230
9764	031562			INLINE <SBC R2>				SUB	#103240,R3
9765	031562	105210							
9766	031564								
9767	031564								
9768									
9769									
9770									
9771									
9772									
9773	031564								
9774	031564	005200							
9775	031566								
9776	031566								
9777	031566	020227	000001						
9778	031572	101406							
9779	031574								
9780	031574	162703	103240						
9781	031600								

9782 031600 005602
 9783 031602
 9784 031602 005302
 9785 031604
 9786 031604 105210
 9787 031606
 9788 031606 000767
 9789 031610
 9790 031610
 9791 031610 020227 000001
 9792 031614 001007
 9793 031616 020327 103240
 9794 031622 103404
 9795 031624
 9796 031624 162703 103240
 9797 031630
 9798 031630 005002
 9799 031632
 9800 031632 105210
 9801 031634
 9802 031634
 9803
 9804
 9805
 9806
 9807
 9808 031634
 9809 031634 005200
 9810 031636
 9811 031636
 9812 031636 020227 000001
 9813 031642 001005
 9814 031644
 9815 031644 162703 023420
 9816 031650
 9817 031650 005602
 9818 031652
 9819 031652 105210
 9820 031654
 9821 031654 000770
 9822 031656
 9823 031656
 9824 031656
 9825 031656 020327 023420
 9826 031662 103404
 9827 031664
 9828 031664 162703 023420
 9829 031670
 9830 031670 105210
 9831 031672
 9832 031672 000771
 9833 031674
 9834
 9835
 9836
 9837

```

    LET R2 := R2 - #1
    LET (R0) := (R0) + #1
  ENDDO

  IF R2 EQ #1 AND R3 HIS #103240 THEN

    LET R3 := R3 - #103240
    LET R2 := #0
    LET (R0) := (R0) + #1
  ENDIF

  :+
  : GET FIFTH BCD DIGIT
  :-

  LET R0 := R0 + #1
  WHILE R2 EQ #1 DO

    LET R3 := R3 - #23420
    INLINE <SBC R2>
    LET (R0) := (R0) + #1
  ENDDO

  WHILE R3 HIS #23420 DO

    LET R3 := R3 - #23420
    LET (R0) := (R0) + #1
  ENDDO

  :+
  : GET FOURTH BCD DIGIT
  :-
  
```

```

SBC R2
DEC R2
INCB (R0)
BR $51227
$51230:
CMP R2,#1
BNE $51231
CMP R3,#103240
BLO $51231
SUB #103240,R3
CLR R2
INCB (R0)
$51231:

INC R0
$51232:
CMP R2,#1
BNE $51233
SUB #23420,R3
SBC R2
INCB (R0)
BR $51232
$51233:
$51234:
CMP R3,#23420
BLO $51235
SUB #23420,R3
INCB (R0)
BR $51234
$51235:
  
```

9838				LET R0 := R0 + #1	
9839	031674				INC R0
9840	031674	005200		WHILE R3 HIS #1750 DO	\$51236:
9841	031676				CMP R3,#1750
9842	031676				BLO \$51237
9843	031676	020327	001750	LET R3 := R3 - #1750	SUB #1750,R3
9844	031702	103404		LET (R0) :B= (R0) + #1	INCB (R0)
9845	031704			ENDDO	BR \$51236
9846	031704	162703	001750		\$51237:
9847	031710				
9848	031710	105210			
9849	031712				
9850	031712	000771			
9851	031714				
9852					
9853				:+	
9854				:+ GET THIRD BCD DIGIT	
9855				:+	
9856				:-	
9857	031714			LET R0 := R0 + #1	
9858	031714	005200			INC R0
9859	031716			WHILE R3 HIS #144 DO	\$51240:
9860	031716				CMP R3,#144
9861	031716	020327	000144		BLO \$51241
9862	031722	103404		LEI R3 :- R3 - #144	SUB #144,R3
9863	031724			LET (R0) :B= (R0) + #1	INCB (R0)
9864	031724	162703	000144	ENDDO	BR \$51240
9865	031730				\$51241:
9866	031730	105210			
9867	031732				
9868	031732	000771			
9869	031734				
9870					
9871				:+	
9872				:+ GET SECOND AND FIRST BCD DIGITS	
9873				:+	
9874				:-	
9875	031734			LET R0 := R0 + #1	
9876	031734	005200			INC R0
9877	031736			WHILE R3 HIS #12 DO	\$51242:
9878	031736				CMP R3,#12
9879	031736	020327	000012		BLO \$51243
9880	031742	103404		LET R3 := R3 - #12	SUB #12,R3
9881	031744			LET (R0) :B= (R0) + #1	INCB (R0)
9882	031744	162703	000012	ENDDO	BR \$51242
9883	031750				\$51243:
9884	031750	105210			
9885	031752				
9886	031752	000771			
9887	031754				
9888	031754			LET R0 := R0 + #1	INC R0
9889	031754	005200			
9890	031756			LET (R0) :B= R3	MOVB R3,(R0)
9891	031756	110310			
9892					
9893				:+	

```
9894 ; DETERMINE THE NUMBER OF SIGNIFICANT DIGITS.
9895 :-
9896
9897 031760 LET R0 : #DSBUF1
9898 031760 012700 064274 MOV #DSBUF1,R0
9899 031764 WHILEB (R0) EQ #0 DO
9900 031764 $51244:
9901 031764 105710 TSTB (R0)
9902 031766 001002 BNE $51245
9903 031770 LET R0 := R0 + #1
9904 031770 005200 INC R0
9905 031772 ENDDO
9906 031772 000774 BR $51244
9907 031774 $51245:
9908 031774 LET R0 := RC - #DSBUF1 ;# OF LEADING 0'S
9909 031774 162700 064274 SUB #DSBUF1,R0
9910 032000 LET DSLEN1 := #10. - R0 ;# OF MOST SIGNIFICANT DIGITS
9911 032000 012737 000012 000514 MOV #10.,DSLEN1
9912 032006 160037 000514 SUB R0,DSLEN1
9913 032012 CALL RESREG ;RESTORE THE GPR'S
9914 032012 004737 057134 JSR PC,RESREG
9915 032016 INLINE <XCVTLI:>
9916 032016
9917 032016 ENDRTN
9918 032016
9919 032016 $51203:
9920 032016 000207 $51204:
9921 RTS PC
9922
9923 ;*****
9924 ;SBTTL CONVERT INTERMEDIATE FORM TO LONG INTEGER ROUTINE
9925 ;*****
9926 ;* THIS ROUTINE CONVERTS A 10 BCD DIGIT INTERMEDIATE FORM INTO
9927 ;* LONG INTEGER FORMAT. THE INPUT IS 10 BCD DIGITS IN THE 10 BYTES STARTING
9928 ;* AT DSBUF1, MOST SIGNIFICANT DIGIT FIRST; AND SIGN0.
9929 ;* THE OUTPUT IS A TWO'S COMPLEMENT 31 BIT BINARY LONG INTEGER IN LOCATIONS
9930 ;* R2.EXP AND R3.EXP.
9931 ;* THIS ROUTINE ASSUMES THAT THE 10 BCD DIGIT NUMBER IS SMALL ENOUGH
9932 ;* TO BE REPRESENTED AS A 31 BIT BINARY INTEGER.
9933
9934 032020 ROUTINE CVTIL
9935 032020 CVTIL:
9936
9937 ;+
9938 ; ROUTINE INITIALIZATION
9939 ;-
9940
9941 032020 CALL SAVREG ;SAVE THE GPR'S
9942 032020 004737 057102 JSR PC,SAVREG
9943 032024 LET R2.EXP :- #0
9944 032024 005037 000530 CLR R2.EXP
9945 032030 LET R3.EXP :- #0
9946 032030 005037 000532 CLR R3.EXP
9947 032034 LET R0 := #DSBUF1
9948 032034 012700 064274 MOV #DSBUF1,R0
9949
```

```
9950      ;+
9951      ;+
9952      ; ADD IN BINARY EQUIVALENT OF TENTH BCD DIGIT
9953      ; -
9954
9955      032040      WHILEB (R0) GT #0 DO
9956      032040
9957      032040      105710
9958      032042      003412
9959      032044
9960      032044      062737      145000      000532      LET R3.EXP := R3.EXP + #145000
9961      032052      005537      000530      INLINE <ADC R2.EXP>
9962      032052
9963      032056      LET R2.EXP := R2.EXP + #35632
9964      032056      062737      035632      000530
9965      032064      LET (R0) :B= (R0) - #1
9966      032064      105310
9967      032066
9968      032066      000764      ENDDO
9969      032070
9970
9971      ;+
9972      ; ADD IN BINARY EQUIVALENT OF NINTH BCD DIGIT
9973      ; -
9974
9975      032070      LET R0 := R0 + #1
9976      032070      005200
9977      032072      WHILEB (R0) GT #0 DO
9978      032072
9979      032072      105710
9980      032074      003412
9981      032076
9982      032076      062737      160400      000532      LET R3.EXP := R3.EXP + #160400
9983      032104      005537      000530      INLINE <ADC R2.EXP>
9984      032104
9985      032110      LET R2.EXP := R2.EXP + #2765
9986      032110      062737      002765      000530
9987      032116      LET (R0) :B= (R0) - #1
9988      032116      105310
9989      032120      ENDDO
9990      032120      000764
9991      032122
9992
9993      ;+
9994      ; ADD IN BINARY EQUIVALENT OF EIGHTH BCD DIGIT
9995      ; -
9996
9997      032122      LET R0 := R0 + #1
9998      032122      005200
9999      032124      WHILEB (R0) GT #0 DO
10000     032124
10001     032124      105710
10002     032126      003412
10003     032130
10004     032130      062737      113200      000532      LET R3.EXP := R3.EXP + #113200
10005     032136      005537      000530      INLINE <ADC R2.EXP>
```

```
$51250:
TSTB (R0)
BLE $51251
ADD #145000,R3.EXP
ADC R2.EXP
ADD #35632,R2.EXP
DECB (R0)
BR $51250
$51251:
```

```
INC R0
$51252:
TSTB (R0)
BLE $51253
ADD #160400,R3.EXP
ADC R2.EXP
ADD #2765,R2.EXP
DECB (R0)
BR $51252
$51253:
```

```
INC R0
$51254:
TSTB (R0)
BLE $51255
ADD #113200,R3.EXP
```

```

10006 032136 005537 000530          ADC R2.EXP
10007 032142          LET R2.EXP := R2.EXP + #230      ADD #230,R2.EXP
10008 032142 062737 000230 000530  LET (R0) :B= (R0) - #1      DECB (R0)
10009 032150          ENDDO
10010 032150 105310          BR $51254
10011 032152          $51255:
10012 032152 000764          ;+
10013 032154          ;+ ADD IN BINARY EQUIVALENT OF SEVENTH BCD DIGIT
10014          ;+
10015          ;+
10016          ;+
10017          ;+
10018          ;+
10019 032154          LET R0 := R0 + #1
10020 032154 005200          WHILEB (R0) GT #0 DO
10021 032156          INC R0
10022 032156          $51256:
10023 032156 105710          TSTB (R0)
10024 032160 003412          BLE $51257
10025 032162          LET R3.EXP := R3.EXP + #41100
10026 032162 062737 041100 000532  INLINE <ADC R2.EXP>
10027 032170          ADD #41100,R3.EXP
10028 032170 005537 000530          ADC R2.EXP
10029 032174          LET R2.EXP := R2.EXP + #17
10030 032174 062737 000017 000530  ADD #17,R2.EXP
10031 032202          LEI (R0) :B= (R0) - #1
10032 032202 105310          DECB (R0)
10033 032204          ENDDO
10034 032204 000764          BR $51256
10035 032206          $51257:
10036          ;+
10037          ;+
10038          ;+
10039          ;+
10040          ;+
10041 032206          LET R0 := R0 + #1
10042 032206 005200          WHILEB (R0) GT #0 DO
10043 032210          INC R0
10044 032210          $51260:
10045 032210 105710          TSTB (R0)
10046 032212 003411          BLE $51261
10047 032214          LET R3.EXP := R3.EXP + #103240
10048 032214 062737 103240 000532  INLINE <ADC R2.EXP>
10049 032222          ADD #103240,R3.EXP
10050 032222 005537 000530          ADC R2.EXP
10051 032226          LET R2.EXP := R2.EXP + #1
10052 032226 005237 000530          INC R2.EXP
10053 032232          LET (R0) :B= (R0) - #1
10054 032232 105310          DECB (R0)
10055 032234          ENDDO
10056 032234 000765          BR $51260
10057 032236          $51261:
10058          ;+
10059          ;+
10060          ;+
10061          ;+
  
```

```

10062
10063 032236          LET R0 := R0 + #1
10064 032236 005200   WHILEB (R0) GT #0 DO
10065 032240
10066 032240
10067 032240 105710
10068 032242 003407
10069 032244          LET R3.EXP := R3.EXP + #23420
10070 032244 062737 023420 000532   INLINE <ADC R2.EXP>
10071 032252          LET (R0) :B= (R0) - #1
10072 032252 005537 000530
10073 032256
10074 032256 105310
10075 032260
10076 032260 000767
10077 032262
10078
10079
10080 :+
10081 : ADD IN BINARY EQUIVALENT OF FOURTH BCD DIGIT
10082 :-
10083 032262          LET R0 :- R0 + #1
10084 032262 005200   WHILEB (R0) GT #0 DO
10085 032264
10086 032264
10087 032264 105710
10088 032266 003407
10089 032270          LET R3.EXP := R3.EXP + #1750
10090 032270 062737 001750 000532   INLINE <ADC R2.EXP>
10091 032276          LET (R0) :B- (R0) - #1
10092 032276 005537 000530
10093 032302
10094 032302 105310
10095 032304
10096 032304 000767
10097 032306
10098
10099
10100 :+
10101 : ADD IN BINARY EQUIVALENT OF THIRD BCD DIGIT
10102 :-
10103 032306          LET R0 := R0 + #1
10104 032306 005200   WHILEB (R0) GT #0 DO
10105 032310
10106 032310
10107 032310 105710
10108 032312 003407
10109 032314          LET R3.EXP := R3.EXP + #144
10110 032314 062737 000144 000532   INLINE <ADC R2.EXP>
10111 032322          LET (R0) :B= (R0) - #1
10112 032322 005537 000530
10113 032326
10114 032326 105310
10115 032330
10116 032330 000767
10117 032332

```

```

INC R0
$51262:
TSTB (R0)
BLE $51263
ADD #23420,R3.EXP
ADC R2.EXP
DECB (R0)
BR $51262
$51263:

```

```

INC R0
$51264:
TSTB (R0)
BLE $51265
ADD #1750,R3.EXP
ADC R2.EXP
DECB (R0)
BR $51264
$51265:

```

```

INC R0
$51266:
TSTB (R0)
BLE $51267
ADD #144,R3.EXP
ADC R2.EXP
DECB (R0)
BR $51266
$51267:

```


10118							
10119							
10120							
10121							
10122							
10123	032332						
10124	032332	005200					
10125	032334						
10126	032334						
10127	032334	105710					
10128	032336	003407					
10129	032340						
10130	032340	062737	000012	000532			
10131	032346						
10132	032346	005537	000530				
10133	032352						
10134	032352	105310					
10135	032354						
10136	032354	000767					
10137	032356						
10138							
10139							
10140							
10141							
10142							
10143	032356						
10144	032356	005200					
10145	032360						
10146	032360	000310					
10147	032362						
10148	032362	061037	000532				
10149	032366						
10150	032366	005537	000530				
10151	032372						
10152	032372	023727	000450	177777			
10153	032400	001011					
10154	032402						
10155	032402	005437	000532				
10156	032406						
10157	032406	005137	000530				
10158	032412						
10159	032412	005737	000532				
10160	032416	001002					
10161	032420						
10162	032420	005237	000530				
10163	032424						
10164	032424						
10165	032424						
10166	032424						
10167	032424						
10168	032424	004737	057134				
10169	032430						
10170	032430						
10171	032430						
10172	032430	000207					
10173							

```

:+
: ADD IN BINARY EQUIVALENT OF SECOND BCD DIGIT
:-

```

```
LET R0 := R0 + #1
```

```
WHILEB (R0) GT #0 DO
```

```
LET R3.EXP := R3.EXP + #12
```

```
INLINE <ADC R2.EXP>
```

```
LET (R0) :B= (R0) - #1
```

```
ENDDO
```

```

:+
: ADD IN FIRST DIGIT
:-

```

```
LET R0 := R0 + #1
```

```
INLINE <SWAB (R0)>
```

```
LET R3.EXP := R3.EXP + (R0)
```

```
INLINE <ADC R2.EXP>
```

```
IF SIGN0 EQ #-1 THEN
```

```
LET R3.EXP := NEGATE R3.EXP
```

```
LET R2.EXP := COMP R2.EXP
```

```
IF R3.EXP EQ #0 THEN
```

```
LET R2.EXP := R2.EXP + #1
```

```
ENDIF
```

```
ENDIF
```

```
CALL RESREG
```

```
;RESTORE GPR'S
```

```
ENDRTN
```

```
INC R0
```

```
$51270:
```

```
TSTB (R0)
```

```
BLE $51271
```

```
ADD #12,R3.EXP
```

```
ADC R2.EXP
```

```
DECB (R0)
```

```
BR $51270
```

```
$51271:
```

```
INC R0
```

```
SWAB (R0)
```

```
ADD (R0),R3.EXP
```

```
ADC R2.EXP
```

```
CMP SIGN0,#-1
```

```
BNE $51272
```

```
NEG R3.EXP
```

```
COM R2.EXP
```

```
TST R3.EXP
```

```
BNE $51273
```

```
INC R2.EXP
```

```
$51273:
```

```
$51272:
```

```
JSR PC,RESREG
```

```
$51246:
```

```
$51247:
```

```
RTS PC
```

10174 032432
10175 032432
10176
10177

INLINE <CISSEND:>

CISSEND:



10178
10179
10180
10181
10182
10183
10184
10185
10186
10187
10188
10189
10190
10191
10192
10193
10194
10195
10196
10197
10198
10199
10200
10201
10202
10203
10204
10205
10206
10207
10208
10209
10210
10211
10212
10213
10214
10215
10216
10217
10218
10219
10220
10221
10222
10223
10224
10225
10226
10227
10228

032432
032432 000004
032434
032434 017737 145766 000476
032442 042737 177700 000476
032450
032450 005737 000476
032454 001406
032456
032456 032737 000004 000476
032464 001002
032466
032466 000137 045230
032472
032472
032472
032472
032472
032472 012737 000006 000546
032500
032500 005037 000456
032504
032504 012737 000340 177776
032512
032512 012737 000001 000552

```
*****
*****
++
      C I S 6   M O D U L E
:
:   THIS MODULE TESTS THE NUMERIC DATA TYPE ARITHMETIC
:   INSTRUCTIONS: ADDN, SUBN, CMF I, AND ASHN.
:
:--
*****
*****

.SBTTL  CIS6 INITIALIZATION
*****
:
:   CIS6 INITIALIZATION
:
:*****
:
:IF SWR<5:0> ARE CLEARED OR IF SWR<2> IS SET THEN EXECUTE
: THE CIS6 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.
:
:
:   INLINE  <SCOPE>
:
:
:   LET TEMPO := @SWR CLR.BY #177700
:
:   MOV     @SWR,TEMPO
:   BIC     #177700,TEMPO
:
:   IF TEMPO NE #0 THEN
:
:       TST     TEMPO
:       BEQ     $51274
:
:       IF #BIT2 NOTSETIN TEMPO THEN
:
:           BIT     #BIT2,TEMPO
:           BNE     $51275
:
:           INLINE  <JMP  CIS6END>
:
:           JMP     CIS6END
:       ENDIF
:   ENDIF
:
:   $51275:
:
:   $51274:
:
:   LET CTL := #6           ;CONTROL CHIP NUMBER
:
:   MOV     #6,CTL
:
:   LET ERRFLG := #0       ;INITIALIZE ERROR FLAG
:
:   CLR     ERRFLG
:
:   LET PSW := #340
:
:   MOV     #340,PSW
:
:   LET ERR := #1
:
:   MOV     #1,ERR
```

```

10229 .SBTTL ADDN
10230 :*****
10231 :
10232 :       ADDN 'EST
10233 :
10234 :*****
10235
10236 032520      INLINE <XADDN.>
10237 032520
10238 032520      INLINE <SCOPE>
10239 032520 000004      XADDN:
10240 032522      SCOPE
10241 032522 012737 000010 000404      LET $TIMES := #10      ;ITERATION COUNT      MOV      #10,$TIMES
10242 032530      LET COUNT := TADD      ;# OF TEST CASES
10243 032530 013737 065104 000470      LET TSTCAS := #TADD+2      ;POINT TO FIRST TEST CASE      MOV      TADD,COUNT
10244 032536      LET TSTCAS := #TADD+2      ;POINT TO FIRST TEST CASE      MOV      #TADD+2,TSTCAS
10245 032536 012737 065106 000544
10246
10247 :+
10248 :       CLR R0.EXT TO R3.EXP
10249 :-
10250
10251 032544      INCR R0 FROM #0 TO #6 BY #2
10252 032544 005000
10253 032546 000402      CLR      R0
10254 032550      BR      $51276
10255 032550 062700 000002      $51277:
10256 032554      ADD      #2,R0
10257 032554 020027 000006      $51276:
10258 032560 003003      CMP      R0,#6
10259 032562      BGT      $51300
10260 032562 005060 000524      INLINE <CLR R0.EXP(R0)>
10261 032566      CLR R0.EXP(R0)
10262 032566 000770      ENDINC
10263 032570
10264      ;REPEAT OUTER LOOP FOR TEST CASES
10265 032570      INLINE <$5050:>
10266 032570
10267 032570      CALL CLRDSB      ;CLR DS BUFFERS
10268 032570 004737 057040      JSR      PC,CLRDSB
10269
10270 :+
10271 :       LOAD DATAXX AND CC.EXP FROM TEST CASE
10272 :-
10273
10274 032574      LET R0 := TSTCAS
10275 032574 013700 000544      MOV      TSTCAS,R0
10276 032600      LET DATA00 := (R0)+      ;SRC1 LENGTH      MOV      (R0)+,DATA00
10277 032600 012037 000562      LET DATA01 := (R0)+      ;POINTER TO SRC1 DATA      MOV      (R0)+,DATA01
10278 032604      LET DATA02 := (R0)+      ;SRC2 LENGTH      MOV      (R0)+,DATA02
10279 032604 012037 000564      LET DATA03 := (R0)+      ;POINTER TO SRC2 DATA      MOV      (R0)+,DATA03
10280 032610      LET DATA04 := (R0)+      ;DEST LENGTH      MOV      (R0)+,DATA04
10281 032610 012037 000566
10282 032614
10283 032614 012037 000570
10284 032620
    
```

10285	032620	012037	000572			MOV	(R0)+,DATA04
10286	032624			LET DATA05 := (R0)+	: POINTER TO DS.EXP DATA (SUM)	(SUM)	
10287	032624	012037	000574			MOV	(R0)+,DATA05
10288	032630			LET CC.EXP := (R0)+	: CC.EXP (SUM)		
10289	032630	012037	000522			MOV	(R0)+,CC.EXP
10290							
10291				:+			
10292				::	LOAD DSBUF1		
10293				:-			
10294							
10295	032634			IF DATA00 NE #0 THEN	: CHECK FOR ZERO LENGTH BUFFER		
10296	032634	005737	000562			TST	DATA00
10297	032640	001413				BEQ	\$51301
10298	032642			LET R0 := DATA01			
10299	032642	013700	000564			MOV	DATA01,R0
10300	032646			LET SIGN0 := (R0)+	: STORE SIGN AND POINT AT FIRST BYTE		
10301	032646	012037	000450			MOV	(R0)+,SIGN0
10302	032652			LET R1 := DATA00	: # OF BYTES TO TRANSFER		
10303	032652	013701	000562			MOV	DATA00,R1
10304	032656			LET R2 := #DSBUF1	: TRANSFER DESTINATION		
10305	032656	012702	064274			MOV	#DSBUF1,R2
10306	032662			INLINE <1\$:>			1\$:
10307	032662						
10308	032662			INLINE <MOVB (R0)+,(R2)+>			MOVB (R0)+,(R2)+
10309	032662	112022					
10310	032664			INLINE <SOB R1,1\$>			SOB R1,1\$
10311	032664	077102					
10312	032666			ELSE			
10313	032666	000402				BR	\$51302
10314	032670					\$51301:	
10315	032670			LET SIGN0 := #0			
10316	032670	005037	000450			CLR	SIGN0
10317	032674			ENDIF			
10318	032674					\$51302:	
10319							
10320				:+			
10321				::	LOAD DSBUF2		
10322				:-			
10323							
10324	032674			IF DATA02 NE #0 THEN			
10325	032674	005737	000566			TST	DATA02
10326	032700	001413				BEQ	\$51303
10327	032702			LET R0 := DATA03			
10328	032702	013700	000570			MOV	DATA03,R0
10329	032706			LET SIGN1 := (R0)+			
10330	032706	012037	000452			MOV	(R0)+,SIGN1
10331	032712			LET R1 := DATA02			
10332	032712	013701	000566			MOV	DATA02,R1
10333	032716			LET R2 := #DSBUF2			
10334	032716	012702	064336			MOV	#DSBUF2,R2
10335	032722			INLINE <2\$:>			2\$:
10336	032722						
10337	032722			INLINE <MOVB (R0)+,(R2)+>			MOVB (R0)+,(R2)+
10338	032722	112022					
10339	032724			INLINE <SOB R1,2\$>			SOB R1,2\$
10340	032724	077102					

10341	032726			ELSE				
10342	032726	000402					BR	\$51304
10343	032730						\$51303:	
10344	032730			LET SIGN1 := #0			CLR	SIGN1
10345	032730	005037	000452				\$51304:	
10346	032734			ENDIF				
10347	032734							
10348								
10349				:+				
10350				:+ LOAD DS.EXP				
10351				:-				
10352								
10353	032734			IF DATA04 NE #0 THEN				
10354	032734	005737	000572				TST	DATA04
10355	032740	001415					BEQ	\$51305
10356	032742			LET R0 := DATA05			MOV	DATA05,R0
10357	032742	013700	000574				MOV	(R0)+,SIGN2
10358	032746			LET SIGN2 := (R0)+			MOV	DATA04,R1
10359	032746	012037	000454				MOV	#DS.EXP,R2
10360	032752			LET R1 := DATA04			3\$:	
10361	032752	013701	000572				MOV	(R0)+,(R2)
10362	032756			LET R2 := #DS.EXP			BISB	#60,(R2)+
10363	032756	012702	064442				SOB	R1,3\$
10364	032762			INLINE <3\$:>				
10365	032762			INLINE <MOV B (R0)+,(R2)>				
10366	032762			INLINE <BIS B #60,(R2)+>				
10367	032762	112012						
10368	032764			INLINE <SOB R1,3\$>				
10369	032764	152722	000060					
10370	032770			ELSE			BR	\$51306
10371	032770	077104					\$51305:	
10372	032772						CLR	SIGN2
10373	032772	000402		LET SIGN2 := #0			\$51306:	
10374	032774			ENDIF				
10375	032774							
10376	032774	005037	000454					
10377	033000							
10378	033000							
10379								
10380				:+				
10381				:+ LOAD DSTMP'S WITH MDS & LSD OF EACH BUFFER				
10382				:-				
10383								
10384	033000			LET DSTMP0 :B= DSBUF1			MOV B	DSBUF1,DSTMP0
10385	033000	113737	064274	000554			MOV	#DSBUF1-1,TEMPO
10386	033006			LET TEMPO := #DSBUF1-1 + DATA00			ADD	DATA00,TEMPO
10387	033006	012737	064273	000476			MOV B	@TEMPO,DSTMP1
10388	033014	063737	000562	000476			MOV B	DSBUF2,DSTMP2
10389	033022			LET DSTMP1 :B= @TEMPO			MOV	#DSBUF2-1,TEMPO
10390	033022	117737	145450	000555			ADD	DATA02,TEMPO
10391	033030			LET DSTMP2 :B= DSBUF2				
10392	033030	113737	064336	000556				
10393	033036			LET TEMPO := #DSBUF2-1 + DATA02				
10394	033036	012737	064335	000476				
10395	033044	063737	000566	000476				
10396	033052			LET DSTMP3 :B= @TEMPO				

10397	033052	117737	145420	000557		MOVB	@TEMPO,DSTMP3
10398	033060				LET DSTMP4 :B= DS.EXP	MOVB	DS.EXP,DSTMP4
10399	033060	113737	064442	000560			
10400	033066				LET TEMPO := #DS.EXP-1 + DATA04		
10401	033066	012737	064441	000476		MOV	#DS.EXP-1,TEMPO
10402	033074	063737	000572	000476		ADD	DATA04,TEMPO
10403	033102				LET DSTMP5 :B= @TEMPO	MOVB	@TEMPO,DSTMP5
10404	033102	117737	145370	000561			
10405	033110				LET DESCR1 := #DSBUF1	MOV	#DSBUF1,DESCR1
10406	033110	012737	064274	000620			
10407	033116				LET DESCR3 := #DSBUF2	MOV	#DSBUF2,DESCR3
10408	033116	012737	064336	000624			
10409	033124				LET DESCR5 : #DS.DST	MOV	#DS.DST,DESCR5
10410	033124	012737	064400	000630			
10411							
10412					: INCR TEMPO FROM #1 TO #5 BY #1		SRC1 DATA TYPE LOOP
10413	033132				INLINE <CLR @TEMPO>		CLR @TEMPO
10414	033132	005037	000476				
10415	033136				INLINE <INC1: INC TEMPO>		INC1: INC TEMPO
10416	033136	005237	000476				
10417	033142				IF TEMPO GT #5 THEN		
10418	033142	023727	000476	000005		CMP	TEMPO,#5
10419	033150	003402				BLE	\$51307
10420	033152				INLINE <JMP END1>		JMP END1
10421	033152	000137	035022				
10422	033156				ENDIF		\$51307:
10423	033156						
10424							
10425					:+ RESTORE MSD & LSD OF DSBUF1		
10426					:+ RESTORE MSD & LSD OF DSBUF1		
10427					:+ RESTORE MSD & LSD OF DSBUF1		
10428					:+ RESTORE MSD & LSD OF DSBUF1		
10429	033156				LET DSBUF1 :B= DSTMP0 ;MSD		
10430	033156	113737	000554	064274		MOVB	DSTMP0,DSBUF1
10431	033164				LET IEMP2 := #DSBUF1-1 + DATA00		
10432	033164	012737	064273	000502		MOV	#DSBUF1-1,TEMP2
10433	033172	063737	000562	000502		ADD	DATA00,TEMP2
10434	033200				LET @TEMP2 :B= DSTMP1 ;LSD		
10435	033200	113777	000555	145274		MOVB	DSTMP1,@TEMP2
10436							
10437					:+ SIGN DSBUF1 BY DATA TYPE		
10438					:+ SIGN DSBUF1 BY DATA TYPE		
10439					:+ SIGN DSBUF1 BY DATA TYPE		
10440					:+ SIGN DSBUF1 BY DATA TYPE		
10441	033206				SELECT TEMPO OF 5		
10442	033206	013746	000476			MOV	TEMPO,-(SP)
10443	033212	002544				BLT	\$51317
10444	033214	006316				ASL	(SP)
10445	033216	062716	033224			ADD	#51310,(SP)
10446	033222	013607				MOV	@(SP)+,PC
10447	033224						\$51310:
10448	033224	033524				.WORD	\$51316
10449	033226	033240				.WORD	\$51315
10450	033230	033300				.WORD	\$51314
10451	033232	033344				.WORD	\$51313
10452	033234	033410				.WORD	\$51312

10509								
10510	033406				CASE 4			
10511	033406	000446						BR \$51317
10512	033410							\$51312:
10513	033410				LET DESCRO := DATA00 SET.BY #40000		;TRAILING SEPARATE	
10514	033410	013737	000562	000616				MOV DATA00,DESCRO
10515	033416	052737	040000	000616				BIS #40000,DESCRO
10516	033424				LET R5 := TEMP2 + #1			
10517	033424	013705	000502					MOV TEMP2,R5
10518	033430	005205						INC R5
10519	033432				IF SIGN0 EQ #0 THEN			
10520	033432	005737	000450					TST SIGN0
10521	033436	001003						BNE \$51324
10522	033440				LET (R5) :B- #53		;+ SIGN	
10523	033440	112715	000053		ELSE			MOVB #53,(R5)
10524	033444							
10525	033444	000402						BR \$51325
10526	033446							\$51324:
10527	033446				LET (R5) :B- #55		;- SIGN	
10528	033446	112715	000055		ENDIF			MOVB #55,(R5)
10529	033452							
10530	033452							\$51325:
10531								
10532	033452				CASE 5			
10533	033452	000424						BR \$51317
10534	033454							\$51311:
10535	033454				LET DESCRO := DATA00 SET.BY #50000		;LEADING SEPARATE	
10536	033454	013737	000562	000616				MOV DATA00,DESCRO
10537	033462	052737	050000	000616				BIS #50000,DESCRO
10538	033470				LET R5 := TEMP2 + #1			
10539	033470	013705	000502					MOV TEMP2,R5
10540	033474	005205						INC R5
10541	033476				LET (R5) :B= #0		;RESTORE TRAILING SEPARATE	
10542	033476	105015						CLRB (R5)
10543	033500				IF SIGN0 EQ #0 THEN			
10544	033500	005737	000450					TST SIGN0
10545	033504	001004						BNE \$51326
10546	033506				LET DSBUF1-1 :B= #53			
10547	033506	112737	000053	064273	ELSE			MOVB #53,DSBUF1-1
10548	033514							
10549	033514	000403						BR \$51327
10550	033516							\$51326:
10551	033516				LET DSBUF1-1 :B= #55			
10552	033516	112737	000055	064273	ENDIF			MOVB #55,DSBUF1-1
10553	033524				ENDSELECT			
10554	033524							\$51327:
10555	033524							
10556	033524							\$51316:
10557	033524							\$51317:
10558								
10559					;INCR TEMP1 FROM #1 TO #5 BY #1		SRC2 DATA TYPES	
10560	033524				INLINE <CLR @TEMP1>			
10561	033524	005037	000500					CLR @TEMP1
10562	033530				INLINE <INC2: INC TEMP1>			
10563	033530	005237	000500					INC2: INL TEMP1
10564	033534				IF TEMP1 GT #5 THEN			

10565	033534	023727	000500	000005					
10566	033542	003402							
10567	033544								
10568	033544	000137	035016		INLINE <JMP END2>				
10569	033550				ENDIF				
10570	033550								\$51330:
10571									
10572									
10573					:+				
10574					:-	RESTORE MSD & LSD OF DSBUF2			
10575									
10576	033550				LET DSBUF2 :B= DSTMP2				
10577	033550	113737	000556	064336					MOVB DSTMP2,DSBUF2
10578	033556				LET TEMP3 := #DSBUF2-1 + DATA02				
10579	033556	012737	064335	000504					MOV #DSBUF2-1,TEMP3
10580	033564	063737	000566	000504					ADD DATA02,TEMP3
10581	033572				LET @TEMP3 :B= DSTMP3				
10582	033572	113777	000557	144704					MOVB DSTMP3,@TEMP3
10583									
10584									
10585					:+				
10586					:-	SIGN DSBUF2 BY DATA TYPE			
10587									
10588	033600				SELECT TEMP1 OF 5				
10589	033600	013746	000500						MOV TEMP1,-(SP)
10590	033604	002544							BLT \$51340
10591	033606	006316							(SP)
10592	033610	062716	033616						ADD #51331,(SP)
10593	033614	013607							MOV @((SP)+,PC
10594	033616								\$51331:
10595	033616	034116							.WORD \$51337
10596	033620	033632							.WORD \$51336
10597	033622	033672							.WORD \$51335
10598	033624	033736							.WORD \$51334
10599	033626	034002							.WORD \$51333
10600	033630	034046							.WORD \$51332
10601	033632				CASE 1				
10602	033632								\$51336:
10603	033632				LET DESCR2 := DATA02				
10604	033632	013737	000566	000622					MOV DATA02,DESCR2
10605	033640				IF SIGN1 EQ #0 THEN				
10606	033640	005737	000452						TST SIGN1
10607	033644	001004							BNE \$51341
10608	033646				LET @TEMP3 :B= @TEMP3 SET.BY #60 ;+				
10609	033646	152777	000060	144630					BISB #60,@TEMP3
10610	033654				ELSE				
10611	033654	000403							BR \$51342
10612	033656								\$51341:
10613	033656				LET @TEMP3 :B= @TEMP3 SET.BY #160;-				
10614	033656	152777	000160	144620					BISB #160,@TEMP3
10615	033664				ENDIF				
10616	033664								\$51342:
10617	033664				LET DSBUF2-1 :B= #0				
10618	033664	105037	064335						CLRB DSBUF2-1
10619									
10620	033670				CASE 2				

10621	033670	000512						BR	\$51340
10622	033672							\$51335:	
10623	033672					LET DESCR2 := DATA02 SET.BY #20000 ;TRAILING OVERPUNCH		MOV	DATA02,DESCR2
10624	033672	013737	000566	000622				BIS	#20000,DESCR2
10625	033700	052737	020000	000622					
10626	033706					LET R5 :B= DSTMP3		MOVB	DSTMP3,R5
10627	033706	113705	000557						
10628	033712					IF SIGN1 EQ #-1 THEN		CMP	SIGN1,#-1
10629	033712	023727	000452	177777				BNE	\$51343
10630	033720	001002							
10631	033722					LET R5 := R5 + #10.		ADD	#10.,R5
10632	033722	062705	000012						
10633	033726					ENDIF		\$51343:	
10634	033726								
10635	033726					LET @TEMP3 :B= OVPNCH(R5)		MOVB	OVPNCH(R5),@TEMP3
10636	033726	116577	056530	144550					
10637									
10638	033734					CASE 3		BR	\$51340
10639	033734	000470						\$51334:	
10640	033736								
10641	033736					LET DESCR2 := DATA02 SET.BY #30000 ;LEADING OVERPUNCH		MOV	DATA02,DESCR2
10642	033736	013737	000566	000622				BIS	#30000,DESCR2
10643	033744	052737	030000	000622					
10644	033752					LET R5 :B= DSTMP2		MOVB	DSTMP2,R5
10645	033752	113705	000556						
10646	033756					IF SIGN1 EQ #-1 THEN		CMP	SIGN1,#-1
10647	033756	023727	000452	177777				BNE	\$51344
10648	033764	001002							
10649	033766					LET R5 := R5 + #10.		ADD	#10.,R5
10650	033766	062705	000012						
10651	033772					ENDIF		\$51344:	
10652	033772								
10653	033772					LET DSBUF2 :B= OVPNCH(R5)		MOVB	OVPNCH(R5),DSBUF2
10654	033772	116537	056530	064336					
10655									
10656	034000					CASE 4		BR	\$51340
10657	034000	000446						\$51333:	
10658	034002								
10659	034002					LET DESCR2 := DATA02 SET.BY #40000 ;TRAILING SEPARATE		MOV	DATA02,DESCR2
10660	034002	013737	000566	000622				BIS	#40000,DESCR2
10661	034010	052737	040000	000622					
10662	034016					LET R5 := TEMP3 + #1		MOV	TEMP3,R5
10663	034016	013705	000504					INC	R5
10664	034022	005205							
10665	034024					IF SIGN1 EQ #0 THEN		TST	SIGN1
10666	034024	005737	000452					BNE	\$51345
10667	034030	001003							
10668	034032					LET (R5) :B= #53		MOVB	#53,(R5)
10669	034032	112715	000053						
10670	034036					ELSE			
10671	034036	000402						BR	\$51346
10672	034040							\$51345:	
10673	034040					LET (R5) :B= #55			
10674	034040	112715	000055					MOVB	#55,(R5)
10675	034044					ENDIF			
10676	034044							\$51346:	

10677					CASE 5		
10678	034044					BR	\$51340
10679	034044	000424				\$51332:	
10680	034046				LET DESCR2 := DATA02 SET.BY #50000 ;LEADING SEPARATE	MOV	DATA02,DESCR2
10681	034046					BIS	#50000,DESCR2
10682	034046	013737	000566	000622		MOV	TEMP3,R5
10683	034054	052737	050000	000622	LET R5 := TEMP3 + #1	INC	R5
10684	034062					CLRB	(R5)
10685	034062	013705	000504		LET (R5) :B= #0	TST	SIGN1
10686	034066	005205			IF SIGN1 EQ #0 THEN	BNE	\$51347
10687	034070					MOVB	#53,DSBUF2-1
10688	034070	105015			LET DSBUF2-1 :B= #53	BR	\$51350
10689	034072				ELSE	\$51347:	
10690	034072	005737	000452		LET DSBUF2-1 :B= #55	MOVB	#55,DSBUF2-1
10691	034076	001004			ENDIF	\$51350:	
10692	034100				ENDSELECT	\$51337:	
10693	034100	112737	000053	064335		\$51340:	
10694	034106						
10695	034106	000403			;INCR DATTYP FROM #1 TO #5 BY #1 ;DS.EXP LOOP		
10696	034110				INLINE <CLR @DATTYP>	CLR @DATTYP	
10697	034110				INLINE <INC3: INC DATTYP>	INC3: INC DATTYP	
10698	034110	112737	000055	064335	IF DATTYP GT #5 THEN	CMP	DATTYP,#5
10699	034116					BLE	\$51351
10700	034116				INLINE <JMP END3>	JMP END3	
10701	034116				ENDIF	\$51351:	
10702	034116						
10703	034116						
10704							
10705							
10706	034116						
10707	034116	005037	000464				
10708	034122						
10709	034122	005237	000464				
10710	034126						
10711	034126	023727	000464	000005			
10712	034134	003402					
10713	034136						
10714	034136	000137	035012				
10715	034142						
10716	034142						
10717							
10718							
10719							
10720							
10721							
10722	034142						
10723	034142	113737	000560	064442	LET DS.EXP :B= DSTMP4 ;MSD	MOVB	DSTMP4,DS.EXP
10724	034150				LET TEMP4 := #DS.EXP-1 + DATA04	MOV	#DS.EXP-1,TEMP4
10725	034150	012737	064441	000506		ADD	DATA04,TEMP4
10726	034156	063737	000572	000506		MOVB	DSTMP5,@TEMP4
10727	034164				LET @TEMP4 :B= DSTMP5 ;LSD		
10728	034164	113777	000561	144314			
10729							
10730							
10731							
10732							

10733							
10734	034172				SELECT DATTYP OF 5		
10735	034172	013746	000464			MOV	DATTYP,-(SP)
10736	034176	002561				BLT	\$51361
10737	034200	006316				ASL	(SP)
10738	034202	062716	034210			ADD	#\$51352,(SP)
10739	034206	013607				MOV	@(SP)+,PC
10740	034210					\$51352:	
10741	034210	034542				.WORD	\$51360
10742	034212	034224				.WORD	\$51357
10743	034214	034272				.WORD	\$51356
10744	034216	034350				.WORD	\$51355
10745	034220	034426				.WORD	\$51354
10746	034222	034472				.WORD	\$51353
10747	034224				CASE 1		
10748	034224					\$51357:	
10749	034224				LET DESCR4 := DATA04		
10750	034224	013737	000572	000626		:SIGNED ZONED	
10751	034232				IF DATA04 NE #0 THEN	MOV	DATA04,DESCR4
10752	034232	005737	000572			TST	DATA04
10753	034236	001414				BEQ	\$51362
10754	034240				IF SIGN2 EQ #0 THEN		
10755	034240	005737	000454			TST	SIGN2
10756	034244	001004				BNE	\$51363
10757	034246				LET @TEMP4 :B= @TEMP4 SET.BY #60		
10758	034246	152777	000060	144232	ELSE	BISB	#60,@TEMP4
10759	034254						
10760	034254	000403				BR	\$51364
10761	034256					\$51363:	
10762	034256				LET @TEMP4 :B= @TEMP4 SET.BY #160		
10763	034256	152777	000160	144222	ENDIF	BISB	#160,@TEMP4
10764	034264						
10765	034264				LET DS.EXP-1 :B= #0	\$51364:	
10766	034264					CLRB	DS.EXP-1
10767	034264	105037	064441		ENDIF	\$51362:	
10768	034270						
10769	034270						
10770							
10771	034270				CASE 2		
10772	034270	000524				BR	\$51361
10773	034272					\$51356:	
10774	034272				LET DESCR4 := DATA04 SET.BY #20000 ;TRAILING OVERPUNCH		
10775	034272	013737	000572	000626		MOV	DATA04,DESCR4
10776	034300	052737	020000	000626		BIS	#20000,DESCR4
10777	034306				IF DATA04 NE #0 THEN		
10778	034306	005737	000572			TST	DATA04
10779	034312	001415				BEQ	\$51365
10780	034314				LET R5 :B= DSTMP5		
10781	034314	113705	000561			MOVB	DSTMP5,R5
10782	034320				LET R5 := R5 - #60		
10783	034320	162705	000060			SUB	#60,R5
10784	034324				IF SIGN2 EQ #-1 THEN		
10785	034324	023727	000454	177777		CMP	SIGN2,-1
10786	034332	001002				BNE	\$51366
10787	034334				LET R5 :- R5 + #10.		
10788	034334	062705	000012			ADD	#10.,R5

10789	034340				ENDIF				
10790	034340							\$51366:	
10791	034340				LET @TEMP4 :B= OVPNCH(R5)				
10792	034340	116577	056530	144140				MOVB	OVPNCH(R5),@TEMP4
10793	034346				ENDIF				
10794	034346							\$51365:	
10795									
10796	034346				CASE 3				
10797	034346	000475						BR	\$51361
10798	034350							\$51355:	
10799	034350				LET DESCR4 := DATA04 SET.BY #30000 ;LEADING OVERPUNCH			MOV	DATA04,DESCR4
10800	034350	013737	000572	000626				BIS	#30000,DESCR4
10801	034356	052737	030000	000626					
10802	034364				IF DATA04 NE #0 THEN			TST	DATA04
10803	034364	005737	000572					BEQ	\$51367
10804	034370	001415							
10805	034372				LET R5 :B= DSTMP4			MOVB	DSTMP4,R5
10806	034372	113705	000560						
10807	034376				LET R5 := R5 - #60			SUB	#60,R5
10808	034376	162705	000060						
10809	034402				IF SIGN? EQ #-1 THEN			CMP	SIGN2,#-1
10810	034402	023727	000454	177777				BNE	\$51370
10811	034410	001002							
10812	034412				LET R5 := R5 + #10.			ADD	#10.,R5
10813	034412	062705	000012		ENDIF				\$51370:
10814	034416								
10815	034416				LET DS.EXP :B= OVPNCH(R5)			MOVB	OVPNCH(R5),DS.EXP
10816	034416	116537	056530	064442					
10817	034416				ENDIF				\$51367:
10818	034424								
10819	034424								
10820									
10821	034424				CASE 4				
10822	034424	000446						BR	\$51361
10823	034426							\$51354:	
10824	034426				LET DESCR4 := DATA04 SET.BY #40000 ;TRAILING SEPARATE			MOV	DATA04,DESCR4
10825	034426	013737	000572	000626				BIS	#40000,DESCR4
10826	034434	052737	040000	000626					
10827	034442				LET R5 := TEMP4 + #1			MOV	TEMP4,R5
10828	034442	013705	000506					INC	R5
10829	034446	005205							
10830	034450				IF SIGN2 EQ #0 THEN			TST	SIGN2
10831	034450	005737	000454					BNE	\$51371
10832	034454	001003							
10833	034456				LET (R5) :B= #53			MOVB	#53,(R5)
10834	034456	112715	000053						
10835	034462				ELSE				
10836	034462	000402						BR	\$51372
10837	034464							\$51371:	
10838	034464				LET (R5) :B= #55				
10839	034464	112715	000055					MOVB	#55,(R5)
10840	034470				ENDIF				\$51372:
10841	034470								
10842									
10843	034470				CASE 5				
10844	034470	000424						BR	\$51361

10845 034472
 10846 034472
 10847 034472 013737 000572 000626
 10848 034500 052737 050000 000626
 10849 034506
 10850 034506 015705 000506
 10851 034512 005205
 10852 034514
 10853 034514 105015
 10854 034516
 10855 034516 005737 000454
 10856 034522 001004
 10857 034524
 10858 034524 112737 000053 064441
 10859 034532
 10860 034532 000403
 10861 034534
 10862 034534
 10863 034534 112737 000055 064441
 10864 034542
 10865 034542
 10866 034542
 10867 034542
 10868 034542
 10869
 10870
 10871
 10872
 10873
 10874 034542
 10875 034542 012700 064377
 10876 034546
 10877 034546 012701 000041
 10878 034552
 10879 034552 105020
 10880 034554
 10881 034554 077102
 10882 034556
 10883 034556 013700 000616
 10884 034562
 10885 034562 013701 000620
 10886 034566
 10887 034566 013702 000622
 10888 034572
 10889 034572 013703 000624
 10890 034576
 10891 034576 013704 000626
 10892 034602
 10893 034602 013705 000630
 10894 034606
 10895 034606 010437 000534
 10896 034612
 10897 034612 010537 000536
 10898 034616
 10899 034616 076050
 10900

LET DESCR4 := DATA04 SET.BY #50000

 LET R5 := TEMP4 + #1

 LET (R5) :B= #0
 IF SIGN2 EQ #0 THEN

 LET DS.EXP-1 :B= #53
 ELSE

 LET DS.EXP-1 :B= #55
 ENDIF
 ENDSELECT

:+
 :
 :-
 CLR DS.DST & LOAD DESCRIPTORS

LET R0 := #DS.DST-1
 LET R1 := #41
 INLINE <DOIT1: CLR (R0)+>
 INLINE <SOB R1, DOIT1>
 LET R0 := DESCR0
 LET R1 := DESCR1
 LET R2 := DESCR2
 LET R3 := DESCR3
 LET R4 := DESCR4
 LET R5 := DESCR5
 LET R4.EXP := R4
 LET R5.EXP := R5
 INLINE <ADDN>

\$51353:
 MOV DATA04,DESCR4
 BIS #50000,DESCR4

 MOV TEMP4,R5
 INC R5
 CLRB (R5)

 TST SIGN2
 BNE \$51373

 MOVB #53,DS.EXP-1

 BR \$51374
 \$51373:
 MOVB #55,DS.EXP-1

 \$51374:

 \$51360:
 \$51361:

 MOV #DS.DST-1,R0
 MOV #41,R1
 DOIT1: CLR (R0)+
 SOB R1, DOIT1
 MOV DESCR0,R0
 MOV DESCR1,R1
 MOV DESCR2,R2
 MOV DESCR3,R3
 MOV DESCR4,R4
 MOV DESCR5,R5
 MOV R4,R4.EXP
 MOV R5,R5.EXP
 ADDN

```

10901
10902          ;+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.
10903          ;-
10904
10905 034620          LET CC := PSW
10906 034620 013737 177776 000520          MOV      PSW,CC
10907 034626          LET CC := CC CLR.BY #177760
10908 034626 042737 177760 000520          BIC      #177760,CC
10909 034634          IF CC NE CC.EXP THEN
10910 034634 023737 000520 000522          CMP      CC,CC.EXP
10911 034642 001402          BEQ      $51375
10912 034644          CALL ERROR
10913 034644 004737 060044          JSR      PC,ERROR
10914 034650          ENDIF
10915 034650          $51375:
10916 034650          IF R0 NE R0.EXP THEN
10917 034650 020037 000524          CMP      R0,R0.EXP
10918 034654 001402          BEQ      $51376
10919 034656          CALL ERROR
10920 034656 004737 060044          JSR      PC,ERROR
10921 034662          ENDIF
10922 034662          $51376:
10923 034662          IF R1 NE R1.EXP THEN
10924 034662 020137 000526          CMP      R1,R1.EXP
10925 034666 001402          BEQ      $51377
10926 034670          CALL ERROR
10927 034670 004737 060044          JSR      PC,ERROR
10928 034674          ENDIF
10929 034674          $51377:
10930 034674          IF R2 NE R2.EXP THEN
10931 034674 020237 000530          CMP      R2,R2.EXP
10932 034700 001402          BEQ      $51400
10933 034702          CALL ERROR
10934 034702 004737 060044          JSR      PC,ERROR
10935 034706          ENDIF
10936 034706          $51400:
10937 034706          IF R3 NE R3.EXP THEN
10938 034706 020337 000532          CMP      R3,R3.EXP
10939 034712 001402          BEQ      $51401
10940 034714          CALL ERROR
10941 034714 004737 060044          JSR      PC,ERROR
10942 034720          ENDIF
10943 034720          $51401:
10944 034720          IF R4 NE R4.EXP THEN
10945 034720 020437 000534          CMP      R4,R4.EXP
10946 034724 001402          BEQ      $51402
10947 034726          CALL ERROR
10948 034726 004737 060044          JSR      PC,ERROR
10949 034732          ENDIF
10950 034732          $51402:
10951 034732          IF R5 NE R5.EXP THEN
10952 034732 020537 000536          CMP      R5,R5.EXP
10953 034736 001402          BEQ      $51403
10954 034740          CALL ERROR
10955 034740 004737 060044          JSR      PC,ERROR
10956 034744          ENDIF
    
```



```

11013      :+
11014      :+      INLINE CASE - ADDNI, USING UNSIGNED ZONED DATA
11015      :-
11016      :+
11017      035046      CALL CLRDSB
11018      035046      004737      057040      JSR      PC,CLRDSB
11019      :+
11020      :+      LOAD DSBUF1
11021      :-
11022      :+
11023      :+
11024      035052      LET R0 := #ADD1+2      ;SOURCE      MOV      #ADD1+2,R0
11025      035052      012700      065264      ;# OF BYTES TO XFER
11026      035056      LET R1 := TADDI CLR.BY #10000      MOV      TADDI,R1
11027      035056      013701      065756      MOV      #10000,R1
11028      035062      042701      010000      BIC
11029      035066      LET R2 := #DSBUF1      ;DESTINATION      MOV      #DSBUF1,R2
11030      035066      012702      064274      :+
11031      035072      INLINE <1$:      MOVB      (R0)+, (R2)+      1$:      MOVB      (R0)+, (R2)+
11032      035072      112022      :+
11033      035074      INLINE <      SOB      R1,      1$>      SOB      R1,      1$
11034      035074      077102      :+
11035      :+
11036      :+      LOAD DSBUF2
11037      :-
11038      :+
11039      :+
11040      035076      LET R0 := #ADD1A+2      MOV      #ADD1A+2,R0
11041      035076      012700      065306      LET R1 := TADDI+4 CLR.BY #10000      MOV      TADDI+4,R1
11042      035102      013701      065762      MOV      #10000,R1
11043      035102      042701      010000      BIC
11044      035106      LET R2 := #DSBUF2      MOV      #DSBUF2,R2
11045      035112      012702      064336      :+
11046      035112      INLINE <2$:      MOVB      (R0)+, (R2)+      2$:      MOVB      (R0)+, (R2)+
11047      035116      112022      :+
11048      035116      INLINE <      SOB      R1,      2$>      SOB      R1,      2$
11049      035120      077102      :+
11050      035120      :+
11051      :+      LOAD DS.EXP
11052      :-
11053      :+
11054      :+
11055      :+
11056      035122      LET R0 := #ADD1B+2      MOV      #ADD1B+2,R0
11057      035122      012700      065332      LET R1 := TADDI+10 CLR.BY #10000      MOV      TADDI+10,R1
11058      035126      013701      065766      MOV      #10000,R1
11059      035126      042701      010000      BIC
11060      035132      LET R2 := #DS.EXP      MOV      #DS.EXP,R2
11061      035136      012702      064442      :+
11062      035136      INLINE <3$:      MOVB      (R0)+, (R2)+      3$:      MOVB      (R0)+, (R2)+
11063      035142      112012      :+
11064      035142      INLINE <      BISB      #60, (R2)+      BISB      #60, (R2)+
11065      035144      152722      000060      :+
11066      035144      INLINE <      SOB      R1,      3$>      SOB      R1,      3$
11067      035150      077104      :+
11068      035150
    
```

```

11069
11070 035152          CALL CLRREG          ;ENSURE R0-R5 ARE CLR
11071 035152 004737 056604      JSR      PC,CLRREG
11072
11073 035156          INLINE <ADDNI>
11074 035156 076150          ADDNI
11075 035160          INLINE <.WORD TADDI>
11076 035160 065756          .WORD TADDI
11077 035162          INLINE <.WORD TADDI+4>
11078 035162 065762          .WORD TADDI+4
11079 035164          INLINE <.WORD TADDI+10>
11080 035164 065766          .WORD TADDI+10
11081 035166          LET CC := PSW
11082 035166 013737 177776 000520  MOV     PSW,CC
11083 035174          LET CC := CC CLR.BY #177760
11084 035174 042737 177760 000520  BIC     #177760,CC
11085 035202          IF CC NE #0 THEN
11086 035202 005737 000520          TST     CC
11087 035206 001402          BEQ     $51410
11088 035210          CALL ERROR
11089 035210 004737 060044      JSR     PC,ERROR
11090 035214          ENDIF
11091 035214
11092
11093          :+
11094          :      ENSURE GPR'S STILL ZERO
11095          :-
11096
11097 035214          IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
11098 035214 005700          TST     R0
11099 035216 001004          BNE     $51411
11100 035220 005701          TST     R1
11101 035222 001002          BNE     $51411
11102 035224 005702          TST     R2
11103 035226 001402          BEQ     $51412
11104 035230          $51411:
11105 035230          CALL ERROR
11106 035230 004737 060044      JSR     PC,ERROR
11107 035234          ENDIF
11108 035234
11109 035234          IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
11110 035234 005703          TST     R3
11111 035236 001004          BNE     $51413
11112 035240 005704          TST     R4
11113 035242 001002          BNE     $51413
11114 035244 005705          TST     R5
11115 035246 001402          BEQ     $51414
11116 035250          $51413:
11117 035250          CALL ERROR
11118 035250 004737 060044      JSR     PC,ERROR
11119 035254          ENDIF
11120 035254
11121
11122          :+
11123          :      CHECK RESULTS
11124          :-
    
```

11125							
11126	035254			LET R0 := #DS.DST			
11127	035254	012700	064400			MOV	#DS.DST,R0
11128	035260			LET R1 := #DS.EXP	;EXPECTED DATA		
11129	035260	012701	064442			MOV	#DS.EXP,R1
11130	035264			LET ERRFLG := #0			
11131	035264	005037	000456			CLR	ERRFLG
11132	035270			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO			
11133	035270					\$51415:	
11134	035270	020027	064440			CMP	R0,#DS.DST+40
11135	035274	001410				BEQ	\$51416
11136	035276	005737	000456			TST	ERRFLG
11137	035302	001005				BNE	\$51416
11138	035304			IFB (R0)+ NE (R1)+ THEN			
11139	035304	122021				(CMPB	(R0)+,(R1)+
11140	035306	001402				BEQ	\$51417
11141	035310			CALL ERROR			
11142	035310	004737	060044			JSR	PC,ERROR
11143	035314			ENDIF			
11144	035314					\$51417:	
11145	035314			ENDDO			
11146	035314	000765				BR	\$51415
11147	035316					\$51416:	
11148							
11149							

```

11150 .SBTTL SUBN
11151 :*****
11152 :
11153 :       SUBN TEST
11154 :
11155 :*****
11156
11157 035316  INLINE <XSUBN:>                                XSUBN:
11158 035316
11159 035316  INLINE <SCOPE>                                SCOPE
11160 035316 000004  LET $TIMES := #10                                ;ITERATION COUNT
11161 035320 012737 000010 000404  MOV #10,$TIMES
11162 035320 013737 065104 000470  LET COUNT := TADD                                ;# OF TEST CASES
11163 035326 013737 065104 000470  MOV TADD,COUNT
11164 035326 012737 065106 000544  LET TSTCAS :- #TADD+2                                ;POINT TO FIRST TEST CASE
11165 035334 012737 065106 000544  MOV #TADD+2,TSTCAS
11166 035334
11167
11168 :+
11169 :       CLR R0.EXP TO R3.EXP
11170 :-
11171
11172 035342  INCR R0 FROM #0 TO #6 BY #2
11173 035342 005000  CLR R0
11174 035344 000402  BR $51420
11175 035346 062700 000002  $51421:
11176 035346 020027 000006  ADD #2,R0
11177 035352 003003  $51420:
11178 035356 000770  CMP R0,#6
11179 035356 000770  BGT $51422
11180 035360 005060 000524  CLR R0.EXP(R0)
11181 035360 000770  ENDINC
11182 035364 000770
11183 035364
11184 035366
11185 ;REPEAT OUTER LOOP FOR TEST CASES
11186 035366  INCR R0 FROM #0 TO #6 BY #2
11187 035366
11188 035366  CALL CLRDSB                                ;CLR DS BUFFERS
11189 035366 004737 057040  JSR PC,CLRDSB
11190
11191 :+
11192 :       LOAD DATAXX AND CC.EXP FROM TEST CASE
11193 :-
    
```

```

11194
11195 035372          LET R0 := TSTCAS
11196 035372 013700 000544          ;SRC1 LENGTH          MOV    TSTCAS,R0
11197 035376          LET DATA00 := (R0)+          ;POINTER TO SRC1 DATA  MOV    (R0)+,DATA00
11198 035376 012037 000562          ;SRC2 LENGTH          MOV    (R0)+,DATA01
11199 035402          LET DATA01 := (R0)+          ;POINTER TO SRC2 DATA  MOV    (R0)+,DATA02
11200 035402 012037 000564          ;DEST LENGTH          MOV    (R0)+,DATA03
11201 035406          LET DATA02 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)  MOV    (R0)+,DATA04
11202 035406 012037 000566          ;CC.EXP (SUM)          MOV    (R0)+,DATA05
11203 035412          LET DATA03 := (R0)+          ;POINTER TO DS.EXP DATA (DIFF)  MOV    (R0)+,DATA05
11204 035412 012037 000570          ;CC.EXP (DIFF)          MOV    (R0)+,CC.EXP
11205 035416          LET DATA04 := (R0)+
11206 035416 012037 000572          ;CHECK FOR ZERO LENGTH BUFFER
11207 035422          LET DATA05 := (R0)+          TST    DATA00
11208 035422 012037 000574          ;CC.EXP (SUM)          BEQ    $51423
11209 035426          LET CC.EXP := (R0)+
11210 035426 012037 000522          ;STORE SIGN AND POINT AT FIRST BYTE  MOV    DATA01,R0
11211 035432          LET DATA05 := (R0)+          ;# OF BYTES TO TRANSFER  MOV    (R0)+,SIGNO
11212 035432 012037 000574          ;TRANSFER DESTINATION  MOV    DATA00,R1
11213 035436          LET CC.EXP := (R0)+          MOV    #DSBUF1,R2
11214 035436 012037 000522          ;CHECK FOR ZERO LENGTH BUFFER
11215
11216          ;+
11217          ;   LOAD DSBUF1
11218          ;-
11219
11220          IF DATA00 NE #0 THEN          ;CHECK FOR ZERO LENGTH BUFFER
11221 035442 005737 000562          TST    DATA00
11222 035446 001413          BEQ    $51423
11223 035450          LET R0 := DATA01          MOV    DATA01,R0
11224 035450 013700 000564          ;STORE SIGN AND POINT AT FIRST BYTE  MOV    (R0)+,SIGNO
11225 035454          LET SIGNO := (R0)+          ;# OF BYTES TO TRANSFER  MOV    DATA00,R1
11226 035454 012037 000450          ;TRANSFER DESTINATION  MOV    #DSBUF1,R2
11227 035460          LET R1 := DATA00
11228 035460 013701 000562          ;CHECK FOR ZERO LENGTH BUFFER
11229 035464          LET R2 := #DSBUF1          TST    DATA00
11230 035464 012702 064274          BEQ    $51423
11231 035470          INLINE <1$:>          1$:
11232 035470          INLINE <MOVB (R0)+,(R2)+>          MOVB (R0)+,(R2)+
11233 035470          INLINE <SOB R1,1$>          SOB R1,1$
11234 035470 112022          ;CHECK FOR ZERO LENGTH BUFFER
11235 035472          ELSE          BR    $51424
11236 035472 077102          ;STORE SIGN AND POINT AT FIRST BYTE  $51423:
11237 035474          LET SIGNO := #0          CLR    SIGNO
11238 035474 000402          ;TRANSFER DESTINATION  $51424:
11239 035476          ENDIF
11240 035476          LET SIGNO := #0
11241 035476 005037 000450          ;CHECK FOR ZERO LENGTH BUFFER
11242 035502          ;STORE SIGN AND POINT AT FIRST BYTE
11243 035502          ;# OF BYTES TO TRANSFER
11244          ;TRANSFER DESTINATION
11245          ;+
11246          ;   LOAD DSBUF2
11247          ;-
11248
11249 035502          IF DATA02 NE #0 THEN

```

11250	035502	005737	000566		TST	DATA02
11251	035506	001413			BEQ	\$51425
11252	035510			LET R0 := DATA03	MOV	DATA03,R0
11253	035510	013700	000570	LET SIGN1 := (R0)+	MOV	(R0)+,SIGN1
11254	035514			LET R1 := DATA02	MOV	DATA02,R1
11255	035514	012037	000452	LET R2 := #DSBUF2	MOV	#DSBUF2,R2
11256	035520			INLINE <2\$:>	2\$:	
11257	035520	013701	000566	INLINE <MOVB (R0)+,(R2)+>	MOVB (R0)+,(R2)+	
11258	035524			INLINE <SOB R1,2\$>	SOB R1,2\$	
11259	035524	012702	064336	ELSE	BR	\$51426
11260	035530				\$51425:	
11261	035530			LET SIGN1 := #0	CLR	SIGN1
11262	035530			ENDIF	\$51426:	
11263	035530	112022				
11264	035532			:+		
11265	035532	077102		LOAD DS.EXP		
11266	035534			:-		
11267	035534	000402		IF DATA04 NE #0 THEN	TST	DATA04
11268	035536				BEQ	\$51427
11269	035536			LET R0 := DATA05	MOV	DATA05,R0
11270	035536	005037	000452	LET SIGN2 := (R0)+	MOV	(R0)+,SIGN2
11271	035542			LET R1 := DATA04	MOV	DATA04,R1
11272	035542			LET R2 := #DS.EXP	MOV	#DS.EXP,R2
11273				INLINE <3\$:>	3\$:	
11274				INLINE <MOVB (R0)+,(R2)+>	MOVB (R0)+,(R2)+	
11275				INLINE <BISB #60,(R2)+>	BISB #60,(R2)+	
11276				INLINE <SOB R1,3\$>	SOB R1,3\$	
11277				ELSE	BR	\$51430
11278	035542			LET SIGN2 := #0	\$51427:	
11279	035542	005737	000572	ENDIF	CLR	SIGN2
11280	035546	001415			\$51430:	
11281	035550			:+		
11282	035550	013700	000574			
11283	035554					
11284	035554	012037	000454			
11285	035560					
11286	035560	013701	000572			
11287	035564					
11288	035564	012702	064442			
11289	035570					
11290	035570					
11291	035570					
11292	035570	112012				
11293	035572					
11294	035572	152722	000060			
11295	035576					
11296	035576	077104				
11297	035600					
11298	035600	000402				
11299	035602					
11300	035602					
11301	035602	005037	000454			
11302	035606					
11303	035606					
11304						
11305				:+		

```

11306          :      LOAD DSTMP'S WITH MSD & LSD OF EACH BUFFER
11307          :-
11308
11309 035606          LET DSTMP0 :B= DSBUF1
11310 035606 113737 064274 000554          MOV      DSBUF1,DSTMP0
11311 035614          LET TEMPO := #DSBUF1-1 + DATA00
11312 035614 012737 064273 000476          MOV      #DSBUF1-1,TEMPO
11313 035622 063737 000562 000476          ADD      DATA00,TEMPO
11314 035630          LET DSTMP1 :B= @TEMPO
11315 035630 117737 142642 000555          MOV      @TEMPO,DSTMP1
11316 035636          LET DSTMP2 :B= DSBUF2
11317 035636 113737 064336 000556          MOV      DSBUF2,DSTMP2
11318 035644          LET TEMPO := #DSBUF2-1 + DATA02
11319 035644 012737 064335 000476          MOV      #DSBUF2-1,TEMPO
11320 035652 063737 000566 000476          ADD      DATA02,TEMPO
11321 035660          LET DSTMP3 :B= @TEMPO
11322 035660 117737 142612 000557          MOV      @TEMPO,DSTMP3
11323 035666          LET DSTMP4 :B= DS.EXP
11324 035666 113737 064442 000560          MOV      DS.EXP,DSTMP4
11325 035674          LET TEMPO := #DS.EXP-1 + DATA04
11326 035674 012737 064441 000476          MOV      #DS.EXP-1,TEMPO
11327 035702 063737 000572 000476          ADD      DATA04,TEMPO
11328 035710          LET DSTMP5 :B= @TEMPO
11329 035710 117737 142562 000561          MOV      @TEMPO,DSTMP5
11330 035716          LET DESCR1 := #DSBUF1
11331 035716 012737 064274 000620          MOV      #DSBUF1,DESCR1
11332 035724          LET DESCR3 := #DSBUF2
11333 035724 012737 064336 000624          MOV      #DSBUF2,DESCR3
11334 035732          LET DESCR5 := #DS.DST
11335 035732 012737 064400 000630          MOV      #DS.DST,DESCR5
11336
11337          ; INCR TEMPO FROM #1 TO #5 BY #1          SRC1 DATA TYPE LOOP
11338 035740          INLINE <CLR @TEMPO>
11339 035740 005037 000476          CLR      @TEMPO
11340 035744          INLINE <INC21: INC TEMPO>
11341 035744 005237 000476          INC21: INC TEMPO
11342 035750          IF TEMPO GT #5 THEN
11343 035750 023727 000476 000005          CMP      TEMPO,#5
11344 035756 003402          BLE      $51431
11345 035760          INLINE <JMP END21>
11346 035760 000137 037634          JMP      END21
11347 035764          ENDIF
11348 035764          $51431:
11349
11350          ;+
11351          ;      RESTORE MSD & LSD OF DSBUF1
11352          ;-
11353
11354 035764          LET DSBUF1 :B= DSTMP0          ;MSD
11355 035764 113737 000554 064274          MOV      DSTMP0,DSBUF1
11356 035772          LET TEMP2 := #DSBUF1-1 + DATA00
11357 035772 012737 064273 000502          MOV      #DSBUF1-1,TEMP2
11358 036000 063737 000562 000502          ADD      DATA00,TEMP2
11359 036006          LET @TEMP2 :B= DSTMP1          ;LSD
11360 036006 113777 000555 142466          MOV      DSTMP1,@TEMP2
11361

```



```

11362          :+
11363          :- SIGN DSBUFF1 BY DATA TYPE
11364          :-
11365
11366 036014          SELECT TEMPO OF 5
11367 036014 013746 000476          MOV      TEMPO,-(SP)
11368 036020 002544          BLT      $51441
11369 036022 006316          ASL      (SP)
11370 036024 062716 036032          ADD      #51432,(SP)
11371 036030 013607          MOV      @ (SP)+,PC
11372 036032          $51432:
11373 036032 036332          .WORD   $51440
11374 036034 036046          .WORD   $51437
11375 036036 036106          .WORD   $51436
11376 036040 036152          .WORD   $51435
11377 036042 036216          .WORD   $51434
11378 036044 036262          .WORD   $51433
11379 036046          $51437:
11380 036046          CASE 1
11381 036046          LET DESCRO := DATA00          ;SIGNED ZONED
11382 036046 013737 000562 000616          MOV      DATA00,DESCRO
11383 036054          IF SIGN0 EQ #0 THEN
11384 036054 005737 000450          TST      SIGN0
11385 036060 001004          BNE      $51442
11386 036062          LET @TEMP2 :B= @TEMP2 SET.BY #60          ;+
11387 036062 152777 000060 142412          BISB    #60,@TEMP2
11388 036070          ELSE
11389 036070 000403          BR      $51443
11390 036072          $51442:
11391 036072          LET @TEMP2 :B= @TEMP2 SET.BY #160          :-
11392 036072 152777 000160 142402          BISB    #160,@TEMP2
11393 036100          ENDIF
11394 036100          $51443:
11395 036100          LET DSBUFF1-1 :B= #0          ;FIX LEADING SEPARATE RESULT
11396 036100 105037 064273          CLRB   DSBUFF1-1
11397
11398 036104          CASE 2
11399 036104 000512          BR      $51441
11400 036106          $51436:
11401 036106          LET DESCRO := DATA00 SET.BY #20000          ;TRAILING OVERPUNCH
11402 036106 013737 000562 000616          MOV      DATA00,DESCRO
11403 036114 052737 020000 000616          BIS      #20000,DESCRO
11404 036122          LET R5 :B= DSTMP1          ;GET LSD
11405 036122 113705 000555          MOV      DSTMP1,R5
11406 036126          IF SIGN0 EQ #-1 THEN
11407 036126 023727 000450 177777          CMP      SIGN0,#-1
11408 036134 001002          BNE      $51444
11409 036136          LET R5 := R5 + #10.          ;CHANGE INDEX FOR -
11410 036136 062705 000012          ADD      #10.,R5
11411 036142          ENDIF
11412 036142          $51444:
11413 036142          LET @TEMP2 :B= OVPNCH(R5)
11414 036142 116577 056530 142332          MOV      OVPNCH(R5),@TEMP2
11415
11416 036150          CASE 3
11417 036150 000470          BR      $51441
    
```

11418	036152								\$51435:
11419	036152					LET DESCRO := DATA00 SET.BY #30000			;LEADING OVERPUNCH
11420	036152	013737	000562	000616					MOV DATA00,DESCRO
11421	036160	052737	030000	000616					BIS #30000,DESCRO
11422	036166					LET R5 :B= DSTMP0 ;GET MSD			MOVB DSTMP0,R5
11423	036166	113705	000554						
11424	036172					IF SIGNO EQ #-1 THEN			
11425	036172	023727	000450	177777					CMP SIGNO,#-1
11426	036200	001002							BNE \$51445
11427	036202					LET R5 := R5 + #10.			
11428	036202	062705	000012						ADD #10.,R5
11429	036206					ENDIF			
11430	036206								\$51445:
11431	036206					LET DSBUF1 :B= OVPNCH(R5)			
11432	036206	116537	056530	064274					MOVB OVPNCH(R5),DSBUF1
11433									
11434									
11435	036214					CASE 4			
11436	036214	000446							BR \$51441
11437	036216								\$51434:
11438	036216					LET DESCRO := DATA00 SET.BY #40000			;TRAILING SEPARATE
11439	036216	013737	000562	000616					MOV DATA00,DESCRO
11440	036224	052737	040000	000616					BIS #40000,DESCRO
11441	036232					LET R5 := TEMP2 + #1			
11442	036232	013705	000502						MOV TEMP2,R5
11443	036236	005205							INC R5
11444	036240					IF SIGNO EQ #0 THEN			
11445	036240	005737	000450						TST SIGNO
11446	036244	001003							BNE \$51446
11447	036246					LET (R5) :B= #53 ;+ SIGN			
11448	036246	112715	000053						MOVB #53,(R5)
11449	036252					ELSE			
11450	036252	000402							BR \$51447
11451	036254								\$51446:
11452	036254					LET (R5) :B= #55 ;- SIGN			
11453	036254	112715	000055						MOVB #55,(R5)
11454	036260					ENDIF			
11455	036260								\$51447:
11456									
11457	036260					CASE 5			
11458	036260	000424							BR \$51441
11459	036262								\$51433:
11460	036262					LET DESCRO := DATA00 SET.BY #50000			;LEADING SEPARATE
11461	036262	013737	000562	000616					MOV DATA00,DESCRO
11462	036270	052737	050000	000616					BIS #50000,DESCRO
11463	036276					LET R5 := TEMP2 + #1			
11464	036276	013705	000502						MOV TEMP2,R5
11465	036302	005205							INC R5
11466	036304					LET (R5) :B= #0 ;RESTORE TRAILING SEPARATE			
11467	036304	105015							CLRB (R5)
11468	036306					IF SIGNO EQ #0 THEN			
11469	036306	005737	000450						TST SIGNO
11470	036312	001004							BNE \$51450
11471	036314					LET DSBUF1-1 :B= #53			
11472	036314	112737	000053	064273					MOVB #53,DSBUF1-1
11473	036322					ELSE			

11474	036322	000403				BR	\$51451
11475	036324					\$51450:	
11476	036324				LET DSBUF1-1 :B= #55		
11477	036324	112737	000055	064273		MOVB	#55,DSBUF1-1
11478	036332				ENDIF		
11479	036332					\$51451:	
11480	036332				ENDSELECT		
11481	036332					\$51440:	
11482	036332					\$51441:	
11483							
11484					:INCR TEMP1 FROM #1 TO #5 BY #1	SRC2 DATA TYPES	
11485	036332				INLINE <CLR @TEMP1>		CLR @TEMP1
11486	036332	005037	000500				
11487	036336				INLINE <INC22: INC TEMP1>		INC22: INC TEMP1
11488	036336	005237	000500				
11489	036342				IF TEMP1 GT #5 THEN		
11490	036342	023727	000500	000005		CMP	TEMP1,#5
11491	036350	003402				BLE	\$51452
11492	036352				INLINE <JMP END22>		JMP END22
11493	036352	000137	037630				
11494	036356				ENDIF		\$51452:
11495	036356				LET R3 := #DSBUF2		
11496	036356					MOV	#DSBUF2,R3
11497	036356	012703	064336				
11498							
11499					:+ RESTORE MSD & LSD OF DSBUF2		
11500					: -		
11501							
11502							
11503	036362				LET DSBUF2 :B= DSTMP2		
11504	036362	113737	000556	064336		MOVB	DSTMP2,DSBUF2
11505	036370				LET TEMP3 := #DSBUF2-1 + DATA02		
11506	036370	012737	064335	000504		MOV	#DSBUF2-1,TEMP3
11507	036376	063737	000566	000504		ADD	DATA02,TEMP3
11508	036404				LET @TEMP3 :B= DSTMP3		
11509	036404	113777	000557	142072		MOVB	DSTMP3,@TEMP3
11510							
11511					:+ SIGN DSBUF2 BY DATA TYPE		
11512					: -		
11513							
11514							
11515	036412				SELECT TEMP1 OF 5		
11516	036412	013746	000500			MOV	TEMP1,-(SP)
11517	036416	002544				BLT	\$51462
11518	036420	006316				ASL	(SP)
11519	036422	062716	036430			ADD	#51453,(SP)
11520	036426	013607				MOV	@(SP)+,PC
11521	036430					\$51453:	
11522	036430	036730				.WORD	\$51461
11523	036432	036444				.WORD	\$51460
11524	036434	036504				.WORD	\$51457
11525	036436	036550				.WORD	\$51456
11526	036440	036614				.WORD	\$51455
11527	036442	036660				.WORD	\$51454
11528	036444				CASE 1		
11529	036444					\$51460:	

11530	036444				LET DESCR2 := DATA02		
11531	036444	013737	000566	000622		MOV	DATA02,DESCR2
11532	036452				IF SIGN1 EQ #0 THEN		
11533	036452	005737	000452			TST	SIGN1
11534	036456	001004				BNE	\$51463
11535	036460				LET @TEMP3 :B= @TEMP3 SET.BY #60 ;+		
11536	036460	152777	000060	142016		BISB	#60,@TEMP3
11537	036466				ELSE		
11538	036466	000403				BR	\$51464
11539	036470					\$51463:	
11540	036470				LET @TEMP3 :B= @TEMP3 SET.BY #160;-		
11541	036470	152777	000160	142006		BISB	#160,@TEMP3
11542	036476				ENDIF		
11543	036476					\$51464:	
11544	036476				LET DSBUF2-1 :B= #0		
11545	036476	105037	064335			CLRB	DSBUF2-1
11546							
11547	036502				CASE 2		
11548	036502	000512				BR	\$51462
11549	036504					\$51457:	
11550	036504				LET DESCR2 := DATA02 SET.BY #20000 ;TRAILING OVERPUNCH		
11551	036504	013737	000566	000622		MOV	DATA02,DESCR2
11552	036512	052737	020000	000622		BIS	#20000,DESCR2
11553	036520				LET R5 :B= DSTMP3		
11554	036520	113705	000557			MOVB	DSTMP3,R5
11555	036524				IF SIGN1 EQ #-1 THEN		
11556	036524	023727	000452	177777		CMP	SIGN1,#-1
11557	036532	001002				BNE	\$51465
11558	036534				LET R5 := R5 + #10.		
11559	036534	062705	000012			ADD	#10.,R5
11560	036540				ENDIF		
11561	036540					\$51465:	
11562	036540				LET @TEMP3 :B= OVPNCH(R5)		
11563	036540	116577	056530	141736		MOVB	OVPNCH(R5),@TEMP3
11564							
11565	036546				CASE 3		
11566	036546	000470				BR	\$51462
11567	036550					\$51456:	
11568	036550				LET DESCR2 := DATA02 SET.BY #30000 ;LEADING OVERPUNCH		
11569	036550	013737	000566	000622		MOV	DATA02,DESCR2
11570	036556	052737	030000	000622		BIS	#30000,DESCR2
11571	036564				LET R5 :B= DSTMP2		
11572	036564	113705	000556			MOVB	DSTMP2,R5
11573	036570				IF SIGN1 EQ #-1 THEN		
11574	036570	023727	000452	177777		CMP	SIGN1,#-1
11575	036576	001002				BNE	\$51466
11576	036600				LET R5 := R5 + #10.		
11577	036600	062705	000012			ADD	#10.,R5
11578	036604				ENDIF		
11579	036604					\$51466:	
11580	036604				LET DSBUF2 :B= OVPNCH(R5)		
11581	036604	116537	056530	064336		MOVB	OVPNCH(R5),DSBUF2
11582							
11583	036612				CASE 4		
11584	036612	000446				BR	\$51462
11585	036614					\$51455:	

11586	036614				LET DESCR2 := DATA02 SET.BY #40000 ;TRAILING SEPARATE		
11587	036614	013737	000566	000622		MOV	DATA02,DESCR2
11588	036622	052737	040000	000622		BIS	#40000,DESCR2
11589	036630				LET R5 := TEMP3 + #1		
11590	036630	013705	000504			MOV	TEMP3,R5
11591	036634	005205				INC	R5
11592	036636				IF SIGN1 EQ #0 THEN		
11593	036636	005737	000452			TST	SIGN1
11594	036642	001003				BNE	\$51467
11595	036644				LET (R5) :B= #53		
11596	036644	112715	000053			MOVB	#53,(R5)
11597	036650				ELSE		
11598	036650	000402				BR	\$51470
11599	036652					\$51467:	
11600	036652				LET (R5) :B= #55		
11601	036652	112715	000055			MOVB	#55,(R5)
11602	036656				ENDIF		
11603	036656					\$51470:	
11604							
11605	036656				CASE 5		
11606	036656	000424				BR	\$51462
11607	036660					\$51454:	
11608	036660				LET DESCR2 := DATA02 SET.BY #50000 ;LEADING SEPARATE		
11609	036660	013737	000566	000622		MOV	DATA02,DESCR2
11610	036666	052737	050000	000622		BIS	#50000,DESCR2
11611	036674				LEI R5 := TEMP3 + #1		
11612	036674	013705	000504			MOV	TEMP3,R5
11613	036700	005205				INC	R5
11614	036702				LET (R5) :B= #0		
11615	036702	105015				(LRB	(R5)
11616	036704				IF SIGN1 EQ #0 THEN		
11617	036704	005737	000452			TST	SIGN1
11618	036710	001004				BNE	\$51471
11619	036712				LET DSBUF2-1 :B= #53		
11620	036712	112737	000053	064335		MOVB	#53,DSBUF2-1
11621	036720				ELSE		
11622	036720	000403				BR	\$51472
11623	036722					\$51471:	
11624	036722				LET DSBUF2-1 :B= #55		
11625	036722	112737	000055	064335		MOVB	#55,DSBUF2-1
11626	036730				ENDIF		
11627	036730					\$51472:	
11628	036730				ENDSELECT		
11629	036730					\$51461:	
11630	036730					\$51462:	
11631							
11632					;INCR DATTYP FROM #1 TO #5 BY #1 ;DS.EXP LOOP		
11633	036730				INLINE <CLR @#DA.TYP>		
11634	036730	005037	000464			CLR @#DATTYP	
11635	036734				INLINE <INC23: INC DATTYP>		
11636	036734	005237	000464			INC23: INC DATTYP	
11637	036740				IF DATTYP GT #5 THEN		
11638	036740	023727	000464	000005		CMP	DATTYP,#5
11639	036746	003402				BLE	\$51473
11640	036750				INLINE <JMP END23>		
11641	036750	000137	037624			JMP END23	

11642	036754				ENDIF	
11643	036754					\$51473:
11644						
11645					:+	
11646					::	RESTORE MSD & LSD OF DS.EXP
11647					:-	
11648						
11649	036754				LET DS.EXP :B= DSTMP4 ;MSD	
11650	036754	113737	000560	064442		MOVB DSTMP4,DS.EXP
11651	036762				LET TEMP4 := #DS.EXP-1 + DATA04	
11652	036762	012737	064441	000506		MOV #DS.EXP-1,TEMP4
11653	036770	063737	000572	000506		ADD DATA04,TEMP4
11654	036776				LET @TEMP4 :B= DSTMP5 ;LSD	
11655	036776	113777	000561	141502		MOVB DSTMP5,@TEMP4
11656						
11657					:+	
11658					::	SIGN DS.EXP BY DATA TYPE
11659					:-	
11660						
11661	037004				SELECT DATTYP OF 5	
11662	037004	013746	000464			MOV DATTYP,-(SP)
11663	037010	002561				BLT \$51503
11664	037012	006316				ASL (SP)
11665	037014	062716	037022			ADD #51474,(SP)
11666	037020	013607				MOV @ (SP)+,PC
11667	037022					\$51474:
11668	037022	037354				.WORD \$51502
11669	037024	037036				.WORD \$51501
11670	037026	037104				.WORD \$51500
11671	037030	037162				.WORD \$51477
11672	037032	037240				.WORD \$51476
11673	037034	037304				.WORD \$51475
11674	037036				CASE 1	
11675	037036				LET DESCR4 := DATA04 ;SIGNED ZONED	\$51501:
11676	037036					
11677	037036	013737	000572	000626	IF DATA04 NE #0 THEN	MOV DATA04,DESCR4
11678	037044					
11679	037044	005737	000572			TST DATA04
11680	037050	001414				BEQ \$51504
11681	037052				IF SIGN2 EQ #C THEN	
11682	037052	005737	000454			TST SIGN2
11683	037056	001004				BNE \$51505
11684	037060				LET @TEMP4 :B= @TEMP4 SET.BY #60	
11685	037060	152777	000060	141420	ELSE	BISB #60,@TEMP4
11686	037066					
11687	037066	000403				BR \$51506
11688	037070					\$51505:
11689	037070				LET @TEMP4 :B= @TEMP4 SET.BY #160	
11690	037070	152777	000160	141410	ENDIF	BISB #160,@TEMP4
11691	037076					
11692	037076				LET DS.EXP-1 :B= #0	\$51506:
11693	037076					
11694	037076	105037	064441		ENDIF	CLRB DS.EXP-1
11695	037102					
11696	037102					\$51504:
11697						

11698 037102
11699 037102 000524
11700 037104
11701 037104
11702 037104 013737 000572 000626
11703 037112 052737 020000 000626
11704 037120
11705 037120 005737 000572
11706 037124 001415
11707 037126
11708 037126 113705 000561
11709 037132
11710 037132 162705 000060
11711 037136
11712 037136 023727 000454 177777
11713 037144 001002
11714 037146
11715 037146 062705 000012
11716 037152
11717 037152
11718 037152
11719 037152 116577 056530 141326
11720 037160
11721 037160
11722
11723 037160
11724 037160 000475
11725 037162
11726 037162
11727 037162 013737 000572 000626
11728 037170 052737 030000 000626
11729 037176
11730 037176 005737 000572
11731 037202 001415
11732 037204
11733 037204 113705 000560
11734 037210
11735 037210 162705 000060
11736 037214
11737 037214 023727 000454 177777
11738 037222 001002
11739 037224
11740 037224 062705 000012
11741 037230
11742 037230
11743 037230
11744 037230 116537 056530 064442
11745 037236
11746 037236
11747
11748 037236
11749 037236 000446
11750 037240
11751 037240
11752 037240 013737 000572 000626
11753 037246 052737 040000 000626

CASE 2

LET DESCR4 := DATA04 SET.BY #20000 ;TRAILING OVERPUNCH

IF DATA04 NE #0 THEN

LET R5 :B= DSTMP5

LET R5 := R5 - #60

IF SIGN2 EQ #-1 THEN

LET R5 := R5 + #10.

ENDIF

LET @TEMP4 :B= OVPNCH(R5)

ENDIF

CASE 3

LET DESCR4 := DATA04 SET.BY #30000 ;LEADING OVERPUNCH

IF DATA04 NE #0 THEN

LET R5 :B= DSTMP4

LET R5 := R5 - #60

IF SIGN2 EQ #-1 THEN

LET R5 := R5 + #10.

ENDIF

LET DS.EXP :B= OVPNCH(R5)

ENDIF

CASE 4

LET DESCR4 := DATA04 SET.BY #40000 ;TRAILING SEPARATE

BR \$51503
\$51500:
MOV DATA04,DESCR4
BIS #20000,DESCR4
TST DATA04
BEQ \$51507
MOVB DSTMP5,R5
SUB #60,R5
CMP SIGN2,#-1
BNE \$51510
ADD #10.,R5
\$51510:
MOVB OVPNCH(R5),@TEMP4
\$51507:

BR \$51503
\$51477:
MOV DATA04,DESCR4
BIS #30000,DESCR4
TST DATA04
BEQ \$51511
MOVB DSTMP4,R5
SUB #60,R5
CMP SIGN2,#-1
BNE \$51512
ADD #10.,R5
\$51512:
MOVB OVPNCH(R5),DS.EXP
\$51511:

BR \$51503
\$51476:
MOV DATA04,DESCR4
BIS #40000,DESCR4

11754	037254			LET R5 := TEMP4 + #1	
11755	037254	013705	000506		MOV TEMP4,R5
11756	037260	005205			INC R5
11757	037262			IF SIGN2 EQ #0 THEN	
11758	037262	005737	000454		TST SIGN2
11759	037266	001003			BNE \$51513
11760	037270			LET (R5) :B= #53	
11761	037270	112715	000053		MOVB #53,(R5)
11762	037274			ELSE	
11763	037274	000402			BR \$51514
11764	037276				\$51513:
11765	037276			LET (R5) :B= #55	
11766	037276	112715	000055		MOVB #55,(R5)
11767	037302			ENDIF	
11768	037302				\$51514:
11769					
11770	037302			CASE 5	
11771	037302	000424			BR \$51503
11772	037304				\$51475:
11773	037304			LET DESCR4 := DATA04 SET.BY #50000	
11774	037304	013737	000572	000626	MOV DATA04,DESCR4
11775	037312	052737	050000	000626	BIS #50000,DESCR4
11776	037320			LET R5 := TEMP4 + #1	
11777	037320	013705	000506		MOV TEMP4,R5
11778	037324	005205			INC R5
11779	037326			LET (R5) :B= #0	
11780	037326	105015			CLRB (R5)
11781	037330			IF SIGN2 EQ #0 THEN	
11782	037330	005737	000454		TST SIGN2
11783	037334	001004			BNE \$51515
11784	037336			LET DS.EXP-1 :B= #53	
11785	037336	112737	000053	064441	MOVB #53,DS.EXP-1
11786	037344			ELSE	
11787	037344	000403			BR \$51516
11788	037346				\$51515:
11789	037346			LET DS.EXP-1 :B= #55	
11790	037346	112737	000055	064441	MOVB #55,DS.EXP-1
11791	037354			ENDIF	
11792	037354				\$51516:
11793	037354			ENDSELECT	
11794	037354				\$51502:
11795	037354				\$51503:
11796					
11797					
11798				:+	
11799				:-	CLR DS.DST & LOAD DESCRIPTORS
11800					
11801	037354			LET R0 := #DS.DST-1	
11802	037354	012700	064377		MOV #DS.DST-1,R0
11803	037360			LET R1 := #41	
11804	037360	012701	000041		MOV #41,R1
11805	037364			INLINE <DOIT: CLRB (R0)+>	
11806	037364	105020			DOIT: CLRB (R0)+
11807	037366			INLINE <SOB R1, DOIT>	
11808	037366	077102			SOB R1, DOIT
11809	037370			LET R0 := DESCRO	

11810	037370	013700	000616		MOV	DESCR0,R0
11811	037374			LET R1 := DESCR1		
11812	037374	013701	000620		MOV	DESCR1,R1
11813	037400			LET R2 := DESCR2		
11814	037400	013702	000622		MOV	DESCR2,R2
11815	037404			LET R3 := DESCR3		
11816	037404	013703	000624		MOV	DESCR3,R3
11817	037410			LET R4 := DESCR4		
11818	037410	013704	000626		MOV	DESCR4,R4
11819	037414			LET R5 := DESCR5		
11820	037414	013705	000630		MOV	DESCR5,R5
11821	037420			LET R4.EXP := R4		
11822	037420	010437	000534		MOV	R4,R4.EXP
11823	037424			LET R5.EXP := R5		
11824	037424	010537	000536		MOV	R5,R5.EXP
11825	037430			INLINE <SUBN>		
11826	037430	076051			SUBN	
11827						
11828						
11829				::: CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
11830				::-		
11831						
11832	037432			LET CC := PSW		
11833	037432	013737	177776		MOV	PSW,CC
11834	037440			LET CC := CC CLR.BY #177760		
11835	037440	042737	177760		BIC	#177760,CC
11836	037446			IF CC NE CC.EXP THEN		
11837	037446	023737	000520		CMP	CC,CC.EXP
11838	037454	001402			BEQ	\$51517
11839	037456			CALL ERROR		
11840	037456	004737	060044		JSR	PC,ERROR
11841	037462			ENDIF		
11842	037462				\$51517:	
11843	037462			IF R0 NE R0.EXP THEN		
11844	037462	020037	000524		CMP	R0,R0.EXP
11845	037466	001402			BEQ	\$51520
11846	037470			CALL ERROR		
11847	037470	004737	060044		JSR	PC,ERROR
11848	037474			ENDIF		
11849	037474				\$51520:	
11850	037474			IF R1 NE R1.EXP THEN		
11851	037474	020137	000526		CMP	R1,R1.EXP
11852	037500	001402			BEQ	\$51521
11853	037502			CALL ERROR		
11854	037502	004737	060044		JSR	PC,ERROR
11855	037506			ENDIF		
11856	037506				\$51521:	
11857	037506			IF R2 NE R2.EXP THEN		
11858	037506	020237	000530		CMP	R2,R2.EXP
11859	037512	001402			BEQ	\$51522
11860	037514			CALL ERROR		
11861	037514	004737	060044		JSR	PC,ERROR
11862	037520			ENDIF		
11863	037520				\$51522:	
11864	037520			IF R3 NE R3.EXP THEN		
11865	037520	020337	000532		CMP	R3,R3.EXP

11866	037524	001402			BEQ	\$51523
11867	037526			CALL ERROR	JSR	PC,ERROR
11868	037526	004737	060044			
11869	037532			ENDIF		\$51523:
11870	037532					
11871	037532			IF R4 NE R4.EXP THEN		
11872	037532	020437	000534		CMP	R4,R4.EXP
11873	037536	001402			BEQ	\$51524
11874	037540			CALL ERROR		
11875	037540	004737	060044		JSR	PC,ERROR
11876	037544			ENDIF		\$51524:
11877	037544					
11878	037544			IF R5 NE R5.EXP THEN		
11879	037544	020537	000536		CMP	R5,R5.EXP
11880	037550	001402			BEQ	\$51525
11881	037552			CALL ERROR		
11882	037552	004737	060044		JSR	PC,ERROR
11883	037556			ENDIF		\$51525:
11884	037556					
11885						
11886						
11887						
11888				:+		
11889				:-		
11890				:-		
11891	037556			LET R0 := #DS.DST-1		
11892	037556	012700	064377		MOV	#DS.DST-1,R0
11893	037562			LET R1 := #DS.EXP-1		
11894	037562	012701	064441		MOV	#DS.EXP-1,R1
11895	037566			LET ERRFLG := #0		
11896	037566	005037	000456		CLR	ERRFLG
11897	037572			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO		
11898	037572					\$51526:
11899	037572	020027	064440		CMP	R0,#DS.DST+40
11900	037576	001410			BEQ	\$51527
11901	037600	005737	000456		TST	ERRFLG
11902	037604	001005			BNE	\$51527
11903	037606			IFB (R0)+ NE (R1)+ THEN		
11904	037606	122021			CMPB	(R0)+,(R1)+
11905	037610	001402			BEQ	\$51530
11906	037612			CALL ERROR		
11907	037612	004737	060044		JSR	PC,ERROR
11908	037616			ENDIF		\$51530:
11909	037616					
11910	037616			ENDDO		
11911	037616	000765			BR	\$51526
11912	037620					\$51527:
11913				:ENDINC		
11914	037620			INLINE <JMP INC23>		JMP INC23
11915	037620	000137	036734			
11916	037624			INLINE <END23:>		END23:
11917	037624					
11918				:ENDINC		
11919	037624			INLINE <JMP INC22>		JMP INC22
11920	037624	000137	036336			
11921	037630			INLINE <END22:>		

11922	037630						END22:
11923					:ENDINC		
11924	037630				INLINE <JMP INC21>		JMP INC21
11925	037630	000137	035744				
11926	037634				INLINE <END21:>		END21:
11927	037634						
11928	037634				LET TSTCAS := TSTCAS + #22		ADD #22,TSTCAS
11929	037634	062737	000022	000544			
11930	037642				LET COUNT := COUNT - #1		DEC COUNT
11931	037642	005337	000470				
11932					:UNTIL COUNT EQ #0		
11933	037646				IF COUNT NE #0 THEN		
11934	037646	005737	000470				TST COUNT
11935	037652	001402					BEQ \$51531
11936	037654				INLINE <JMP \$5051>		JMP \$5051
11937	037654	000137	035366				
11938	037660				ENDIF		\$51531:
11939	037660						
11940							
11941					:+		
11942					:+ INLINE CASE - SUBNI, USING UNSIGNED ZONED DATA		
11943					:+		
11944					:+		
11945	037660				CALL CLRDSB		
11946	037660	004737	057040				JSR PC,CLRDSB
11947							
11948					:+		
11949					:+ LOAD DSBUF1		
11950					:+		
11951					:+		
11952	037664				LET R0 := #ADD1+2	:SOURCE	
11953	037664	012700	065264				MOV #ADD1+2,R0
11954	037670				LET R1 := TADDI CLR.BY #10000	:# OF BYTES TO XFER	
11955	037670	013701	065756				MOV TADDI,R1
11956	037674	042701	010000				BIC #10000,R1
11957	037700				LET R2 := #DSBUF1	:DESTINATION	
11958	037700	012702	064274				MOV #DSBUF1,R2
11959	037704				INLINE <1\$: MOVB (R0)+, (R2)+>		1\$: MOVB (R0)+, (R2)+
11960	037704	112022					
11961	037706				INLINE < SOB R1, 1\$>		SOB R1, 1\$
11962	037706	077102					
11963							
11964					:+		
11965					:+ LOAD DSBUF2		
11966					:+		
11967					:+		
11968	037710				LET R0 := #ADD1A+2		
11969	037710	012700	065306				MOV #ADD1A+2,R0
11970	037714				LET R1 := TADDI+4 CLR.BY #10000		
11971	037714	013701	065762				MOV TADDI+4,R1
11972	037720	042701	010000				BIC #10000,R1
11973	037724				LET R2 := #DSBUF2		
11974	037724	012702	064336				MOV #DSBUF2,R2
11975	037730				INLINE <2\$: MOVB (R0)+, (R2)+>		2\$: MOVB (R0)+, (R2)+
11976	037730	112022					
11977	037732				INLINE < SOB R1, 2\$>		

11978	037732	077102				SOB	R1,	2\$
11979								
11980								
11981								
11982								
11983								
11984	037734							
11985	037734	012700	065360			MOV	#ADD1C+2,R0	
11986	037740							
11987	037740	013701	065766			MOV	TADDI+10,R1	
11988	037744	042701	010000			BIC	#10000,R1	
11989	037750							
11990	037750	012702	064442			MOV	#DS.EXP,R2	
11991	037754							
11992	037754	112012				3\$:	MOVB (R0)+, (R2)	
11993	037756							
11994	037756	152722	000060				BISB #60, (R2)+	
11995	037762							
11996	037762	077104					SOB R1, 3\$	
11997								
11998	037764							
11999	037764	004737	056604			CALL CLRREG		
12000								
12001	037770							
12002	037770	076151						
12003	037772							
12004	037772	065756						
12005	037774							
12006	037774	065762						
12007	037776							
12008	037776	065766						
12009	040000							
12010	040000	013737	177776	000520		MOV	PSW,CC	
12011	040006							
12012	040006	042737	177760	000520		LET CC := CC CLR.BY #177760		
12013	040014							
12014	040014	005737	000520			IF CC NE #0 THEN		
12015	040020	001402						
12016	040022							
12017	040022	004737	060044			CALL ERROR		
12018	040026							
12019	040026					ENDIF		
12020								
12021								
12022								
12023								
12024								
12025	040026							
12026	040026	005700						
12027	040030	001004						
12028	040032	005701						
12029	040034	001002						
12030	040036	005702						
12031	040040	001402						
12032	040042							
12033	040042					CALL ERROR		

```

SOB R1, 2$
MOV #ADD1C+2,R0
MOV TADDI+10,R1
BIC #10000,R1
MOV #DS.EXP,R2
3$: MOVB (R0)+, (R2)
BISB #60, (R2)+
SOB R1, 3$
JSR PC,CLRREG
SUBNI
.WORD TADDI
.WORD TADDI+4
.WORD TADDI+10
MOV PSW,CC
BIC #177760,CC
TST CC
BEQ $51532
JSR PC,ERROR
$51532:
TST R0
BNE $51533
TST R1
BNE $51533
TST R2
BEQ $51534
$51533:
    
```

12034	040042	004737	060044		JSR	PC,ERROR
12035	040046			ENDIF		
12036	040046				\$51534:	
12037	040046			IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN		
12038	040046	005703			TST	R3
12039	040050	001004			BNE	\$51535
12040	040052	005704			TST	R4
12041	040054	001002			BNE	\$51535
12042	040056	005705			TST	R5
12043	040060	001402			BEQ	\$51536
12044	040062				\$51535:	
12045	040062			CALL ERROR		
12046	040062	004737	060044	ENDIF	JSR	PC,ERROR
12047	040066				\$51536:	
12048	040066					
12049						
12050				:+		
12051				:+ CHECK RESULTS		
12052				:-		
12053						
12054	040066			LET R0 := #DS.DST		
12055	040066	012700	064400		MOV	#DS.DST,R0
12056	040072			LET R1 := #DS.EXP ;EXPECTED DATA		
12057	040072	012701	064442		MOV	#DS.EXP,R1
12058	040076			LET ERRFLG := #0		
12059	040076	005037	000456		CLR	ERRFLG
12060	040102			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO		
12061	040102				\$51537:	
12062	040102	020027	064440		CMP	R0,#DS.DST+40
12063	040106	001410			BEQ	\$51540
12064	040110	005737	000456		TST	ERRFLG
12065	040114	001005			BNE	\$51540
12066	040116			IFB (R0)+ NE (R1)+ THEN		
12067	040116	122021			CMPB	(R0)+,(R1)+
12068	040120	001402			BEQ	\$51541
12069	040122			CALL ERROR		
12070	040122	004737	060044	ENDIF	JSR	PC,ERROR
12071	040126				\$51541:	
12072	040126					
12073	040126			ENDDO		
12074	040126	000765			BR	\$51537
12075	040130				\$51540:	
12076						
12077						

```
12078 .SBTTL CMPN
12079 :*****
12080 :
12081 :   CMPN TEST
12082 :
12083 :*****
12084
12085 040130  INLINE <XCMPN:>
12086 040130  XCMCN:
12087 040130  INLINE <SCOPE>
12088 040130 000004  SCOPE
12089 040132  LET $TIMES := #20 ;DO 20 ITERATIONS
12090 040132 012737 000020 000404  MOV #20,$TIMES
12091 040140  LET COUNT := TCMP ;GET TEST COUNT
12092 040140 013737 064502 000470  MOV TCMP,COUNT
12093 040146  LET TSTCAS := #TCMP+2 ;POINT TO FIRST TEST CASE
12094 040146 012737 064504 000544  MOV #TCMP+2,TSTCAS
12095 ;REPEAT
12096 040154  INLINE <$5052:>
12097 040154  $5052:
12098 040154  CALL CLRDSB
12099 040154 004737 057040  JSR PC,CLRDSB
12100 040160  LET CC.EXP := #0
12101 040160 005037 000522  CLR CC.EXP
12102 040164  LET SIGN0 := #0
12103 040164 005037 000450  CLR SIGN0
12104 040170  LET DAT TYP := #0
12105 040170 005037 000464  CLR DAT TYP
12106
12107 ;+
12108 ; GET SOURCE DESCRIPTORS FROM TABLE
12109 ;-
12110
12111 040174  LET R0 := TSTCAS
12112 040174 013700 000544  MOV TSTCAS,R0
12113 040200  LET DATA00 := (R0)+ ;SRC1 LENGTH
12114 040200 012037 000562  MOV (R0)+,DATA00
12115 040204  LET DATA01 := (R0)+ ;SRC1 POINTER
12116 040204 012037 000564  MOV (R0)+,DATA01
12117 040210  LET DATA02 := (R0)+ ;SRC2 LENGTH
12118 040210 012037 000566  MOV (R0)+,DATA02
12119 040214  LET DATA03 := (R0) ;SRC2 POINTER
12120 040214 011037 000570  MOV (R0),DATA03
12121
12122 ;+
12123 ; LOAD SRC1 AND SRC2 INTO DSBUF1 AND DSBUF2.
12124 ;-
12125
12126 040220  LET R2 := DATA01 + #2 ;GET SRC1 ADDRESS
12127 040220 013702 000564  MOV DATA01,R2
12128 040224 062702 000002  ADD #2,R2
12129 040230  LET R3 := #DSBUF1
12130 040230 012703 064274  MOV #DSBUF1,R3
12131 040234  LET DSTMP0 :=B= (R2) ;SAVE SRC1 MSD
12132 040234 111237 000554  MOVB (R2),DSTMP0
12133 040240  DECR R1 FROM DATA00 TO #1 BY #1
```

12134	040240	013701	000562		MOV	DATA00,R1
12135	040244	000401			BR	\$51542
12136	040246				\$51543:	
12137	040246	005301			DEC	R1
12138	040250				\$51542:	
12139	040250	020127	000001		CMP	R1,#1
12140	040254	002402			BLT	\$51544
12141	040256			LET (R3)+ :B= (R2)+		
12142	040256	112223			MOVB	(R2)+,(R3)+
12143	040260			ENDDEC		
12144	040260	000772			BR	\$51543
12145	040262				\$51544:	
12146	040262			IF DATA00 HI #0 THEN		
12147	040262	005737	000562		TST	DATA00
12148	040266	101402			BLOS	\$51545
12149	040270			LET DSTMP1 :B= -(R2)	;SAVE SRC1 LSD	
12150	040270	114237	000555		MOVB	-(R2),DSTMP1
12151	040274			ENDIF		
12152	040274				\$51545:	
12153	040274			LET R2 := DATA03 + #2	;GET SRC2 ADDRESS	
12154	040274	013702	000570		MOV	DATA03,R2
12155	040300	062702	000002		ADD	#2,R2
12156	040304			LET R3 := #DSBUF2	MOV	#DSBUF2,R3
12157	040304	012703	064336		MOV	#DSBUF2,R3
12158	040310			LET DSTMP2 :B= (R2)	;SAVE SRC2 MSD	
12159	040310	111237	000556		MOVB	(R2),DSTMP2
12160	040314			DECR R1 FROM DATA02 TO #1 BY #1		
12161	040314	013701	000566		MOV	DATA02,R1
12162	040320	000401			BR	\$51546
12163	040322				\$51547:	
12164	040322	005301			DEC	R1
12165	040324				\$51546:	
12166	040324	020127	000001		CMP	R1,#1
12167	040330	002402			BLT	\$51550
12168	040332			LET (R3)+ :B= (R2)+		
12169	040332	112223			MOVB	(R2)+,(R3)+
12170	040334			ENDDEC		
12171	040334	000772			BR	\$51547
12172	040336				\$51550:	
12173	040336			LET DSTMP3 :B= -(R2)	;SAVE SRC2 LSD	
12174	040336	114237	000557		MOVB	-(R2),DSTMP3
12175						
12176				:+		
12177				: COMPARE SRC1 AND SRC2, AND SET N AND Z BITS		
12178				: IN CC.EXP ACCORDINGLY.		
12179				:-		
12180						
12181	040342			IF DATA00 EQ #0 AND DATA02 EQ #0 THEN	;SRC1 LENGTH AND SRC2 LENGTH = 0	
12182	040342	005737	000562		TST	DATA00
12183	040346	001007			BNE	\$51551
12184	040350	005737	000566		TST	DATA02
12185	040354	001004			BNE	\$51551
12186	040356			LET CC.EXP := CC.EXP SET.BY #ZBIT		
12187	040356	052737	000004 000522		BIS	#ZBIT,CC.EXP
12188	040364			ELSE		
12189	040364	000424			BR	\$51552

12190	040366								\$51551:
12191	040366								TST DATA00
12192	040366	005737	000562						BNE \$51553
12193	040372	001007							TST @DATA03
12194	040374	005777	140170						BNE \$51553
12195	040400	001004							
12196	040402								BIS #NBIT,CC.EXP
12197	040402	052737	000010	000522					BR \$51554
12198	040410								\$51553:
12199	040410	000412							;SRC2 LENGTH = 0 & SRC1 SIGN IS -
12200	040412								TST DATA02
12201	040412								BNE \$51555
12202	040412	005737	000566						IF DATA02 EQ #0 AND @DATA01 EQ #-1 THEN
12203	040416	001007							CMPI @DATA01,#-1
12204	040420	027727	140140	177777					BNE \$51555
12205	040426	001003							
12206	040430								LET CC.EXP := CC.EXP SET.BY #NBIT
12207	040430	052737	000010	000522					BIS #NBIT,CC.EXP
12208	040436								ENDIF
12209	040436								\$51555:
12210	040436								ENDIF
12211	040436								\$51554:
12212	040436								ENDIF
12213	040436								\$51552:
12214	040436								;SRC1 LENGTH AND SRC2 LENGTH > 0
12215	040436	005737	000562						TST DATA00
12216	040442	001520							BEQ \$51556
12217	040444	005737	000566						TST DATA02
12218	040450	001515							BEQ \$51556
12219	040452								;SRC1 SIGN AND SRC2 SIGN ARE DIFFERENT
12220	040452	027777	140106	140110					CMPI @DATA01,@DATA03
12221	040460	001410							BEQ \$51557
12222	040462								;SRC1 SIGN IS -
12223	040462	027727	140076	177777					CMPI @DATA01,#-1
12224	040470	001003							BNE \$51560
12225	040472								LET CC.EXP := CC.EXP SET.BY #NBIT
12226	040472	052737	000010	000522					BIS #NBIT,CC.EXP
12227	040500								ENDIF
12228	040500								\$51560:
12229	040500								;SRC1 SIGN AND SRC2 SIGN ARE EQUAL
12230	040500	000501							BR \$51561
12231	040502								\$51557:
12232	040502								;SAVE THE COMMON SIGN
12233	040502	017737	140056	000450					MOV @DATA01,SIGN0
12234	040510								;SRC1 LENGTH < SRC2 LENGTH AND SIGNS
12235	040510	023737	000562	000566					CMPI DATA00,DATA02
12236	040516	103007							BHIS \$51562
12237	040520	005737	000450						TST SIGN0
12238	040524	001004							BNE \$51562
12239	040526								LET CC.EXP := CC.EXP SET.BY #NBIT
12240	040526	052737	000010	000522					BIS #NBIT,CC.EXP
12241	040534								ELSE
12242	040534	000413							BR \$51563
12243	040536								\$51562:
12244	040536								;SRC1 LENGTH > SRC2 LENGTH AND
12245	040536	023737	000562	000566					CMPI DATA00,DATA02

12246 040544 101407
 12247 040546 023727 000450 177777
 12248 040554 001003
 12249 040556
 12250 040556 052737 000010 000522
 12251 040564
 12252 040564
 12253 040564
 12254 040564
 12255 040564
 12256 040564 023737 000562 000566
 12257 040572 001044
 12258
 12259
 12260
 12261
 12262
 12263 040574
 12264 040574 013700 000562
 12265 040600
 12266 040600 012701 064274
 12267 040604
 12268 040604 012702 064336
 12269 040610
 12270 040610
 12271 040610 105700
 12272 040612 001404
 12273 040614 022122
 12274 040616 001002
 12275 040620
 12276 040620 005300
 12277 040622
 12278 040622 000772
 12279 040624
 12280 040624
 12281 040624 124142
 12282 040626 001423
 12283 040630
 12284 040630 121112
 12285 040632 103007
 12286 040634 005737 000450
 12287 040640 001004
 12288 040642
 12289 040642 052737 000010 000522
 12290 040650
 12291 040650 000411
 12292 040652
 12293 040652
 12294 040652 121112
 12295 040654 101407
 12296 040656 023727 000450 177777
 12297 040664 001003
 12298 040666
 12299 040666 052737 000010 000522
 12300 040674
 12301 040674

```

    LET CC.EXP := CC.EXP SET.BY #NBIT
  ENDIF
ENDIF
IF DATA00 EQ DATA02 THEN

```

```

  :+
  : COMPARE SRC1 AND SRC2 BYTE BY BYTE
  :-

```

```

  LET R0 := DATA00 ;SRC BUFFER LENGTH
  MOV DATA00,R0
  LET R1 := #DSBUF1 ;SRC1 BUFFER
  MOV #DSBUF1,R1
  LET R2 := #DSBUF2 ;SRC2 BUFFER
  MOV #DSBUF2,R2
  WHILEB R0 NE #0 AND (R1)+ EQ (R2)+ DO

```

```

    LET R0 := R0 - #1 ;DECREMENT COUNTER
  ENDDO
  DEC R0

```

```

  IFB -(R1) NE -(R2) THEN ;SOURCES WERE NOT =
  CMPB -(R1),-(R2)
  BEQ $51570

```

```

    IFB (R1) LO (R2) AND SIGN0 EQ #0 THEN
  CMPB (R1),(R2)
  BHIS $51571
  TST SIGN0
  BNE $51571

```

```

    LET CC.EXP := CC.EXP SET.BY #NBIT
  ELSE
  BR $51572

```

```

  IFB (R1) HI (R2) AND SIGN0 EQ #-1 THEN
  CMPB (R1),(R2)
  BLOS $51573
  CMP SIGN0,#-1
  BNE $51573

```

```

    LET CC.EXP := CC.EXP SET.BY #NBIT
  ENDIF
  BR $51573

```

```

BLOS $51564
CMP SIGN0,#-1
BNE $51564
BIS #NBIT,CC.EXP
$51564:
$51563:
;SRC1 LENGTH AND SRC2 LENGTH ARE EQUAL
CIP DATA00,DATA02
BNE $51565

```

```

$51566:
TSTB R0
BEQ $51567
CMP (R1)+,(R2)+
BNE $51567

```

```

BR $51566
$51567:
CMPB -(R1),-(R2)
BEQ $51570

```

```

CMPB (R1),(R2)
BHIS $51571
TST SIGN0
BNE $51571

```

```

BIS #NBIT,CC.EXP
BR $51572
$51571:

```

```

CMPB (R1),(R2)
BLOS $51573
CMP SIGN0,#-1
BNE $51573

```

```

BIS #NBIT,CC.EXP
$51573:

```

12302	040674				ENDIF		
12303	040674					\$51572:	
12304	040674				ELSE	; SOURCES WERE EQUAL	
12305	040674	000403				BR	\$51574
12306	040676					\$51570:	
12307	040676				LET CC.EXP := CC.EXP SET.BY #ZBIT		
12308	040676	052737	000004	000522		BIS	#ZBIT,CC.EXP
12309	040704				ENDIF		
12310	040704					\$51574:	
12311	040704				ENDIF		
12312	040704					\$51565:	
12313	040704				ENDIF		
12314	040704					\$51561:	
12315	040704				ENDIF		
12316	040704					\$51556:	
12317							
12318							
12319					:*		
12320					: PERFORM THE CMPN INSTRUCTION WITH ALL COMBINATIONS		
12321					: OF SIGNED ZONE, TRAILING AND LEADING OVERPUNCH, AND		
12322					: TRAILING AND LEADING SEPARATE (UNSIGNED ZONE WILL		
12323					: BE TESTED IN THE INLINE PORTION OF THE TEST).		
12324					: CHECK THE RESULTS AND REPORT ERRORS. 'DATTYP'		
12325					: WILL CONTAIN SRC1 DATA TYPE AND 'TEMP1' WILL		
12326					: CONTAIN SRC2 DATA TYPE CURRENTLY BEING TESTED.		
12327					: -		
12328							
12329	040704				:REPEAT		
12330	040704				INLINE <\$5201:>		
12331	040704						
12332	040704	023727	000464	010000	IF DATTYP EQ #10000 THEN	;SKIP UNSIGNED ZONED	
12333	040712	001003				CMP	DATTYP,#10000
12334	040714				LET DATTYP := DATTYP + #10000	BNE	\$51575
12335	040714	062737	010000	000464		ADD	#10000,DATTYP
12336	040722				ENDIF		
12337	040722					\$51575:	
12338					:RESTORE THE MSD AND LSD OF SRC1		
12339	040722	113737	000554	064274	LET DSBUF1 :B= DSTMP0	;SRC1 M.	
12340	040722					MOVB	DSTMP0,DSBUF1
12341	040730				LET RO := #DSBUF1-1 + DATA00		
12342	040730	012700	064273			MOV	#DSBUF1-1,RO
12343	040734	063700	000562			ADD	DATA00,RO
12344	040740				LEI (RO) :B= DSTMP1	;SRC1 LSD	
12345	040740	113710	000555			MOVB	DSTMP1,(RO)
12346					:PUT THE PROPER SIGN INTO SRC1, ACCORDING TO DATTYP.		
12347	040744				IF DATTYP EQ #0 THEN	;*** SIGNED ZONED ***	
12348	040744	005737	000464			TST	DATTYP
12349	040750	001017				BNE	\$51576
12350	040752				LET RO := #DSBUF1-1 + DATA00		
12351	040752	012700	064273			MOV	#DSBUF1-1,RO
12352	040756	063700	000562			ADD	DATA00,RO
12353	040762				IF @DATA01 EQ #0 THEN	;SIGN IS +	
12354	040762	005777	137576			TST	@DATA01
12355	040766	001003				BNE	\$51577
12356	040770				LET (RO) :B= (RO) SET.BY #60		
12357	040770	152710	000060			BISB	#60,(RO)

12358	040774			ELSE		;SIGN IS -	
12359	040774	000402					BR \$51600
12360	040776						\$51577:
12361	040776			LET (R0) :B= (R0) SET.BY #160			BISB #160,(R0)
12362	040776	152710	000160	ENDIF			\$51600:
12363	041002						
12364	041002			LET R0 := #DSBUF1-1			MOV #DSBUF1-1,R0
12365	041002						;RESTORE AFTER LEADING SEPARATE
12366	041002	012700	064273	LET (R0) :B= #0			CLRB (R0)
12367	041006						
12368	041006	105010		ENDIF			\$51576:
12369	041010						;*** TRAILING OVERPUNCH ***
12370	041010			IF DAT TYP EQ #20000 THEN			CMP DAT TYP,#20000
12371	041010						BNE \$51601
12372	041010	023727	000464 020000				
12373	041016	001021		IF DATA00 HI #0 THEN			TST DATA00
12374	041020						BLOS \$51602
12375	041020	005737	000562	LET R0 := #DSBUF1-1 + DATA00			MOV #DSBUF1-1,R0
12376	041024	101416					ADD DATA00,R0
12377	041026						
12378	041026	012700	064273	IF @DATA01 EQ #0 THEN		;SIGN IS +	TST @DATA01
12379	041032	063700	000562				BNE \$51603
12380	041036			LET R1 := #0			CLR R1
12381	041036	005777	137522	ELSE		;SIGN IS -	BR \$51604
12382	041042	001002					\$51603:
12383	041044			LET R1 := #10.			MOV #10.,R1
12384	041044	005001		ENDIF			\$51604:
12385	041046			LET R1 :B= R1 SET.BY (R0)		;FORM TABLE OFFSET	BISB (R0),R1
12386	041046	000402		LET (R0) :B= OVPNCH(R1)		;PUT IN SIGN/DIGIT	MOVB OVPNCH(R1),(R0)
12387	041050			ENDIF			\$51602:
12388	041050			ENDIF			\$51601:
12389	041050	012701	000012	IF DAT TYP EQ #30000 THEN		;*** LEADING OVERPUNCH ***	CMP DAT TYP,#30000
12390	041054						BNE \$51605
12391	041054			IF DATA00 HI #0 THEN			TST DATA00
12392	041054						BLOS \$51606
12393	041054	151001		LET R0 := #DSBUF1			MOV #DSBUF1,R0
12394	041056			IF @DATA01 EQ #0 THEN		;SIGN IS +	TST @DATA01
12395	041056	116110	056530				BNE \$51607
12396	041062			LET R1 := #0			CLR R1
12397	041062			ELSE		;SIGN IS -	
12398	041062						
12399	041062						
12400	041062						
12401	041062	023727	000464 030000				
12402	041070	001017					
12403	041072						
12404	041072	005737	000562				
12405	041076	101414					
12406	041100						
12407	041100	012700	064274				
12408	041104						
12409	041104	005777	137454				
12410	041110	001002					
12411	041112						
12412	041112	005001					
12413	041114						

12414	041114	000402				BR	\$51610
12415	041116					\$51607:	
12416	041116				LET R1 := #10.	MOV	#10.,R1
12417	041116	012701	000012		ENDIF		
12418	041122					\$51610:	
12419	041122				LET R1 :B= P1 SET.BY (R0)	:FORM TABLE OFFSET	
12420	041122					BISB	(R0),R1
12421	041122	151001			LET (R0) :B= OVPNCH(R1)	:PUT IN SIGN/DIGIT	
12422	041124					MOVB	OVPNCH(R1),(R0)
12423	041124	116110	056530		ENDIF		
12424	041130				ENDIF	\$51606:	
12425	041130					\$51605:	
12426	041130				IF DATTYP EQ #40000 THEN	;*** TRAILING SEPARATE ***	
12427	041130					CMP	DATTYP,#40000
12428	041130					BNE	\$51611
12429	041130	023727	000464	040000	LET R0 := #DSBUF1 + DATA00	MOV	#DSBUF1,R0
12430	041136	001014				ADD	DATA00,R0
12431	041140					;SIGN IS +	
12432	041140	012700	064274		IF @DATA01 EQ #0 THEN	TST	@DATA01
12433	041144	063700	000562			BNE	\$51612
12434	041150					MOVB	#53,(R0)
12435	041150	005777	137410		LET (R0) :B= #53	;SIGN IS -	
12436	041154	001003			ELSE	BR	\$51613
12437	041156					\$51612:	
12438	041162	112710	000053		LET (R0) :B= #55	MOVB	#55,(R0)
12439	041162				ENDIF		
12440	041162	000402			ENDIF	\$51613:	
12441	041164					\$51611:	
12442	041164				IF DATTYP EQ #50000 THEN	;*** LEADING SEPARATE ***	
12443	041164	112710	000055			CMP	DATTYP,#50000
12444	041170				LET R0 := #DSBUF1-1	BNE	\$51614
12445	041170					MOV	#DSBUF1-1,R0
12446	041170				IF @DATA01 EQ #0 THEN	;SIGN IS +	
12447	041170					TST	@DATA01
12448	041170				LET (R0) :B- #53	BNE	\$51615
12449	041170	023727	000464	050000	ELSE	MOVB	#53,(R0)
12450	041176	001017				;SIGN IS -	
12451	041200				LET (R0) :B- #55	BR	\$51616
12452	041200	012700	064273			\$51615:	
12453	041204				ENDIF	MOVB	#55,(R0)
12454	041204	005777	137354		LET R0 := #DSBUF1 + DATA00		
12455	041210	001003				\$51616:	
12456	041212				LET (R0) :B= #0	MOV	#DSBUF1,R0
12457	041212	112710	000053			ADD	DATA00,R0
12458	041216					;RESTORE AFTER TRAILING SEPARATE	
12459	041216	000402				(LRB	(R0)
12460	041220						
12461	041220						
12462	041220	112710	000055				
12463	041224						
12464	041224						
12465	041224						
12466	041224	012700	064274				
12467	041230	063700	000562				
12468	041234						
12469	041234	105010					

12470	041236				ENDIF	
12471	041236				LET TEMP1 := #0	\$51614: ;CLEAR SRC2 DATA TYPE
12472	041236					CLR TEMP1
12473	041236	005037	000500			
12474					;REPEAT	
12475	041242				INLINE <\$5202:>	
12476	041242					\$5202:
12477	041242				IF TEMP1 EQ #10000 THEN	;SKIP UNSIGNED ZONED
12478	041242	023727	000500	010000		CMP TEMP1,#10000
12479	041250	001003				BNE \$51617
12480	041252				LET TEMP1 := TEMP1 + #10000	
12481	041252	062737	010000	000500		ADD #10000,TEMP1
12482	041260				ENDIF	
12483	041260					\$51617:
12484					;RESTORE THE MSD AND LSD OF SRC2	
12485	041260				LET DSBUF2 :B= DSTMP2	;SRC2 MSD
12486	041260	113737	000556	064336		MOVB DSTMP2,DSBUF2
12487	041266				LET R0 := #DSBUF2-1 + DATA02	
12488	041266	012700	064335			MOV #DSBUF2-1,R0
12489	041272	063700	000566			ADD DATA02,R0
12490	041276				LET (R0) :B= DSTMP3	;SRC2 LSD
12491	041276	113710	000557			MOVB DSTMP3,(R0)
12492					;PUT THE PROPER SIGN INTO SRC2, ACCORDING TO TEMP1	
12493	041302				IF TEMP1 EQ #0 THEN	;*** SIGNED ZONED ***
12494	041302	005737	000500			TST TEMP1
12495	041306	001017				BNE \$51620
12496	041310				LET R0 := #DSBUF2-1 + DATA02	
12497	041310	012700	064335			MOV #DSBUF2-1,R0
12498	041314	063700	000566			ADD DATA02,R0
12499	041320				IF @DATA03 EQ #0 THEN	;SIGN IS +
12500	041320	005777	137244			TST @DATA03
12501	041324	001003				BNE \$51621
12502	041326				LET (R0) :B= (R0) SET.BY #60	
12503	041326	152710	000060			BISB #60,(R0)
12504	041332				ELSE	;SIGN IS -
12505	041332	000402				BR \$51622
12506	041334					\$51621:
12507	041334				LET (R0) :B= (R0) SET.BY #160	
12508	041334	152710	000160			BISB #160,(R0)
12509	041340				ENDIF	
12510	041340					\$51622:
12511	041340				LET R0 := #DSBUF2-1	
12512	041340	012700	064335			MOV #DSBUF2-1,R0
12513	041344				LET (R0) :B= #0	;RESTORE AFTER LEADING SEPARATE
12514	041344	105010				CLRB (R0)
12515	041346				ENDIF	
12516	041346					\$51620:
12517	041346				IF TEMP1 EQ #20000 THEN	;*** TRAILING OVERPUNCH ***
12518	041346	023727	000500	020000		CMP TEMP1,#20000
12519	041354	001016				BNE \$51623
12520	041356				LET R0 := #DSBUF2-1 + DATA02	
12521	041356	012700	064335			MOV #DSBUF2-1,R0
12522	041362	063700	000566			ADD DATA02,R0
12523	041366				IF @DATA03 EQ #0 THEN	;SIGN IS +
12524	041366	005777	137176			TST @DATA03
12525	041372	001002				BNE \$51624

12526	041374				LET R1 := #0	
12527	041374	005001			ELSE	;SIGN IS -
12528	041376					BR \$51625
12529	041376	000402				\$51624:
12530	041400				LET R1 := #10.	MOV #10.,R1
12531	041400				ENDIF	\$51625:
12532	041400	012701	000012		LET R1 :B= R1 SET.BY (R0)	;FORM TABLE OFFSET
12533	041404				LET (R0) :B= OVPNCH(R1)	BISB (R0),R1
12534	041404				ENDIF	;PUT IN SIGN/DIGIT
12535	041404					MOVB OVPNCH(R1),(R0)
12536	041404	151001			IF TEMP1 EQ #30000 THEN	\$51623:
12537	041406					*** LEADING OVERPUNCH ***
12538	041406	116110	056530			CMP TEMP1,#30000
12539	041412					BNE \$51626
12540	041412				LET R0 := #DSBUF2	MOV #DSBUF2,R0
12541	041412				IF @DATA03 EQ #0 THEN	;SIGN IS +
12542	041412	023727	000500	030000		TST @DATA03
12543	041420	001014				BNE \$51627
12544	041422				LET R1 := #0	CLR R1
12545	041422	012700	064336		ELSE	;SIGN IS -
12546	041426					BR \$51630
12547	041426	005777	137136			\$51627:
12548	041432	001002			LET R1 := #10.	MOV #10.,R1
12549	041434				ENDIF	\$51630:
12550	041434	005001			LET R1 :B= R1 SET.BY (R0)	;FORM TABLE OFFSET
12551	041436				LET (R0) :B= OVPNCH(R1)	BISB (R0),R1
12552	041436	000402			ENDIF	;PUT IN SIGN/DIGIT
12553	041440				IF TEMP1 EQ #40000 THEN	MOVB OVPNCH(R1),(R0)
12554	041440					\$51626:
12555	041440	012701	000012		LET R0 := #DSBUF2 + DATA02	*** TRAILING SEPARATE ***
12556	041444				IF @DATA03 EQ #0 THEN	CMP TEMP1,#40000
12557	041444					BNE \$51631
12558	041444				LET (R0) :B= #53	MOV #DSBUF2,R0
12559	041444	151001			ELSE	ADD DATA02,R0
12560	041446					;SIGN IS +
12561	041446	116110	056530		LET (R0) :B= #55	TST @DATA03
12562	041452					BNE \$51632
12563	041452				ENDIF	MOVB #53,(R0)
12564	041452	023727	000500	040000		;SIGN IS -
12565	041452	001014				BR \$51633
12566	041460					\$51632:
12567	041462				LET (R0) :B= #55	MOVB #55,(R0)
12568	041462	012700	064336		ENDIF	\$51633:
12569	041466	063700	000566			
12570	041472					
12571	041472	005777	137072			
12572	041476	001003				
12573	041500					
12574	041500	112710	000053			
12575	041504					
12576	041504	000402				
12577	041506					
12578	041506					
12579	041506	112710	000055			
12580	041512					
12581	041512					

12582	041512				ENDIF	
12583	041512					\$51631:
12584	041512				IF TEMP1 EQ #50000 THEN	;*** LEADING SEPARATE ***
12585	041512	023727	000500	050000		CMP TEMP1,#50000
12586	041520	001017				BNE \$51634
12587	041522				LET R0 := #DSBUF2-1	--
12588	041522	012700	064335			MOV #DSBUF2-1,R0
12589	041526				IF @DATA03 EQ #0 THEN	;SIGN IS +
12590	041526	005777	137036			TST @DATA03
12591	041532	001003				BNE \$51635
12592	041534				LET (R0) :B= #53	
12593	041534	112710	000053		ELSE	;SIGN IS -
12594	041540					MOVB #53,(R0)
12595	041540	000402				BR \$51636
12596	041542					\$51635:
12597	041542				LET (R0) :B= #55	
12598	041542	112710	000055		ENDIF	MOVB #55,(R0)
12599	041546					\$51636:
12600	041546				LET R0 := #DSBUF2 + DATA02	
12601	041546					MOV #DSBUF2,R0
12602	041546	012700	064336			ADD DATA02,R0
12603	041552	063700	000566			SEPARATE
12604	041556				LET (R0) :B= #0	;RESTORE AFTER TRAILING
12605	041556	105010			ENDIF	CLRB (R0)
12606	041560					\$51634:
12607	041560					
12608						
12609					;+ ; LOAD THE GPR'S AND DO THE CMPN INSTRUCTION ;-	
12610						
12611						
12612						
12613	041560				LET R0 := DATTYP SET.BY DATA00	;SRC1.DSCR
12614	041560	013700	000464			MOV DATTYP,R0
12615	041564	053700	000562			BIS DATA00,R0
12616	041570				LET R1 := #DSBUF1	; "
12617	041570	012701	064274			MOV #DSBUF1,R1
12618	041574				LET R2 := TEMP1 SET.BY DATA02	;SRC2.DSCR
12619	041574	013702	000500			MOV TEMP1,R2
12620	041600	053702	000566			BIS DATA02,R2
12621	041604				LET R3 := #DSBUF2	; "
12622	041604	012703	064336			MOV #DSBUF2,R3
12623	041610				LET R4 := #0	
12624	041610	005004				CLR R4
12625	041612				LET R5 := #0	
12626	041612	005005				CLR R5
12627	041614				INLINE <CMPN>	
12628	041614	076052				CMPN
12629						
12630					;+ ; CHECK THE RESULTS ;-	
12631						
12632						
12633						
12634	041616				LET CC := PSW	
12635	041616	013737	177776	000520		MOV PSW,CC
12636	041624				LET CC := CC CLR.BY #177760	
12637	041624	042737	177760	000520		BIC #177760,CC

12638	041632				IF CC NE CC.EXP THEN		
12639	041632	023737	000520	000522		CMP	CC,CC.EXP
12640	041640	001402				BEQ	\$51637
12641	041642				CALL ERROR		
12642	041642	004737	060044		ENDIF	JSR	PC,ERROR
12643	041646						\$51637:
12644	041646				IF R0 NE #0 THEN		
12645	041646					TST	R0
12646	041646	005700				BEQ	\$51640
12647	041650	001402			CALL ERROR		
12648	041652				ENDIF	JSR	PC,ERROR
12649	041652	004737	060044				\$51640:
12650	041656				IF R1 NE #0 THEN		
12651	041656					TST	R1
12652	041656	005701			CALL ERROR	BEQ	\$51641
12653	041656	001402			ENDIF	JSR	PC,ERROR
12654	041660				IF R2 NE #0 THEN		
12655	041662					TST	R2
12656	041662	004737	060044		CALL ERROR	BEQ	\$51642
12657	041666				ENDIF	JSR	PC,ERROR
12658	041666				IF R3 NE #0 THEN		
12659	041666					TST	R3
12660	041666	005702			CALL ERROR	BEQ	\$51643
12661	041670	001402			ENDIF	JSR	PC,ERROR
12662	041672				IF R4 NE #0 THEN		
12663	041672	004737	060044		CALL ERROR		
12664	041676				ENDIF		\$51642:
12665	041676				IF R3 NE #0 THEN		
12666	041676					TST	R3
12667	041676	005703			CALL ERROR	BEQ	\$51643
12668	041700	001402			ENDIF	JSR	PC,ERROR
12669	041702				IF R4 NE #0 THEN		
12670	041702	004737	060044		CALL ERROR		
12671	041706				ENDIF		\$51643:
12672	041706				IF R5 NE #0 THEN		
12673	041706					TST	R4
12674	041706	005704			CALL ERROR	BEQ	\$51644
12675	041710	001402			ENDIF	JSR	PC,ERROR
12676	041712				IF R5 NE #0 THEN		
12677	041712	004737	060044		CALL ERROR		
12678	041716				ENDIF		\$51644:
12679	041716				IF R5 NE #0 THEN		
12680	041716					TST	R5
12681	041716	005705			CALL ERROR	BEQ	\$51645
12682	041720	001402			ENDIF	JSR	PC,ERROR
12683	041722				LET TEMP1 := TEMP1 + #10000		\$51645:
12684	041722	004737	060044			:UPDATE	SRC2 TO NEXT DATA TYPE
12685	041726					ADD	#10000,TEMP1
12686	041726				:UNTIL TEMP1 EQ #60000		
12687	041726				IF TEMP1 NE #60000 THEN		
12688	041726	062737	010000	000500			
12689					INLINE <JMP \$5202>		
12690	041734						
12691	041734	023727	000500	060000		CMP	TEMP1,#60000
12692	041742	001402				BEQ	\$51646
12693	041744						

12694	041744	000137	041242				JMP	\$5202
12695	041750				ENDIF			
12696	041750				LET DATTYP := DATTYP + #10000			\$51646:
12697	041750						:UPDATE	SRC1 TO NEXT DATA TYPE
12698	041750	062737	010000	000464			ADD	#10000,DATTYP
12699					:UNTIL DATTYP EQ #60000			
12700	041756				IF DATTYP NE #60000 THEN			
12701	041756	023727	000464	060000			CMP	DATTYP,#60000
12702	041764	001402					BEQ	\$51647
12703	041766				INLINE <JMP \$5201>			
12704	041766	000137	040704		ENDIF		JMP	\$5201
12705	041772							\$51647:
12706	041772							
12707								
12708	041772				LET TSTCAS := TSTCAS + #10			
12709	041772	062737	000010	000544			ADD	#10,TSTCAS
12710	042000				LET COUNT := COUNT - #1			
12711	042000	005337	000470				DEC	COUNT
12712					:UNTIL COUNT EQ #0 THEN			
12713	042004				IF COUNT NE #0 THEN			
12714	042004	005737	000470				TST	COUNT
12715	042010	001402					BEQ	\$51650
12716	042012				INLINE <JMP \$5052>			
12717	042012	000137	040154		ENDIF		JMP	\$5052
12718	042016							\$51650:
12719	042016							
12720								
12721								
12722					:+ DO ONE INLINE CASE - CMPNI. USE UNSIGNED ZONED DATA TYPE.			
12723					:-			
12724								
12725								
12726	042016				LET R0 := #-1		MOV	#-1,R0
12727	042016	012700	177777					
12728	042022				LET R1 := #-1		MOV	#-1,R1
12729	042022	012701	177777					
12730	042026				LET R2 := #-1		MOV	#-1,R2
12731	042026	012702	177777					
12732	042032				LET R3 := #-1		MOV	#-1,R3
12733	042032	012703	177777					
12734	042036				LET R4 := #-1		MOV	#-1,R4
12735	042036	012704	177777					
12736	042042				LET R5 := #-1		MOV	#-1,R5
12737	042042	012705	177777					
12738	042046				INLINE <CMPNI>			
12739	042046	076152					CMPNI	
12740	042050				INLINE <.WORD TCMPI>			
12741	042050	042164					.WORD	TCMPI
12742	042052				INLINE <.WORD TCMPI+4>			
12743	042052	042170					.WORD	TCMPI+4
12744	042054				LET CC := PSW			
12745	042054	013737	177776	000520			MOV	PSW,CC
12746	042062				LET CC := CC CLR.BY #177760			
12747	042062	042737	177760	000520			BIC	#177760,CC
12748	042070				IF CC NE #4 THEN			
12749	042070	023727	000520	000004			CMP	CC,#4

```

12750 042076 001402          BEQ      $51651
12751 042100          CALL ERROR
12752 042100 004737 060044   JSR      PC,ERROR
12753 042104          ENDIF
12754 042104
12755 042104          IF R0 NE #-1 OR R1 NE #-1 OR R2 NE #-1 THEN
12756 042104 020027 177777   CMP      R0,#-1
12757 042110 001006          BNE      $51652
12758 042112 020127 177777   CMP      R1,#-1
12759 042116 001003          BNE      $51652
12760 042120 020227 -177777  CMP      R2,#-1
12761 042124 001402          BEQ      $51653
12762 042126          $51652:
12763 042126          CALL ERROR
12764 042126 004737 060044   JSR      PC,ERROR
12765 042132          ENDIF
12766 042132
12767 042132          IF R3 NE #-1 OR R4 NE #-1 OR R5 NE #-1 THEN
12768 042132 020327 177777   CMP      R3,#-1
12769 042136 001006          BNE      $51654
12770 042140 020427 177777   CMP      R4,#-1
12771 042144 001003          BNE      $51654
12772 042146 020527 177777   CMP      R5,#-1
12773 042152 001402          BEQ      $51655
12774 042154          $51654:
12775 042154          CALL ERROR
12776 042154 004737 060044   JSR      PC,ERROR
12777 042160          ENDIF
12778 042160          $51655:
12779 042160          INLINE <JMP XASHN>
12780 042160 000137 042202   JMP      XASHN
12781
12782
12783
12784          ;+
12785          ; INLINE TEST CASE
12786          ;-
12787 042164 010005          TCMPI:  .WORD 010005          ;SRC1 DATA TYPE = UNSIGNED ZONED, LENGTH 5
12788 042166 042174          .WORD  CMPN11
12789 042170 010005          .WORD 010005          ;SRC2 DATA TYPE - UNSIGNED ZONED, LENGTH - 5
12790 042172 042174          .WORD  CMPN11
12791
12792 042174 001 002 003 003 CMPN11: .BYTE 001,002,003,004,065 ;DECIMAL STRING - 12345
12793 042177 004 065
12794 042202
12795
    
```

```

12796 .SBTTL ASHN
12797 :*****
12798 :
12799 : ASHN TEST
12800 :*****
12801 :
12802 :
12803 042202 INLINE <XASHN:> XASHN:
12804 042202
12805 042202 INLINE <SCOPE> SCOPE
12806 042202 000004
12807 042204 LET $TIMES := #20 ;DO 20 ITERATIONS
12808 042204 012737 000020 000404 MOV #20,$TIMES
12809 042212 LET COUNT := TASH ;GET TEST COUNT
12810 042212 013737 064750 000470 MOV TASH,COUNT
12811 042220 LET TSTCAS :- #TASH+2 ;POINT TO FIRST TEST CASE
12812 042220 012737 064752 000544 MOV #TASH+2,TSTCAS
12813 042226 LET SIGNO := #0
12814 042226 005037 000450 CLR SIGNO
12815 :REPEAT
12816 042232 INLINE <$5056:> $5056:
12817 042232
12818 042232 LET CC.EXP := #0 CLR CC.EXP
12819 042232 005037 000522
12820 042236 LET DATTYP :- #0 CLR DATTYP
12821 042236 005037 000464
12822 :
12823 :+
12824 : GET TEST DATA FROM TABLE
12825 :-
12826 :
12827 042242 LET R0 := TSTCAS
12828 042242 013700 000544 MOV TSTCAS,R0
12829 042246 LET DATA00 := (R0)+ ;SRC.LENGTH
12830 042246 012037 000562 MOV (R0)+,DATA00
12831 042252 LET DATA01 :- (R0)+ ;DST.LENGTH
12832 042252 012037 000564 MOV (R0)+,DATA01
12833 042256 LET DATA02 := (R0) ;SHIFT.DSCR
12834 042256 011037 000566 MOV (R0),DATA02
12835 042262 CALL CLRDSB
12836 042262 004737 057040 JSR PC,CLRDSB
12837 :
12838 :+
12839 : PUT TEST STRING INTO DSBUF1 WITHOUT THE SIGN
12840 :-
12841 :
12842 042266 LET R0 :- DATA00 ;SRC.LENGTH
12843 042266 013700 000562 MOV DATA00,R0
12844 042272 LET R1 := #DSBUF1 ;SRC.ADDRESS
12845 042272 012701 064274 MOV #DSBUF1,R1
12846 042276 WHILE R0 NE #0 DO
12847 042276 $51656:
12848 042276 005700 TST R0
12849 042300 001404 BEQ $51657
12850 042302 LET (R1)+ :B- #4
12851 042302 112721 000004 MOVB #4,(R1)+
    
```

12852	042306				LET R0 := R0 - #1		
12853	042306	005300				DEC	R0
12854	042310				ENDDO		
12855	042310	000772				BR	\$51656
12856	042312					\$51657:	
12857							
12858					:+		
12859					:+ LOAD DS.EXP AND CC.EXP WITH EXPECTED RESULTS.		
12860					:-		
12861							
12862					:GET SHIFT COUNT MAGNITUDE AND PUT IT INTO TEMPO		
12863							
12864	042312				LET TEMPO := DATA02		
12865	042312	013737	000566	000476		MOV	DATA02,TEMPO
12866	042320				IF #BIT07 SET IN TEMPO THEN	:NEGATIVE SHIFT	COUNT
12867	042320	032737	000200	000476		BIT	#BIT07,TEMPO
12868	042326	001402				BEQ	\$51660
12869	042330				LET TEMPO := NEGATE TEMPO		
12870	042330	005437	000476			NEG	TEMPO
12871	042334				ENDIF		
12872	042334					\$51660:	
12873	042334				LET TEMPO := TEMPO CLR.BY #177600		
12874	042334	042737	177600	000476		BIC	#177600,TEMPO
12875							
12876					:TAKE CARE OF EXPECTED NBIT		
12877							
12878	042342				IF SIGN0 EQ #-1 AND DATA00 NE #0 THEN		
12879	042342	023727	000450	177777		CMP	SIGN0,#-1
12880	042350	001006				BNE	\$51661
12881	042352	005737	000562			TST	DATA00
12882	042356	001403				BEQ	\$51661
12883	042360				LET CC.EXP := CC.EXP SET.BY #NBIT		
12884	042360	052737	000010	000522		BIS	#NBIT,CC.EXP
12885	042366				ENDIF		
12886	042366					\$51661:	
12887							
12888					:TAKE CARE OF ZERO LENGTH SOURCE OR DESTINATION		
12889							
12890	042366				IF DATA01 NE #0 THEN		
12891	042366	005737	000564			TST	DATA01
12892	042372	001414				BEQ	\$51662
12893	042374				LET R0 := #DS.EXP		
12894	042374	012700	064442			MOV	#DS.EXP,R0
12895	042400				INCR R1 FROM #1 TO DATA01 BY #1		
12896	042400	012701	000001			MOV	#1,R1
12897	042404	000401				BR	\$51663
12898	042406					\$51664:	
12899	042406	005201				INC	R1
12900	042410					\$51663:	
12901	042410	020137	000564			CMP	R1,DATA01
12902	042414	003003				BGT	\$51665
12903	042416				LET (R0)+ :B= #60		
12904	042416	112720	000060			MOVB	#60,(R0)+
12905	042422				ENDINC		
12906	042422	000771				BR	\$51664
12907	042424					\$51665:	

12908	042424				ENDIF		
12909	042424					\$51662:	
12910	042424				IF DATA00 EQ #0 OR DATA01 EQ #0 THEN		
12911	042424	005737	000562			TST	DATA00
12912	042430	001403				BEQ	\$51666
12913	042432	005737	000564			TST	DATA01
12914	042436	001004				BNE	\$51667
12915	042440					\$51666:	
12916	042440				LET CC.EXP := CC.EXP SET.BY #ZBIT		
12917	042440	052737	000004	000522		BIS	#ZBIT,CC.EXP
12918	042446				ELSE		
12919	042446	000567				BR	\$51670
12920	042450					\$51667:	
12921							
12922					;TAKE CARE OF NEGATIVE SHIFT		
12923							
12924	042450				IF #BIT07 SET IN DATA02 THEN		
12925	042450	032737	000200	000566		BIT	#BIT07,DATA02
12926	042456	001465				BEQ	\$51671
12927	042460				LET R0 := DATA00 - TEMPO	;SRC.LEN - SHIFT COUNT	
12928	042460	013700	000562			MOV	DATA00,R0
12929	042464	163700	000476			SUB	TEMPO,R0
12930	042470				LET R1 := #DS.EXP + DATA01		
12931	042470	012701	064442			MOV	#DS.EXP,R1
12932	042474	063701	000564			ADD	DATA01,R1
12933	042500				LET R2 := #1		
12934	042500	012702	000001			MOV	#1,R2
12935	042504				LET R3 := DATA01		
12936	042504	013703	000564			MOV	DATA01,R3
12937	042510				IF R0 EQ #0 THEN		
12938	042510	005700				TST	R0
12939	042512	001012				BNE	\$51672
12940	042514				IF DATA02 HIS #3000 THEN		
12941	042514	023727	000566	003000		CMP	DATA02,#3000
12942	042522	103403				BLO	\$51673
12943	042524				LET -(R1) :B= #61	;ROUNDING	
12944	042524	112741	000061			MOVB	#61,-(R1)
12945	042530				ELSE		
12946	042530	000403				RR	\$51674
12947	042532					\$51673:	
12948	042532				LET CC.EXP := CC.EXP SET.BY #ZBIT		
12949	042532	052737	000004	000522		BIS	#ZBIT,CC.EXP
12950	042540				ENDIF		
12951	042540					\$51674:	
12952	042540				ENDIF		
12953	042540					\$51672:	
12954	042540				IF R0 LT #0 THEN		
12955	042540	005700				TST	R0
12956	042542	002003				BGE	\$51675
12957	042544				LET CC.EXP := CC.EXP SET.BY #ZBIT		
12958	042544	052737	000004	000522		BIS	#ZBIT,CC.EXP
12959	042552				ENDIF		
12960	042552					\$51675:	
12961	042552				WHILE R2 LE R0 AND R3 HI #0 DO		
12962	042552					\$51676:	
12963	042552	020200				CMP	R2,R0

12964 042554 003017
 12965 042556 005703
 12966 042560 101415
 12967 042562
 12968 042562 112741 000064
 12969 042566
 12970 042566 020227 000001
 12971 042572 001005
 12972 042574 023727 000566 003000
 12973 042602 103401
 12974 042604
 12975 042604 105211
 12976 042606
 12977 042606
 12978 042606
 12979 042606 005202
 12980 042610
 12981 042610 005303
 12982 042612
 12983 042612 000757
 12984 042614
 12985 042614
 12986 042614 005703
 12987 042616 001005
 12988 042620 020200
 12989 042622 003003
 12990 042624
 12991 042624 052737 000002 000522
 12992 042632
 12993 042632
 12994 042632
 12995 042632
 12996
 12997
 12998
 12999 042632
 13000 042632 032737 000200 000566
 13001 042640 001072
 13002 042642
 13003 042642 013737 000562 000500
 13004 042650 063737 000476 000500
 13005 042656
 13006 042656 023737 000500 000564
 13007 042664 101403
 13008 042666
 13009 042666 052737 000002 000522
 13010 042674
 13011 042674
 13012 042674
 13013 042674 023737 000476 000564
 13014 042702 103420
 13015 042704
 13016 042704 052737 000004 000522
 13017 042712
 13018 042712 012701 064442
 13019 042716

LET -(R1) :B= #64
 IF R2 EQ #1 AND DATA02 HIS #3000 THEN
 LET (R1) :B= (R1) + #1
 ENDIF
 LET R2 := R2 + #1
 LET R3 := R3 - #1
 ENDDO
 IF R3 EQ #0 AND R2 LE R0 THEN
 LET CC.EXP := CC.EXP SET.BY #VBIT
 ENDIF
 ENDIF
 ;TAKE CARE OF POSITIVE SHIFT
 IF #BIT07 NOTSET IN DATA02 THEN
 LET TEMP1 := DATA00 + TEMPO
 IF TEMP1 HI DATA01 THEN
 LET CC.EXP := CC.EXP SET.BY #VBIT
 ENDIF
 IF TEMPO HIS DATA01 THEN
 LET CC.EXP := CC.EXP SET.BY #ZBIT
 LET R1 := #DS.EXP
 INCR R0 FROM #1 TO DATA01 BY #1

BGT \$51677
 TST R3
 BLOS \$51677
 MOVB #64, -(R1)
 CMP R2, #1
 BNE \$51700
 CMP DATA02, #3000
 BLO \$51700
 ;ROUNDING
 INCB (R1)
 \$51700:
 INC R2
 DEC R3
 BR \$51676
 \$51677:
 TST R3
 BNE \$51701
 CMP R2, R0
 BGT \$51701
 BIS #VBIT, CC.EXP
 \$51701:
 \$51671:
 BIT #BIT07, DATA02
 BNE \$51702
 ;SRC.LEN + SHIFT COUNT
 MOV DATA00, TEMP1
 ADD TEMPO, TEMP1
 CMP TEMP1, DATA01
 BLOS \$51703
 BIS #VBIT, CC.EXP
 \$51703:
 ;SHIFT COUNT >= DST.LEN
 CMP TEMPO, DATA01
 BLO \$51704
 BIS #ZBIT, CC.EXP
 MOV #DS.EXP, R1

13020	042716	012700	000001		MOV	#1,R0
13021	042722	000401			BR	\$51705
13022	042724				\$51706:	
13023	042724	005200			INC	R0
13024	042726				\$51705:	
13025	042726	020037	000564		CMP	R0,DATA01
13026	042732	003003			BGT	\$51707
13027	042734			LET (R1)+ :B= #60		
13028	042734	112721	000060		MOVB	#60,(R1)+
13029	042740			ENDINC		
13030	042740	000771			BR	\$51706
13031	042742				\$51707:	
13032	042742			ELSE	:SHIFT COUNT < DST.LEN	
13033	042742	000431			BR	\$51710
13034	042744				\$51704:	
13035	042744			LET R1 := #DS.EXP + DATA01		
13036	042744	012701	064442		MOV	#DS.EXP,R1
13037	042750	063701	000564		ADD	DATA01,R1
13038	042754			INCR R0 FROM #1 TO TEMPO BY #1		
13039	042754	012700	000001		MOV	#1,R0
13040	042760	000401			BR	\$51711
13041	042762				\$51712:	
13042	042762	005200			INC	R0
13043	042764				\$51711:	
13044	042764	020037	000476		CMP	R0,TEMPO
13045	042770	003003			BGT	\$51713
13046	042772			LET -(R1) :B= #60		
13047	042772	112741	000060		MOVB	#60,-(R1)
13048	042776			ENDINC		
13049	042776	000771			BR	\$51712
13050	043000				\$51713:	
13051	043000			LET R0 := DATA00	:SRC.LEN	
13052	043000	013700	000562		MOV	DATA00,R0
13053	043004			WHILE R1 HI #DS.EXP AND R0 HI #0 DO		
13054	043004				\$51714:	
13055	043004	020127	064442		CMP	R1,#DS.EXP
13056	043010	101406			BLOS	\$51715
13057	043012	005700			TST	R0
13058	043014	101404			BLOS	\$51715
13059	043016			LET -(R1) :B= #64		
13060	043016	112741	000064		MOVB	#64,-(R1)
13061	043022			LET R0 := R0 - #1		
13062	043022	005300			DEC	R0
13063	043024			ENDDO		
13064	043024	000767			BR	\$51714
13065	043026				\$51715:	
13066	043026			ENDIF	\$51710:	
13067	043026			ENDIF	\$51702:	
13068	043026			ENDIF	\$51670:	
13069	043026			IF #ZBIT SET IN CC.EXP THEN		
13070	043026					
13071	043026					
13072	043026					
13073	043026	032737	000004 000522		BIT	#ZBIT,CC.EXP
13074	043034	001403			BEQ	\$51716
13075	043036			LET CC.EXP := CC.EXP CLR.BY #NBIT		

13076	043036	042737	000010	000522		BIC	#NBIT,CC.EXP
13077	043044				ENDIF		
13078	043044					\$51716:	
13079							
13080					:SAVE THE FIRST AND LAST EXPECTED DST DIGITS		
13081							
13082	043044				LET DSTMP0 :B= DS.EXP		
13083	043044	113737	064442	000554		MOVB	DS.EXP,DSTMP0
13084	043052				LET R0 := #DS.EXP + DATA01 - #1		
13085	043052	012700	064442			MOV	#DS.EXP,R0
13086	043056	063700	000564			ADD	DATA01,R0
13087	043062	005300				DEC	R0
13088	043064				LET DSTMP1 .B= (R0)		
13089	043064	111037	000555			MOVB	(R0),DSTMP1
13090							
13091					:+ : PERFORM THE ASHN INSTRUCTION WITH ALL SIX NUMERIC : DATA TYPES. 'DATTYP' WILL CONTAIN THE DATA TYPE : CURRENTLY BEING TESTED. :-		
13092							
13093							
13094							
13095							
13096					:INCR DATTYP FROM #0 TO #50000 BY #10000		
13097					INLINE <CLR DATTYP>		
13098	043070					CLR	DATTYP
13099	043070	005037	000464				
13100	043074				INLINE <BR \$5602>		
13101	043074	000403				BR	\$5602
13102	043076				INLINE <\$5601:>		
13103	043076					\$5601:	
13104	043076				LET DATTYP := DATTYP + #10000		
13105	043076	062737	010000	000464		ADD	#10000,DATTYP
13106	043104				INLINE <\$5602:>		
13107	043104					\$5602:	
13108	043104				IF DATTYP GT #50000 THEN		
13109	043104	023727	000464	050000		CMP	DATTYP,#50000
13110	043112	003402				BLE	\$51717
13111	043114				INLINE <JMP \$5603>		
13112	043114	000137	044674			JMP	\$5603
13113	043120				ENDIF		
13114	043120					\$51717:	
13115							
13116					:RESTORE FIRST AND LAST SRC DIGITS		
13117	043120				IF DATA00 HI #0 THEN	:SRC.LEN > 0	
13118	043120	005737	000562			TST	DATA00
13119	043124	101411				BLOS	\$51720
13120	043126				LET DSBUF1 :B= #4		
13121	043126	112737	000004	064274		MOVB	#4,DSBUF1
13122	043134				LET R0 := #DSBUF1-1 + DATA00		
13123	043134	012700	064273			MOV	#DSBUF1-1,R0
13124	043140	063700	000562			ADD	DATA00,R0
13125	043144				LET (R0) :B= #4		
13126	043144	112710	000004			MOVB	#4,(R0)
13127	043150				ENDIF		
13128	043150					\$51720:	
13129					:RESTORE FIRST AND LAST DST DIGITS		
13130	043150				IF DATA01 HI #0 THEN	:DST.LEN > 0	
13131	043150	005737	000564			TST	DATA01


```

13132 043154 101411
13133 043156
13134 043156 113737 000554 064442
13135 043164
13136 043164 012700 064441
13137 043170 063700 000564
13138 043174
13139 043174 113710 000555
13140 043200
13141 043200
13142
13143 043200
13144 043200 005737 000464
13145 043204 001067
13146 043206
13147 043206 105037 064273
13148 043212
13149 043212 005737 000562
13150 043216 101414
13151 043220
13152 043220 012700 064273
13153 043224 063700 000562
13154 043230
13155 043230 005737 000450
13156 043234 001003
13157 043236
13158 043236 112710 000064
13159 043242
13160 043242 000402
13161 043244
13162 043244
13163 043244 112710 000164
13164 043250
13165 043250
13166 043250
13167 043250
13168 043250
13169 043250 105037 064441
13170 043254
13171 043254 005737 000564
13172 043260 101441
13173 043262
13174 043262 012700 064441
13175 043266 063700 000564
13176 043272
13177 043272 005737 000450
13178 043276 001407
13179 043300 005737 000562
13180 043304 001404
13181 043306 032737 000004 000522
13182 043314 001403
13183 043316
13184 043316
13185 043316 152710 000060
13186 043322
13187 043322 000402
    LET DS.EXP :B= DSTMP0
    LET RO := #DS.EXP-1 + DATA01
    LET (RO) :B= DSTMP1
ENDIF
;FINISH SETTING UP EXPECTED RESULTS ACCORDING TO DATA TYPE
IF DATTYP EQ #0 THEN
    LET DSBUF1-1 :B= #0
    IF DATA00 HI #0 THEN
        LET RO := #DSBUF1-1 + DATA00
        IF SIGN0 EQ #0 THEN
            LET (RO) :B= #64
        ELSE
            LET (RO) :B= #164
        ENDIF
    ENDIF
ENDIF
LET DS.EXP-1 :B= #0
IF DATA01 HI #0 THEN
    LET RO :- #DS.EXP-1 + DATA01
    IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SET IN CC.EXP THEN
        LET (RO) :B= (RO) SET.BY #60
    ELSE

```

```

BLOS $51721
MOVB DSTMP0,DS.EXP
MOV #DS.EXP-1,RO
ADD DATA01,RO
MOVB DSTMP1,(RO)
$51721:
;*** SIGNED ZONED ***
TST DATTYP
BNE $51722
;RESTORE SRC AFTER LEADING SEPARATE
CLRB DSBUF1-1
;SRC.LEN > 0
TST DATA00
BLOS $51723
MOV #DSBUF1-1,RO
ADD DATA00,RO
;SIGN IS +
TST SIGN0
BNE $51724
MOVB #64,(RO)
;SIGN IS -
BR $51725
$51724:
MOVB #164,(RO)
$51725:
$51723:
;RESTORE DST AFTER LEADING SEPARATE
CLRB DS.EXP-1
;DST.LEN > 0
TST DATA01
BLOS $51726
MOV #DS.EXP-1,RO
ADD DATA01,RO
;SIGN IS +
TST SIGN0
BEQ $51727
TST DATA00
BEQ $51727
BIT #ZBIT,CC.EXP
BEQ $51730
$51727:
BISB #60,(RO)
;SIGN IS -
BR $51731

```

13188	043324						\$51730:
13189	043324				LET (R0) :B= (R0) SET.BY #160		BISB #160,(R0)
13190	043324	152710	000160				
13191	043330				ENDIF		\$51731:
13192	043330						AND SIGN0 EQ #-1 THEN
13193	043330				IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02		CMPS DATA02,DATA00
13194	043330	123737	000566	000562			BHIS \$51732
13195	043336	103012					BIT #BIT07,DATA02
13196	043340	032737	000200	000566			BNE \$51732
13197	043346	001006					CMPS SIGN0,#-1
13198	043350	023727	000450	177777			BNE \$51732
13199	043356	001002					
13200	043360				LET (R0) :B= (R0) SET.BY #160		
13201	043360	152710	000160				BISB #160,(R0)
13202	043364				ENDIF		\$51732:
13203	043364				ENDIF		\$51726:
13204	043364				ENDIF		\$51722:
13205	043364						;*** UNSIGNED ZONED ***
13206	043364				IF DATTYP EQ #10000 THEN		CMPS DATTYP,#10000
13207	043364						BNE \$51733
13208	043364						
13209	043364	023727	000464	010000			BIC #NBIT,CC.EXP
13210	043372	001025			LET CC.EXP := CC.EXP CLR.BY #NBIT		;SRC.LEN > 0
13211	043374						TST DATA00
13212	043374	042737	000010	000522	IF DATA00 HI #0 THEN		BLOS \$51734
13213	043402						
13214	043402	005737	000562		LET R0 := #DSBUF1-1 + DATA00		MOV #DSBUF1-1,R0
13215	043406	101406					ADD DATA00,R0
13216	043410						
13217	043410	012700	064273		LET (R0) :B= #64		MOVB #64,(R0)
13218	043414	063700	000562				
13219	043420				ENDIF		\$51734:
13220	043420	112710	000064				;DST.LEN > 0
13221	043424				IF DATA01 HI #0 THEN		TST DATA01
13222	043424						BLOS \$51735
13223	043424						
13224	043424	005737	000564		LET R0 := #DS.EXP-1 + DATA01		MOV #DS.EXP-1,R0
13225	043430	101406					ADD DATA01,R0
13226	043432						
13227	043432	012700	064441		LET (R0) :B= (R0) SET.BY #60		BISB #60,(R0)
13228	043436	063700	000564				\$51735:
13229	043442				ENDIF		\$51733:
13230	043442	152710	000060		ENDIF		;*** TRAILING OVERPUNCH ***
13231	043446						CMPS DATTYP,#20000
13232	043446				IF DATTYP EQ #20000 THEN		BNE \$51736
13233	043446						
13234	043446						;SRC.LEN > 0
13235	043446						TST DATA00
13236	043446	023727	000464	020000	IF DATA00 HI #0 THEN		BLOS \$51737
13237	043454	001102					
13238	043456						
13239	043456	005737	000562		IF SIGN0 EQ #-1 AND #ZBIT NOTSETIN CC.EXP THEN		CMPS SIGN0,#-1
13240	043462	101427					BNE \$51740
13241	043464						
13242	043464	023727	000450	177777			
13243	043472	001007					

13244	043474	032737	000004	000522		BIT	#ZBIT,CC.EXP
13245	043502	001003				BNE	\$51740
13246	043504				LET CC.EXP := CC.EXP SET.BY #NBIT	:RESTORE	NBIF AFTER UNSIGNED ZONED
13247	043504	052737	000010	000522		BIS	#NBIF,CC.EXP
13248	043512				ENDIF		
13249	043512				LET R0 := #DSBUF1-1 + DATA00		\$51740:
13250	043512					MOV	#DSBUF1-1,R0
13251	043512	012700	064273			ADD	DATA00,R0
13252	043516	063700	000562		IF SIGN0 EQ #0 THEN	:SIGN IS +	
13253	043522					TST	SIGN0
13254	043522	005737	000450			BNE	\$51741
13255	043526	001003			LET (R0) :B= #104		
13256	043530				ELSE	:SIGN IS -	
13257	043530	112710	000104			MOVB	#104,(R0)
13258	043534					BR	\$51742
13259	043534	000402					\$51741:
13260	043536				LET (R0) :B= #115		
13261	043536					MOVB	#115,(R0)
13262	043536	112710	000115		ENDIF		
13263	043542				ENDIF		\$51742:
13264	043542				IF DATA01 HI #0 THEN	:DST.LEN > 0	
13265	043542					TST	DATA01
13266	043542					BLOS	\$51743
13267	043542				LET R0 := #DS.EXP-1 + DATA01		
13268	043542	005737	000564			MOV	#DS.EXP-1,R0
13269	043546	101445				ADD	DATA01,R0
13270	043550				IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SET IN CC.EXP THEN	:SIGN IS +	
13271	043550	012700	064441			TST	SIGN0
13272	043554	063700	000564			BEQ	\$51744
13273	043560					TST	DATA00
13274	043560	005737	000450			BEQ	\$51744
13275	043564	001407				BIT	#ZBIT,CC.EXP
13276	043566	005737	000562			BEQ	\$51745
13277	043572	001404					\$51744:
13278	043574	032737	000004	000522		CLRB	R1
13279	043602	001402			LET R1 :B= #0	:SIGN IS -	
13280	043604				ELSE	BR	\$51746
13281	043604						\$51745:
13282	043604	105001			LET R1 :B= #10.		
13283	043606					MOVB	#10.,R1
13284	043606	000402			ENDIF		\$51746:
13285	043610				IFB DATA02 LD DATA00 AND #BIT07 NOTSET IN DATA02	AND SIGN0 EQ #-1 THEN	
13286	043610	112701	000012			CMPB	DATA02,DATA00
13287	043614					BHIS	\$51747
13288	043614					BIT	#BIT07,DATA02
13289	043614					BNE	\$51747
13290	043614	123737	000566	000562		CMP	SIGN0,#-1
13291	043614	103012				BNE	\$51747
13292	043622	032737	000200	000566			
13293	043624	001006			LET R1 :B= #10.		
13294	043632	001006				MOVB	#10.,R1
13295	043634	023727	000450	177777			
13296	043642	001002			ENDIF		
13297	043644						
13298	043644	112701	000012				
13299	043650						

13300 043650
 13301 043650
 13302 043650 151001
 13303 043652
 13304 043652 042701 177760
 13305 043656
 13306 043656 116110 056530
 13307 043662
 13308 043662
 13309 043662
 13310 043662
 13311 043662
 13312 043662 023727 000464 030000
 13313 043670 001063
 13314 043672
 13315 043672 005737 000562
 13316 043676 101412
 13317 043700
 13318 043700 005737 000450
 13319 043704 001004
 13320 043706
 13321 043706 112737 000104 064274

LET R1 :B= R1 SET.BY (R0)
 LET R1 := R1 CLR.BY #177760
 LET (R0) :B= OVPNCH(R1)
 ENDIF
 ENDIF
 IF DATTYP EQ #30000 THEN
 IF DATA00 HI #0 THEN
 IF SIGN0 EQ #0 THEN
 LET DSBUF1 :B- #104

\$51747:
 BISB (R0),R1
 BIC #177760,R1
 MOVB OVPNCH(R1),(R0)
 \$51743:
 \$51736:
 :*** LEADING OVERPUNCH ***
 CMP DATTYP,#30000
 BNE \$51750
 :SRC.LEN > 0
 TST DATA00
 BLOS \$51751
 :SIGN IS +
 TST SIGN0
 BNE \$51752
 MOVB #104,DSBUF1

```

13322 043714          ELSE
13323 043714 000403
13324 043716
13325 043716          LET DSBUF1 :B= #115
13326 043716 112737 000115 064274
13327 043724          ENDIF
13328 043724          ENDIF
13329 043724          ENDIF
13330 043724          IF DATA01 HI #0 THEN
13331 043724          ;DST.LEN > 0
13332 043724 005737 000564          TST DATA01
13333 043730 101443          BLOS $51754
13334 043732          IF SIGN0 EQ #0 OR DATA00 EQ #0 OR #ZBIT SETIN CC.EXP THEN ;SIGN IS +
13335 043732 005737 000450          TST SIGN0
13336 043736 001407          BEQ $51755
13337 043740 005737 000562          TST DATA00
13338 043744 001404          BEQ $51755
13339 043746 032737 000004 000522          BIT #ZBIT,CC.EXP
13340 043754 001402          BEQ $51756
13341 043756          $51755:
13342 043756          LET R1 :B= #0
13343 043756 105001          ELSE
13344 043760          ;SIGN IS -
13345 043760 000402          BR $51757
13346 043762          $51756:
13347 043762          LET R1 :B= #10.
13348 043762 112701 000012          MOVB #10.,R1
13349 043766          ENDIF
13350 043766          $51757:
13351 043766          IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGN0 EQ #-1 THEN
13352 043766 123737 000566 000562          CMPB DATA02,DATA00
13353 043774 103012          BHIS $51760
13354 043776 032737 000200 000566          BIT #BIT07,DATA02
13355 044004 001006          BNE $51760
13356 044006 023727 000450 177777          CMP SIGN0,#-1
13357 044014 001002          BNE $51760
13358 044016          LET R1 :B= #10.
13359 044016 112701 000012          MOVB #10.,R1
13360 044022          ENDIF
13361 044022          $51760:
13362 044022          LET R1 :B= R1 SET.BY DS.EXP
13363 044022 153701 064442          BISB DS.EXP,R1
13364 044026          LET R1 := R1 CLR.BY #177760
13365 044026 042701 177760          BIC #177760,R1
13366 044032          LET DS.EXP :B= OVPNCH(R1)
13367 044032 116137 056530 064442          MOVB OVPNCH(R1),DS.EXP
13368 044040          ENDIF
13369 044040          $51754:
13370 044040          ENDIF
13371 044040          $51750:
13372 044040          IF DATTYP EQ #40000 THEN ;*** TRAILING SEPARATE ***
13373 044040 023727 000464 040000          CMP DATTYP,#40000
13374 044046 001071          BNE $51761
13375
13376          ;PUT SIGN INTO SOURCE
13377
    
```

13378	044050				LET TEMPO :- #DSBUF1 + DATA00		
13379	044050	012737	064274	000476		MOV	#DSBUF1,TEMPO
13380	044056	063737	000562	000476		ADD	DATA00,TEMPO
13381	044064				IF SIGN0 EQ #0 THEN	:SIGN IS +	
13382	044064	005737	000450			TST	SIGN0
13383	044070	001004				BNE	\$51762
13384	044072				LET @TEMPO :B- #53		
13385	044072	112777	000053	134376		MOVB	#53,@TEMPO
13386	044100				ELSE	:SIGN IS -	
13387	044100	000403				BR	\$51763
13388	044102					\$51762:	
13389	044102				LET @TEMPO :B= #55		
13390	044102	112777	000055	134366		MOVB	#55,@TEMPO
13391	044110				ENDIF		
13392	044110					\$51763:	
13393							
13394					:PUT EXPECTED SIGN INTO DS.EXP		
13395					LET TEMPO := #DS.EXP + DATA01		
13396	044110					MOV	#DS.EXP,TEMPO
13397	044110	012737	064442	000476		ADD	DATA01,TEMPO
13398	044116	063737	000564	000476			
13399	044124				IF SIGN0 EQ #0 OR #ZBIT SETIN CC.EXP THEN		
13400	044124	005737	000450			TST	SIGN0
13401	044130	001404				BEQ	\$51764
13402	044132	032737	000004	000522		BIT	#ZBIT,CC.EXP
13403	044140	001404				BEQ	\$51765
13404	044142					\$51764:	
13405	044142				LET @TEMPO :B= #53		
13406	044142	112777	000053	134326		MOVB	#53,@TEMPO
13407	044150				ELSE		
13408	044150	000403				BR	\$51766
13409	044152					\$51765:	
13410	044152				LET @TEMPO :B= #55		
13411	044152	112777	000055	134316		MOVB	#55,@TEMPO
13412	044160				ENDIF		
13413	044160					\$51766:	
13414	044160				IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGN0 EQ #-1 THEN		
13415	044160	123737	000566	000562		CMPB	DATA02,DATA00
13416	044166	103013				BHIS	\$51767
13417	044170	032737	000200	000566		BIT	#BIT07,DATA02
13418	044176	001007				BNE	\$51767
13419	044200	023727	000450	177777		CMP	SIGN0,#-1
13420	044206	001003				BNE	\$51767
13421	044210				LET @TEMPO :B- #55		
13422	044210	112777	000055	134260		MOVB	#55,@TEMPO
13423	044216				ENDIF		
13424	044216					\$51767:	
13425	044216				IF DATA01 EQ #0 THEN	:DST.LEN = 0	
13426	044216	005737	000564			TST	DATA01
13427	044222	001003				BNE	\$51770
13428	044224				LET CC.EXP := CC.EXP CLR.BY #VBIT		
13429	044224	042737	000002	000522		BIC	#VBIT,CC.EXP
13430	044232				ENDIF		
13431	044232					\$51770:	
13432	044232				ENDIF		
13433	044232					\$51761:	

13434 044232
 13435 044232 023727 000464 050000
 13436 044240 001075
 13437 044242
 13438 044242 012737 064442 000476
 13439 044250 063737 000564 000476
 13440 044256
 13441 044256 105077 134214
 13442 044262
 13443 044262 012737 064400 000476
 13444 044270 063737 000564 000476
 13445 044276
 13446 044276 105077 134174
 13447
 13448
 13449
 13450 044302
 13451 044302 005737 000450
 13452 044306 001004
 13453 044310
 13454 044310 112737 000053 064273
 13455 044316
 13456 044316 000403
 13457 044320
 13458 044320
 13459 044320 112737 000055 064273
 13460 044326
 13461 044326
 13462 044326
 13463 044326 005737 000450
 13464 044332 001404
 13465 044334 032737 000004 000522
 13466 044342 001404
 13467 044344
 13468 044344
 13469 044344 112737 000053 064441
 13470 044352
 13471 044352 000403
 13472 044354
 13473 044354
 13474 044354 112737 000055 064441
 13475 044362
 13476 044362
 13477 044362
 13478 044362 123737 000566 000562
 13479 044370 103013
 13480 044372 032737 000200 000566
 13481 044400 001007
 13482 044402 023727 000450 177777
 13483 044410 001003
 13484 044412
 13485 044412 112737 000055 064441
 13486 044420
 13487 044420
 13488 044420
 13489 044420 005737 000564

IF DATTYP EQ #50000 THEN
 LET TEMPO := #DS.EXP + DATA01
 LET @TEMPO :B= #0
 LET TEMPO := #DS.DST + DATA01
 LET @TEMPO :B= #0
 ;PUT SIGN INTO SOURCE
 IF SIGN0 EQ #0 THEN
 LET DSBUF1-1 :B= #53
 ELSE
 LET DSBUF1-1 :B= #55
 ENDIF
 IF SIGN0 EQ #0 OR #ZBIT SETIN CC.EXP THEN
 LET DS.EXP-1 :B= #53
 ELSE
 LET DS.EXP-1 :B= #55
 ENDIF
 IFB DATA02 LO DATA00 AND #BIT07 NOTSETIN DATA02 AND SIGN0 EQ #-1 THEN
 LET DS.EXP-1 :B= #55
 ENDIF
 IF DATA01 EQ #0 THEN

;*** LEADING SEPARATE ***
 CMP DATTYP,#50000
 BNE \$51771
 MOV #DS.EXP,TEMPO
 ADD DATA01,TEMPO
 ;RESTORE DS.EXP FROM TRAILING SEPARATE
 CLRB @TEMPO
 MOV #DS.DST,TEMPO
 ADD DATA01,TEMPO
 ;RESTORE DS.DST FROM TRAILING SEPARATE
 CLRB @TEMPO
 ;SIGN IS +
 TST SIGN0
 BNE \$51772
 MOVB #53,DSBUF1-1
 ;SIGN IS -
 BR \$51773
 \$51772:
 MOVB #55,DSBUF1-1
 \$51773:
 TST SIGN0
 BEQ \$51774
 BIT #ZBIT,CC.EXP
 BEQ \$51775
 \$51774:
 MOVB #53,DS.EXP-1
 BR \$51776
 \$51775:
 MOVB #55,DS.EXP-1
 \$51776:
 CMPB DATA02,DATA00
 BHIS \$51777
 BIT #BIT07,DATA02
 BNE \$51777
 CMP SIGN0,#-1
 BNE \$51777
 MOVB #55,DS.EXP-1
 \$51777:
 ;DST.LEN = 0
 TST DATA01

13490	044424	001003				BNE	\$52000
13491	044426				LET CC.EXP := CC.EXP CLR.BY #VBIT		
13492	044426	042737	000002	000522		BIC	#VBIT,CC.EXP
13493	044434				ENDIF		
13494	044434					\$52000:	
13495	044434				ENDIF		
13496	044434					\$51771:	
13497							
13498					:+		
13499					: SET UP THE GPR'S AND DO THE ASHN INSTRUCTION		
13500					:-		
13501							
13502	044434				LET R0 :- DATA00 SET.BY DATTYP		
13503	044434	013700	000562			MOV	DATA00,R0
13504	044440	053700	000464			BIS	DATTYP,R0
13505	044444				LET R1 :- #DSBUF1		
13506	044444	012701	064274			MOV	#DSBUF1,R1
13507	044450				LET R2 := DATA01 SET.BY DATTYP		
13508	044450	013702	000564			MOV	DATA01,R2
13509	044454	053702	000464			BIS	DATTYP,R2
13510	044460				LET R3 := #DS.DST		
13511	044460	012703	064400			MOV	#DS.DST,R3
13512	044464				LET R4 := DATA02		
13513	044464	013704	000566			MOV	DATA02,R4
13514	044470				LET R5 := #-1		
13515	044470	012705	177777			MOV	#-1,R5
13516	044474				INLINE <ASHN>		
13517	044474	076056				ASHN	
13518							
13519					:+		
13520					: CHECK THE RESULTS AND REPORT ERRORS		
13521					:-		
13522							
13523	044476				LET CC := PSW		
13524	044476	013737	177776	000520		MOV	PSW,CC
13525	044504				LET CC := CC CLR.BY #177760		
13526	044504	042737	177760	000520		BIC	#177760,CC
13527	044512				IF CC NE CC.EXP THEN		
13528	044512	023737	000520	000522		CMP	CC,CC.EXP
13529	044520	001402				BEQ	\$52001
13530	044522				CALL ERROR		
13531	044522	004737	060044			JSR	PC,ERROR
13532	044526				ENDIF		
13533	044526					\$52001:	
13534	044526				IF R0 NE #0 OR R1 NE #0 OR R4 NE #0 THEN		
13535	044526	005700				TST	R0
13536	044530	001004				BNE	\$52002
13537	044532	005701				TST	R1
13538	044534	001002				BNE	\$52002
13539	044536	005704				TST	R4
13540	044540	001402				BEQ	\$52003
13541	044542					\$52002:	
13542	044542				CALL ERROR		
13543	044542	004737	060044			JSR	PC,ERROR
13544	044546				ENDIF		
13545	044546					\$52003:	

13546	044546			LET TEMPO := DATA01 SET.BY DATTYP	
13547	044546	013737	000564		MOV DATA01,TEMPO
13548	044554	053737	000464		BIS DATTYP,TEMPO
13549	044562			IF R2 NE TEMPO THEN	
13550	044562	020237	000476		CMP R2,TEMPO
13551	044566	001402			BEQ \$52004
13552	044570			CALL ERROR	
13553	044570	004737	060044		JSR PC,ERROR
13554	044574			ENDIF	
13555	044574				\$52004:
13556	044574			IF R3 NE #DS.DST THEN	
13557	044574	020327	064400		CMP R3,#DS.DST
13558	044600	001402			BEQ \$52005
13559	044602			CALL ERROR	;DS.EXP AND DS.DST DO NOT MATCH
13560	044602	004737	060044		JSR PC,ERROR
13561	044606			ENDIF	
13562	044606				\$52005:
13563				;COMPARE DS.EXP WITH DS.DST	
13564	044606			LET ERRFLG := #0	;CLEAR ERROR FLAG
13565	044606	005037	000456		CLR ERRFLG
13566	044612			LET R0 := #41	
13567	044612	012700	000041		MOV #41,R0
13568	044616			LET R1 := #DS.DST-1	
13569	044616	012701	064377		MOV #DS.DST-1,R1
13570	044622			LET R2 := #DS.EXP-1	
13571	044622	012702	064441		MOV #DS.EXP-1,R2
13572	044626			WHILE R0 NE #0 AND ERRFLG EQ #0 DO	
13573	044626				\$52006:
13574	044626	005700			TST R0
13575	044630	001412			BEQ \$52007
13576	044632	005737	000456		TST ERRFLG
13577	044636	001007			BNE \$52007
13578	044640			IFB (R1)+ NE (R2)+ THEN	
13579	044640	122122			CMPB (R1)+,(R2)+
13580	044642	001403			BEQ \$52010
13581	044644			LET ERRFLG := #-1	
13582	044644	012737	177777	000456	MOV #-1,ERRFLG
13583	044652			ENDIF	
13584	044652				\$52010:
13585	044652			LET R0 := R0 - #1	
13586	044652	005300			DEC R0
13587	044654			ENDDO	
13588	044654	000764			BR \$52006
13589	044656				\$52007:
13590	044656			IF ERRFLG NE #0 THEN	
13591	044656	005737	000456		TST ERRFLG
13592	044662	001402			BEQ \$52011
13593	044664			CALL ERROR	
13594	044664	004737	060044		JSR PC,ERROR
13595	044670			ENDIF	
13596	044670				\$52011:
13597				;ENDINC	
13598	044670			INLINE <JMP \$5601>	
13599	044670	000137	043076		JMP \$5601
13600	044674			INLINE <\$5603:>	
13601	044674				\$5603:

13602						
13603	044674			INCR R0 FROM #DS.EXP-1 TO #DS.EXP+37 BY #1		
13604	044674	012700	064441		MOV #DS.EXP-1,R0	
13605	044700	000401			BR \$52012	
13606	044702				\$52013:	
13607	044702	005200			INC R0	
13608	044704				\$52012:	
13609	044704	020027	064501		CMP R0,#DS.EXP+37	
13610	044710	003002			BGT \$52014	
13611	044712			LET (R0) :B= #0	:CLEAR DS.DST	
13612	044712	105010			CLRB (R0)	
13613	044714			ENDINC		
13614	044714	000772			BR \$52013	
13615	044716				\$52014:	
13616	044716			LET SIGN0 := COMP SIGN0	:REPEAT THE TEST CASE WITH A NEGATIVE	
13617	044716	005137	000450		COM SIGN0	
13618					: SIGN BEFORE MOVING ON TO THE NEXT ONE	
13619	044722			IF SIGN0 EQ #0 THEN		
13620	044722	005737	000450		TST SIGN0	
13621	044726	001005			BNE \$52015	
13622	044730			LET TSTCAS := TSTCAS + #6		
13623	044730	062737	000006 000544		ADD #6,TSTCAS	
13624	044736			LET COUNT := COUNT - #1		
13625	044736	005337	000470		DEC COUNT	
13626	044742			ENDIF		
13627	044742				\$52015:	
13628				:UNTIL COUNT EQ #0		
13629	044742			IF COUNT NE #0 THEN		
13630	044742	005737	000470		TST COUNT	
13631	044746	001402			BEQ \$52016	
13632	044750			INLINE <JMP \$5056>		
13633	044750	000137	042232		JMP \$5056	
13634	044754			ENDIF		
13635	044754				\$52016:	
13636						
13637						
13638				:+		
13639				: DO ONE INLINE CASE - ASHNI		
13640				: -		
13641						
13642	044754			CALL CLRDSB		
13643	044754	004737	057040		JSR PC,CLRDSB	
13644				:LOAD SRC STRING INTO DSBUF1		
13645	044760			LET R0 := #DSBUF1-1		
13646	044760	012700	064273		MOV #DSBUF1-1,R0	
13647	044764			LET (R0)+ :B= #53	:LEADING SEPARATE SIGN OF +	
13648	044764	112720	000053		MOVB #53,(R0)+	
13649	044770			LET (R0)+ :B= #9.	:PUT SRC MAGNITUDE OF 9999	
13650	044770	112720	000011		MOVB #9.,(R0)+	
13651	044774			LET (R0)+ :B= #9.	: INTO DSBUF1	
13652	044774	112720	000011		MOVB #9.,(R0)+	
13653	045000			LET (R0)+ :B= #9.		
13654	045000	112720	000011		MOVB #9.,(R0)+	
13655	045004			LET (R0) :B= #9.		
13656	045004	112710	000011		MOVB #9.,(R0)	
13657	045010			LET R0 := #0		

13658	045010	005000				CLR	R0
13659	045012				LET R1 := #0		
13660	045012	005001				CLR	R1
13661	045014				LET R2 := #0		
13662	045014	005002				CLR	R2
13663	045016				LET R3 := #0		
13664	045016	005003				CLR	R3
13665	045020				LET R4 := #0		
13666	045020	005004				CLR	R4
13667	045022				LET R5 := #0		
13668	045022	005005				CLR	R5
13669	045024				INLINE <ASHNI>		
13670	045024	076156				ASHNI	
13671	045026				INLINE <.WORD TASHNI>		
13672	045026	045220				.WORD TASHNI	
13673	045030				INLINE <.WORD TASHNI+4>		
13674	045030	045224				.WORD TASHNI+4	
13675	045032				INLINE <.WORD 000002>		
13676	045032	000002				;SHIFT COUNT = 2, RND DIGIT = 0	
13677	045034				LET CC := PSW	.WORD 000002	
13678	045034	013737	177776	000520		MOV	PSW,CC
13679	045042				LET CC := CC CLR.BY #177760		
13680	045042	042737	177760	000520		BIC	#177760,CC
13681	045050				IF CC NE #6 THEN	;VBIT SHOULD BE SET	
13682	045050	023727	000520	000006		CMP	CC,#6
13683	045056	001402				BEQ	\$52017
13684	045060				CALL ERROR		
13685	045060	004737	060044			JSR	PC,ERROR
13686	045064				ENDIF		
13687	045064					\$52017:	
13688	045064				IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN		
13689	045064	005700				TST	R0
13690	045066	001004				BNE	\$52020
13691	045070	005701				TST	R1
13692	045072	001002				BNE	\$52020
13693	045074	005702				TST	R2
13694	045076	001402				BEQ	\$52021
13695	045100					\$52020:	
13696	045100				CALL ERROR		
13697	045100	004737	060044			JSR	PC,ERROR
13698	045104				ENDIF		
13699	045104					\$52021:	
13700	045104				IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN		
13701	045104	005703				TST	R3
13702	045106	001004				BNE	\$52022
13703	045110	005704				TST	R4
13704	045112	001002				BNE	\$52022
13705	045114	005705				TST	R5
13706	045116	001402				BEQ	\$52023
13707	045120					\$52022:	
13708	045120				CALL ERROR		
13709	045120	004737	060044			JSR	PC,ERROR
13710	045124				ENDIF		
13711	045124					\$52023:	
13712					;CHECK DS.DST		
13713	045124				LET R0 := #DS.DST		

13714	045124	012700	064400			MOV	#DS.DST,RO
13715	045130			IFB (RO)+ NE #173 THEN			
13716	045130	122027	000173			CMPB	(RO)+,#173
13717	045134	001402				BEQ	\$52024
13718	045136			CALL ERROR			
13719	045136	004737	060044			JSR	PC,ERROR
13720	045142			ENDIF			
13721	045142					\$52024:	
13722	045142			IFB (RO)+ NE #60 THEN			
13723	045142	122027	000060			CMPB	(RO)+,#60
13724	045146	001402				BEQ	\$52025
13725	045150			CALL ERROR			
13726	045150	004737	060044			JSR	PC,ERROR
13727	045154			ENDIF			
13728	045154					\$52025:	
13729	045154			LET ERRFLG := #0			
13730	045154	005037	000456			CLR	ERRFLG
13731	045160			WHILE ERRFLG EQ #0 AND RO NE #DS.DST+37 DO			
13732	045160					\$52026:	
13733	045160	005737	000456			TST	ERRFLG
13734	045164	001013				BNE	\$52027
13735	045166	020027	064437			CMP	RO,#DS.DST+37
13736	045172	001410				BEQ	\$52027
13737	045174			IFB (RO)+ NE #0 THEN			
13738	045174	105720				TSTB	(RO)+
13739	045176	001405				BEQ	\$52030
13740	045200			CALL ERROR			
13741	045200	004737	060044			JSR	PC,ERROR
13742	045204			LET ERRFLG := #-1			
13743	045204	012737	177777 000456			MOV	#-1,ERRFLG
13744	045212			ENDIF			
13745	045212					\$52030:	
13746	045212			ENDDO			
13747	045212	000762				BR	\$52026
13748	045214					\$52027:	
13749	045214			INLINE <JMP CIS6END>			
13750	045214	000137	045230			JMP	CIS6END
13751							
13752							
13753				::+ :: INLINE TEST CASE ::-			
13754							
13755							
13756							
13757	045220	050004		TASHNI: .WORD 050004		:SRC.DSCR, LEADING SEPARATE, SRC.LEN = 4	
13758	045222	064274		.WORD DSBUF1		:SRC.DSCR	
13759	045224	030002		.WORD 030002		:DST.DSCR, LEADING OVERPUNCH, DST.LEN = 2	
13760	045226	064400		.WORD DS.DST		:DST.DSCR	
13761							
13762							
13763							
13764	045230			INLINE <CIS6END:>			
13765	045230					CIS6END:	
13766							
13767							

13768
 13769
 13770
 13771
 13772
 13773
 13774
 13775
 13776
 13777
 13778
 13779
 13780
 13781
 13782
 13783
 13784
 13785
 13786
 13787
 13788
 13789
 13790
 13791
 13792
 13793
 13794
 13795
 13796
 13797
 13798
 13799
 13800
 13801
 13802
 13803
 13804
 13805
 13806
 13807
 13808
 13809
 13810
 13811
 13812
 13813
 13814
 13815
 13816
 13817
 13818

045230
 045230 000004
 045232
 045232 017737 133170 000476
 045240 042737 177700 000476
 045246
 045246 005737 000476
 045252 001406
 045254
 045254 032737 000010 000476
 045262 001002
 045264
 045264 000137 053006
 045270
 045270
 045270
 045270
 045270
 045270 012737 000007 000546
 045276
 045276 005037 000456
 045302
 045302 012737 000340 177776
 045310
 045310 012737 000001 000552

```

*****
*****
**
**          C I S 7      M O D U L E
**
** THIS MODULE TESTS THE PACKED DATA TYPE ARITHMETIC
** INSTRUCTIONS: ADDP, SUBP, CMPP, ASHP.
**
*****
*****

.SBTTL CIS7 INITIALIZATION
*****
:
: CIS7 INITIALIZATION
*****
:
: IF SWR<5:0> ARE CLEARED OR IF SWR<3> IS SET THEN EXECUTE
: THE CIS7 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.

INLINE <SCOPE>
                                SCOPE
LET TEMPO := @SWR CLR.BY #177700
                                MOV @SWR,TEMPO
                                BIC #177700,TEMPO
IF TEMPO NE #0 THEN
                                TST TEMPO
                                BEQ $52031
    IF #BIT3 NOTSETIN TEMPO THEN
                                BIT #BIT3,TEMPO
                                BNE $52032
        INLINE <JMP CIS7END>
                                JMP CIS7END
    ENDIF
ENDIF
                                $52032:
ENDIF
                                $52031:
LET CTL := #7 ;CONTROL CHIP NUMBER
                                MOV #7,CTL
LET ERRFLG := #0 ;INITIALIZE ERROR FLAG
                                CLR ERRFLG
LET PSW := #340
                                MOV #340,PSW
LET ERR := #1
                                MOV #1,ERR

```

```
13819 .SBTTL ADDP
13820 :*****
13821 :
13822 :           ADDP TEST
13823 :
13824 :*****
13825
13826 045316      INLINE <XADDP:>
13827 045316      XADDP:
13828 045316      INLINE <SCOPE>
13829 045316 000004      SCOPE
13830 045320      LET $TIMES := #20      ;ITERATION COUNT
13831 045320 012737 000020 000404      MOV      #20,$TIMES
13832 045326      LET COUNT := TADD      ;# OF TEST CASES
13833 045326 013737 065104 000470      MOV      TADD,COUNT
13834 045334      LET TSTCAS :- #TADD+2      ;POINT TO FIRST TEST CASE
13835 045334 012737 065106 000544      MOV      #TADD+2,TSTCAS
13836
13837 :+
13838 :           CLR R0.EXP TO R3.EXP
13839 :-
13840
13841 045342      INCR R0 FROM #0 TO #6 BY #2
13842 045342 005000      CLR      R0
13843 045344 000402      BR      $52033
13844 045346      $52034:
13845 045346 062700 000002      ADD      #2,R0
13846 045352      $52033:
13847 045352 020027 000006      CMP      R0,#6
13848 045356 003003      BGT      $52035
13849 045360      INLINE <CLR R0.EXP(R0)>
13850 045360 005060 000524      CLR R0.EXP(R0)
13851 045364      ENDINC
13852 045364 000770      BR      $52034
13853 045366      $52035:
13854      ;REPEAT OUTER LOOP FOR TEST CASES
13855 045366      INLINE <$5070:>
13856 045366      $5070:
13857 045366      CALL CLRDSB      ;CLR DS BUFFERS
13858 045366 004737 057040      JSR      PC,CLRDSB
13859
13860 :+
13861 :           LOAD DATA0 AND CC.EXP FROM TEST CASE
13862 :-
13863
13864 045372      LET R0 := TSTCAS
13865 045372 013700 000544      MOV      TSTCAS,R0
13866 045376      LET DATA00 := (R0)+      ;SRC1 LENGTH
13867 045376 012037 000562      MOV      (R0)+,DATA00
13868 045402      LET DATA01 := (R0)+      ;POINTER TO SRC1 DATA
13869 045402 012037 000564      MOV      (R0)+,DATA01
13870 045406      LET DATA02 := (R0)+      ;SRC2 LENGTH
13871 045406 012037 000566      MOV      (R0)+,DATA02
13872 045412      LET DATA03 := (R0)+      ;POINTER TO SRC2 DATA
13873 045412 012037 000570      MOV      (R0)+,DATA03
13874 045416      LET DATA04 :- (R0)+      ;DEST LENGTH
```

```
13875 045416 012037 000572          MOV      (R0)+,DATA04
13876 045422          LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (SUM)
13877 045422 012037 000574          MOV      (R0)+,DATA05
13878 045426          LET CC.EXP := (R0)+          ;CC.EXP (SUM)
13879 045426 012037 000522          MOV      (R0)+,CC.EXP
13880
13881          ;+
13882          ;   PACK BUFFERS FOR SRC1, SRC2, & DEST
13883          ;-
13884
13885 045432          LET R5 := #CHRSET          ;R5 IS OPERAND STACK FOR ROUTINE
13886 045432 012705 063672          MOV      #CHRSET,R5
13887 045436          CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
13888 045436 010546          MOV      R5,-(SP)
13889 045440 010605          MOV      SP,R5
13890 045442 162706 000006          SUB      #6,SP
13891 045446 013745 000562          MOV      DATA00,-(R5)
13892 045452 013745 000564          MOV      DATA01,-(R5)
13893 045456 012745 064274          MOV      #DSBUF1,-(R5)
13894 045462 004737 057420          JSR      PC,PACK
13895 045466 062706 000006          ADD      #6,SP
13896 045472 012605          MOV      (SP)+,R5
13897 045474          CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
13898 045474 010546          MOV      R5,-(SP)
13899 045476 010605          MOV      SP,R5
13900 045500 162706 000006          SUB      #6,SP
13901 045504 013745 000566          MOV      DATA02,-(R5)
13902 045510 013745 000570          MOV      DATA03,-(R5)
13903 045514 012745 064336          MOV      #DSBUF2,-(R5)
13904 045520 004737 057420          JSR      PC,PACK
13905 045524 062706 000006          ADD      #6,SP
13906 045530 012605          MOV      (SP)+,R5
13907 045532          CALL PACK IN <#DS.EXP,DATA05,DATA04> ;DEST
13908 045532 010546          MOV      R5,-(SP)
13909 045534 010605          MOV      SP,R5
13910 045536 162706 000006          SUB      #6,SP
13911 045542 013745 000572          MOV      DATA04,-(R5)
13912 045546 013745 000574          MOV      DATA05,-(R5)
13913 045552 012745 064442          MOV      #DS.EXP,-(R5)
13914 045556 004737 057420          JSR      PC,PACK
13915 045562 062706 000006          ADD      #6,SP
13916 045566 012605          MOV      (SP)+,R5
13917
13918          ;+
13919          ;   LOAD GPR'S FROM DATAXX & ADD DATA TYPE
13920          ;-
13921
13922 045570          LET R0 := DATA00 SET.BY #60000
13923 045570 013700 000562          MOV      DATA00,R0
13924 045574 052700 060000          BIS      #60000,R0
13925 045600          LET R1 := #DSBUF1
13926 045600 012701 064274          MOV      #DSBUF1,R1
13927 045604          LET R2 := DATA02 SET.BY #60000
13928 045604 013702 000566          MOV      DATA02,R2
13929 045610 052702 060000          BIS      #60000,R2
13930 045614          LET R3 := #DSBUF2
```

13931	045614	012703	064336		MOV	#DSBUF2,R3
13932	045620			LET R4 := DATA04 SET.BY #60000		
13933	045620	013704	000572		MOV	DATA04,R4
13934	045624	052704	060000		BIS	#60000,R4
13935	045630			LET R5 := #DS.DST		
13936	045630	012705	064400		MOV	#DS.DST,R5
13937						
13938				:+		
13939				:-	FINISH SET UP & DO CIS INSTRUCTION	
13940						
13941						
13942						
13943	045634			LET R4.EXP := R4		
13944	045634	010437	000534		MOV	R4,R4.EXP
13945	045640			LET R5.EXP := R5		
13946	045640	010537	000536		MOV	R5,R5.EXP
13947	045644			INLINE <ADDP>		
13948	045644	076070			ADDP	
13949						
13950				:+		
13951				:-	CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.	
13952						
13953						
13954	045646			LET CC := PSW		
13955	045646	013737	177776	000520	MOV	PSW,CC
13956	045654			LET CL := CC CLR.BY #177760		
13957	045654	042737	177760	000520	BIC	#177760,CC
13958	045662			IF CC NE CC.EXP THEN		
13959	045662	023737	000520	000522	CMP	CC,CC.EXP
13960	045670	001402			BEQ	\$52036
13961	045672			CALL ERROR		
13962	045672	004737	060044		JSR	PC,ERROR
13963	045676			ENDIF		
13964	045676				\$52036:	
13965	045676			IF R0 NE R0.EXP THEN		
13966	045676	020037	000524		CMP	R0,R0.EXP
13967	045702	001402			BEQ	\$52037
13968	045704			CALL ERROR		
13969	045704	004737	060044		JSR	PC,ERROR
13970	045710			ENDIF		
13971	045710				\$52037:	
13972	045710			IF R1 NE R1.EXP THEN		
13973	045710	020137	000526		CMP	R1,R1.EXP
13974	045714	001402			BEQ	\$52040
13975	045716			CALL ERROR		
13976	045716	004737	060044		JSR	PC,ERROR
13977	045722			ENDIF		
13978	045722				\$52040:	
13979	045722			IF R2 NE R2.EXP THEN		
13980	045722	020237	000530		CMP	R2,R2.EXP
13981	045726	001402			BEQ	\$52041
13982	045730			CALL ERROR		
13983	045730	004737	060044		JSR	PC,ERROR
13984	045734			ENDIF		
13985	045734				\$52041:	
13986	045734			IF R3 NE R3.EXP THEN		

13987	045734	020337	000532						
13988	045740	001402							CMP R3,R3.EXP
13989	045742			CALL ERROR					BEQ \$52042
13990	045742	004737	060044						JSR PC,ERROR
13991	045746			ENDIF					
13992	045746								\$52042:
13993	045746			IF R4 NE R4.EXP THEN					
13994	045746	020437	000534						CMP R4,R4.EXP
13995	045752	001402							BEQ \$52043
13996	045754			CALL ERROR					
13997	045754	004737	060044						JSR PC,ERROR
13998	045760			ENDIF					
13999	045760								\$52043:
14000	045760			IF R5 NE R5.EXP THEN					
14001	045760	020537	000536						CMP R5,R5.EXP
14002	045764	001402							BEQ \$52044
14003	045766			CALL ERROR					
14004	045766	004737	060044						JSR PC,ERROR
14005	045772			ENDIF					
14006	045772								\$52044:
14007									
14008				::+					
14009				::	CHECK RESULTS VS EXPECTED RESULTS				
14010				::-					
14011									
14012	045772			LET RU := #DS.DST					
14013	045772	012700	064400						MOV #DS.DST,R0
14014	045776			LET R1 := #DS.EXP					
14015	045776	012701	064442						MOV #DS.EXP,R1
14016	046002			LET ERRFLG := #0					
14017	046002	005037	000456						CLR ERRFLG
14018	046006			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO					
14019	046006								\$52045:
14020	046006	020027	064440						CMP R0,#DS.DST+40
14021	046012	001410							BEQ \$52046
14022	046014	005737	000456						TST ERRFLG
14023	046020	001005							BNE \$52046
14024	046022			IFB (R0)+ NE (R1)+ THEN					
14025	046022	122021							CMPB (R0)+,(R1)+
14026	046024	001402							BEQ \$52047
14027	046026			CALL ERROR					
14028	046026	004737	060044						JSR PC,ERROR
14029	046032			ENDIF					
14030	046032								\$52047:
14031	046032			ENDDO					
14032	046032	000765							BR \$52045
14033	046034								\$52046:
14034	046034			LET TSTCAS := TSTCAS + #22					
14035	046034	062737	000022 000544						ADD #22,TSTCAS
14036	046042			LET COUNT := COUNT - #1					
14037	046042	005337	000470						DEC COUNT
14038				:UNTIL COUNT EQ #0					
14039	046046			IF COUNT NE #0 THEN					
14040	046046	005737	000470						TST COUNT
14041	046052	001402							BEQ \$52050
14042	046054			INLINE <JMP \$5070>					

```

14043 046054 000137 045366                                JMP $5070
14044 046060                                ENDIF
14045 046060                                $52050:
14046
14047
14048      :+      INLINE CASE - ADDPI, USING UNSIGNED PACKED DATA
14049      :-
14050
14051 046060                                CALL CLRDSB
14052 046060 004737 057040                                JSR      PC,CLRDSB
14053
14054      :+
14055      :-      LOAD DATAXX AND CC.EXP FROM TEST CASE
14056
14057
14058 046064                                LET R0 := #TADD+2
14059 046064 012700 065106                                MOV      #TADD+2,R0
14060 046070                                LET DATA00 := (R0)+      ;SRC1 LENGTH
14061 046070 012037 000562                                MOV      (R0)+,DATA00
14062 046074                                LET DATA01 := (R0)+      ;POINTER TO SRC1 DATA
14063 046074 012037 000564                                MOV      (R0)+,DATA01
14064 046100                                LET DATA02 := (R0)+      ;SRC2 LENGTH
14065 046100 012037 000566                                MOV      (R0)+,DATA02
14066 046104                                LET DATA03 := (R0)+      ;POINTER TO SRC2 DATA
14067 046104 012037 000570                                MOV      (R0)+,DATA03
14068 046110                                LET DATA04 := (R0)+      ;DEST LENGTH
14069 046110 012037 000572                                MOV      (R0)+,DATA04
14070 046114                                LET DATA05 := (R0)+      ;POINTER TO DS.EXP DATA (SUM)
14071 046114 012037 000574                                MOV      (R0)+,DATA05
14072
14073      :+
14074      :-      PACK BUFFERS FOR SRC1, SRC2, & DEST
14075
14076
14077 046120                                LET R5 := #CHRSET      ;R5 IS OPERAND STACK FOR ROUTINE
14078 046120 012705 063672                                MOV      #CHRSET,R5
14079 046124                                CALL PACK IN <#DSBUF1,DATA01,DATA00> ;SRC1
14080 046124 010546                                MOV      R5,-(SP)
14081 046126 010605                                MOV      SP,R5
14082 046130 162706 000006                                SUB      #6,SP
14083 046134 013745 000562                                MOV      DATA00,-(R5)
14084 046140 013745 000564                                MOV      DATA01,-(R5)
14085 046144 012745 064274                                MOV      #DSBUF1,-(R5)
14086 046150 004737 057420                                JSR      PC,PACK
14087 046154 062706 000006                                ADD      #6,SP
14088 046160 012605                                MOV      (SP)+,R5
14089 046162                                CALL PACK IN <#DSBUF2,DATA03,DATA02> ;SRC2
14090 046162 010546                                MOV      R5,-(SP)
14091 046164 010605                                MOV      SP,R5
14092 046166 162706 000006                                SUB      #6,SP
14093 046172 013745 000566                                MOV      DATA02,-(R5)
14094 046176 013745 000570                                MOV      DATA03,-(R5)
14095 046202 012745 064336                                MOV      #DSBUF2,-(R5)
14096 046206 004737 057420                                JSR      PC,PACK
14097 046212 062706 000006                                ADD      #6,SP
14098 046216 012605                                MOV      (SP)+,R5

```

14099	046220			CALL PACK IN <#DS.EXP,DATA05,DATA04>	:DEST		
14100	046220	010546				MOV	R5,-(SP)
14101	046222	010605				MOV	SP,R5
14102	046224	162706	000006			SUB	#6,SP
14103	046230	013745	000572			MOV	DATA04,-(R5)
14104	046234	013745	000574			MOV	DATA05,-(R5)
14105	046240	012745	064442			MOV	#DS.EXP,-(R5)
14106	046244	004737	057420			JSR	PC,PACK
14107	046250	062706	000006			ADD	#6,SP
14108	046254	012605				MOV	(SP)+,R5
14109							
14110	046256			CALL CLRREG	:ENSURE GPR'S 0-5 ARE ZERO		
14111	046256	004737	056604			JSR	PC,CLRREG
14112							
14113							
14114	046262			INLINE <ADDP>			
14115	046262	076170				ADDP	
14116	046264			INLINE <.WORD TADDP>			
14117	046264	065772				.WORD	TADDP
14118	046266			INLINE <.WORD TADDP+4>			
14119	046266	065776				.WORD	TADDP+4
14120	046270			INLINE <.WORD TADDP+10>			
14121	046270	066002				.WORD	TADDP+10
14122	046272			LET CC := PSW			
14123	046272	013737	177776	000520		MOV	PSW,CC
14124	046300			LET CC := CC CLR.BY #177760			
14125	046300	042737	177760	000520		BIC	#177760,CC
14126	046306			IF CC NE #0 THEN			
14127	046306	005737	000520			TST	CC
14128	046312	001402				BEQ	\$52051
14129	046314			CALL ERROR			
14130	046314	004737	060044			JSR	PC,ERROR
14131	046320			ENDIF			
14132	046320						\$52051:
14133							
14134				:+			
14135				: ENSURE GPR'S STILL ZERO			
14136				:-			
14137							
14138	046320			IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN			
14139	046320	005700				TST	R0
14140	046322	001004				BNE	\$52052
14141	046324	005701				TST	R1
14142	046326	001002				BNE	\$52052
14143	046330	005702				TST	R2
14144	046332	001402				BEQ	\$52053
14145	046334						\$52052:
14146	046334			CALL ERROR			
14147	046334	004737	060044			JSR	PC,ERROR
14148	046340			ENDIF			
14149	046340						\$52053:
14150	046340			IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN			
14151	046340	005703				TST	R3
14152	046342	001004				BNE	\$52054
14153	046344	005704				TST	R4
14154	046346	001002				BNE	\$52054

14155	046350	005705				TST	R5
14156	046352	001402				BEQ	\$52055
14157	046354					\$52054:	
14158	046354			CALL ERROR			
14159	046354	004737	060044			JSR	PC,ERROR
14160	046360			ENDIF			
14161	046360					\$52055:	
14162							
14163				:+			
14164				:+ CHECK RESULTS			
14165				:+			
14166				:-			
14167	046360			LET R0 := #DS.DST			
14168	046360	012700	064400			MOV	#DS.DST,R0
14169	046364			LET R1 := #DS.EXP	:EXPECTED DATA		
14170	046364	012701	064442			MOV	#DS.EXP,R1
14171	046370			LET R2 := TADDPI+10 CLR.BY #70000;GET STRING LENGTH			
14172	046370	013702	066002			MOV	TADDPI+10,R2
14173	046374	042702	070000			BIC	#70000,R2
14174	046400			INLINE <ASR R2>	:LENGTH/2 (PACKING FACTOR)		
14175	046400	006202				ASR	R2
14176	046402			LET R2 := R2 + #DS.EXP	:R2 POINTS TO MSD/SIGN		
14177	046402	062702	064442			ADD	#DS.EXP,R2
14178	046406			LET (R2) :B= (R2) SET.BY #17	:MAKE IT UNSIGNED		
14179	046406	152712	000017			BISB	#17,(R2)
14180	046412			LET ERRFLG := #0			
14181	046412	005037	000456			CLR	ERRFLG
14182	046416			WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO			
14183	046416					\$52056:	
14184	046416	020027	064437			CMP	R0,#DS.DST+37
14185	046422	001410				BEQ	\$52057
14186	046424	005737	000456			TST	ERRFLG
14187	046430	001005				BNE	\$52057
14188	046432			IFB (R0)+ NE (R1)+ THEN			
14189	046432	122021				CMPB	(R0)+,(R1)+
14190	046434	001402				BEQ	\$52060
14191	046436			CALL ERROR			
14192	046436	004737	060044			JSR	PC,ERROR
14193	046442			ENDIF			
14194	046442					\$52060:	
14195	046442			ENDDO			
14196	046442	000765				BR	\$52056
14197	046444					\$52057:	
14198							
14199							

```
14200 .SBTTL SUBP
14201 :*****
14202 :
14203 :       SUBP TEST
14204 :
14205 :*****
14206
14207 046444      INLINE <XSUBP:>
14208 046444      XSUBP:
14209 046444      INLINE <SCOPE>
14210 046444      SCOPE
14211 046446      LET $TIMES := #20           ; ITERATION COUNT
14212 046446 012737 000020 000404      MOV      #20,$TIMES
14213 046454      LET COUNT := TADD           ; # OF TEST CASES
14214 046454 013737 065104 000470      MOV      TADD,COUNT
14215 046462      LET TSTCAS :- #TADD+2      ; POINT TO FIRST TEST CASE
14216 046462 012737 065106 000544      MOV      #TADD+2,TSTCAS
14217
14218 :+
14219 :       CLR R0.EXT TO R3.EXP
14220 :-
14221
14222 046470      INCR R0 FROM #0 TO #6 BY #2
14223 046470 005000      CLR      R0
14224 046472 000402      BR       $52061
14225 046474      $52062:
14226 046474 062700 000002      ADD      #2,R0
14227 046500      $52061:
14228 046500 020027 000006      CMP      R0,#6
14229 046504 003003      BGT      $52063
14230 046506      INLINE <CLR R0.EXP(R0)>
14231 046506 005060 000524      CLR R0.EXP(R0)
14232 046512      ENDINC
14233 046512 000770      BR       $52062
14234 046514      $52063:
14235      ; REPEAT OUTER LOOP FOR TEST CASES
14236 046514      INLINE <$5071:>
14237 046514      $5071:
14238 046514      CALL CLRDSB           ; CLR DS BUFFERS
14239 046514 004737 057040      JSR      PC,CLRDSB
14240
14241 :+
14242 :       LOAD DATAXX AND CC.EXP FROM TEST CASE
14243 :-
14244
14245 046520      LET R0 := TSTCAS
14246 046520 013700 000544      MOV      TSTCAS,R0
14247 046524      LET DATA00 := (R0)+           ; SRC1 LENGTH
14248 046524 012037 000562      MOV      (R0)+,DATA00
14249 046530      LET DATA01 := (R0)+           ; POINTER TO SRC1 DATA
14250 046530 012037 000564      MOV      (R0)+,DATA01
14251 046534      LET DATA02 := (R0)+           ; SRC2 LENGTH
14252 046534 012037 000566      MOV      (R0)+,DATA02
14253 046540      LET DATA03 :- (R0)+           ; POINTER TO SRC2 DATA
14254 046540 012037 000570      MOV      (R0)+,DATA03
14255 046544      LET DATA04 := (R0)+           ; DEST LENGTH
```

```

14256 046544 012037 000572
14257 046550
14258 046550 012037 000574
14259 046554
14260 046554 012037 000522
14261 046560
14262 046560 012037 000574
14263 046564
14264 046564 012037 000522
14265
14266
14267
14268
14269
14270 046570
14271 046570 012705 063672
14272 046574
14273 046574 010546
14274 046576 010605
14275 046600 162706 000006
14276 046604 013745 000562
14277 046610 013745 000564
14278 046614 012745 064274
14279 046620 004737 057420
14280 046624 062706 000006
14281 046630 012605
14282 046632
14283 046632 010546
14284 046634 010605
14285 046636 162706 000006
14286 046642 013745 000566
14287 046646 013745 000570
14288 046652 012745 064336
14289 046656 004737 057420
14290 046662 062706 000006
14291 046666 012605
14292 046670
14293 046670 010546
14294 046672 010605
14295 046674 162706 000006
14296 046700 013745 000572
14297 046704 013745 000574
14298 046710 012745 064442
14299 046714 004737 057420
14300 046720 062706 000006
14301 046724 012605
14302
14303
14304
14305
14306
14307
14308 046726
14309 046726 013700 000562
14310 046732 052700 060000
14311 046736

;+
;:
;:-
PACK BUFFERS FOR SRC1, SRC2, & DEST

LET R5 := #CHRSET ;R5 IS OPERAND STACK FOR ROUTINE
CALL PACK IN <#DSBUF1, DATA01, DATA00> ;SRC1
MOV #CHRSET, R5
MOV R5, -(SP)
MOV SP, R5
SUB #6, SP
MOV DATA00, -(R5)
MOV DATA01, -(R5)
MOV #DSBUF1, -(R5)
JSR PC, PACK
ADD #6, SP
MOV (SP)+, R5

CALL PACK IN <#DSBUF2, DATA03, DATA02> ;SRC2
MOV R5, -(SP)
MOV SP, R5
SUB #6, SP
MOV DATA02, -(R5)
MOV DATA03, -(R5)
MOV #DSBUF2, -(R5)
JSR PC, PACK
ADD #6, SP
MOV (SP)+, R5

CALL PACK IN <#DS.EXP, DATA05, DATA04> ;DEST
MOV R5, -(SP)
MOV SP, R5
SUB #6, SP
MOV DATA04, -(R5)
MOV DATA05, -(R5)
MOV #DS.EXP, -(R5)
JSR PC, PACK
ADD #6, SP
MOV (SP)+, R5

;+
;:
;:-
LOAD GPR'S FROM DATAXX & ADD DATA TYPE

LET R0 := DATA00 SET.BY #60000
MOV DATA00, R0
BIS #60000, R0

LET R1 := #DSBUF1
    
```

14312	046736	012701	064274		MOV	#DSBUF1,R1
14313	046742			LET R2 := DATA02 SET.BY #60000		
14314	046742	013702	000566		MOV	DATA02,R2
14315	046746	052702	060000		BIS	#60000,R2
14316	046752			LET R3 := #DSBUF2		
14317	046752	012703	064336		MOV	#DSBUF2,R3
14318	046756			LET R4 := DATA04 SET.BY #60000		
14319	046756	013704	000572		MOV	DATA04,R4
14320	046762	052704	060000		BIS	#60000,R4
14321	046766			LET R5 := #DS.DST		
14322	046766	012705	064400		MOV	#DS.DST,R5
14323						
14324				:+		
14325				:+ FINISH SET UP & DO CIS INSTRUCTION		
14326				:-		
14327						
14328						
14329	046772			LET R4.EXP := R4		
14330	046772	010437	000534		MOV	R4,R4.EXP
14331	046776			LET R5.EXP := R5		
14332	046776	010537	000536		MOV	R5,R5.EXP
14333	047002			INLINE <SUBP>		
14334	047002	076071			SUBP	
14335						
14336				:+		
14337				:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
14338				:-		
14339						
14340	047004			LET CC := PSW		
14341	047004	013737	177776	000520	MOV	PSW,CC
14342	047012			LET CC := CC CLR.BY #177760		
14343	047012	042737	177760	000520	BIC	#177760,CC
14344	047020			IF CC NE CC.EXP THEN		
14345	047020	023737	000520	000522	CMP	CC,CC.EXP
14346	047026	001402			BEQ	\$52064
14347	047030			CALL ERROR		
14348	047030	004737	060044		JSR	PC,ERROR
14349	047034			ENDIF		
14350	047034					\$52064:
14351	047034			IF R0 NE R0.EXP THEN		
14352	047034	020037	000524		CMP	R0,R0.EXP
14353	047040	001402			BEQ	\$52065
14354	047042			CALL ERROR		
14355	047042	004737	060044		JSR	PC,ERROR
14356	047046			ENDIF		
14357	047046					\$52065:
14358	047046			IF R1 NE R1.EXP THEN		
14359	047046	020137	000526		CMP	R1,R1.EXP
14360	047052	001402			BEQ	\$52066
14361	047054			CALL ERROR		
14362	047054	004737	060044		JSR	PC,ERROR
14363	047060			ENDIF		
14364	047060					\$52066:
14365	047060			IF R2 NE R2.EXP THEN		
14366	047060	020237	000530		CMP	R2,R2.EXP
14367	047064	001402			BEQ	\$52067

14368	047066			CALL ERROR		
14369	047066	004737	060044		JSR	PC,ERROR
14370	047072			ENDIF		
14371	047072				\$52067:	
14372	047072			IF R3 NE R3.EXP THEN		
14373	047072	020337	000532		CMP	R3,R3.EXP
14374	047076	001402			BEQ	\$52070
14375	047100			CALL ERROR		
14376	047100	004737	060044		JSR	PC,ERROR
14377	047104			ENDIF		
14378	047104				\$52070:	
14379	047104			IF R4 NE R4.EXP THEN		
14380	047104	020437	000534		CMP	R4,R4.EXP
14381	047110	001402			BEQ	\$52071
14382	047112			CALL ERROR		
14383	047112	004737	060044		JSR	PC,ERROR
14384	047116			ENDIF		
14385	047116				\$52071:	
14386	047116			IF R5 NE R5.EXP THEN		
14387	047116	020537	000536		CMP	R5,R5.EXP
14388	047122	001402			BEQ	\$52072
14389	047124			CALL ERROR		
14390	047124	004737	060044		JSR	PC,ERROR
14391	047130			ENDIF		
14392	047130				\$52072:	
14393						
14394						
14395				:+		
14396				:-	CHECK RESULTS VS EXPECTED RESULTS	
14397						
14398						
14399	047130			LET R0 := #DS.DST		
14400	047130	012700	064400		MOV	#DS.DST,R0
14401	047134			LET R1 := #DS.EXP		
14402	047134	012701	064442		MOV	#DS.EXP,R1
14403	047140			LET ERRFLG := #0		
14404	047140	005037	000456		CLR	ERRFLG
14405	047144			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO		
14406	047144				\$52073:	
14407	047144	020027	064440		CMP	R0,#DS.DST+40
14408	047150	001410			BEQ	\$52074
14409	047152	005737	000456		TST	ERRFLG
14410	047156	001005			BNE	\$52074
14411	047160			IFB (R0)+ NE (R1)+ THEN		
14412	047160	122021			CMPB	(R0)+,(R1)+
14413	047162	001402			BEQ	\$52075
14414	047164			CALL ERROR		
14415	047164	004737	060044		JSR	PC,ERROR
14416	047170			ENDIF		
14417	047170				\$52075:	
14418	047170			ENDDO		
14419	047170	000765			BR	\$52073
14420	047172				\$52074:	
14421	047172			LET TSTCAS := TSTCAS + #22		
14422	047172	062737	000022 000544		ADD	#22,TSTCAS
14423	047200			LET COUNT := COUNT - #1		

14480	047330			CALL PACK IN <#DSBUF2,DATA03,DATA02>	;SRC2		
14481	047330	010546				MOV	R5,-(SP)
14482	047332	010605				MOV	SP,R5
14483	047334	162706	000006			SUB	#6,SP
14484	047340	013745	000566			MOV	DATA02,-(R5)
14485	047344	013745	000570			MOV	DATA03,-(R5)
14486	047350	012745	064336			MOV	#DSBUF2,-(R5)
14487	047354	004737	057420			JSR	PC,PACK
14488	047360	062706	000006			ADD	#6,SP
14489	047364	012605				MOV	(SP)+,R5
14490	047366			CALL PACK IN <#DS.EXP,DATA05,DATA04>	;DEST		
14491	047366	010546				MOV	R5,-(SP)
14492	0-7370	010605				MOV	SP,R5
14493	047372	162706	000006			SUB	#6,SP
14494	047376	013745	000572			MOV	DATA04,-(R5)
14495	047402	013745	000574			MOV	DATA05,-(R5)
14496	047406	012745	064442			MOV	#DS.EXP,-(R5)
14497	047412	004737	057420			JSR	PC,PACK
14498	047416	062706	000006			ADD	#6,SP
14499	047422	012605				MOV	(SP)+,R5
14500	047424			CALL CLRREG	;ENSURE GPR'S 0-5 ARE ZERO		
14501	047424	004737	056604			JSR	PC,CLRREG
14502	047430			INLINE <SUBPI>			
14503	047430	076171				SUBPI	
14504	047432			INLINE <.WORD TADDPI>			
14505	047432	065772				.WORD	TADDPI
14506	047434			INLINE <.WORD TADDPI+4>			
14507	047434	065776				.WORD	TADDPI+4
14508	047436			INLINE <.WORD TADDPI+10>			
14509	047436	066002				.WORD	TADDPI+10
14510	047440			LET CC := PSW			
14511	047440	013737	177776	000520		MOV	PSW,CC
14512	047446			LET CC := CC CLR.BY #177760			
14513	047446	042737	177760	000520		BIC	#177760,CC
14514	047454			IF CC NE #0 THEN			
14515	047454	005737	000520			TST	CC
14516	047460	001402				BEQ	\$52077
14517	047462			CALL ERROR			
14518	047462	004737	060044			JSR	PC,ERROR
14519	047466			ENDIF			
14520	047466					\$52077:	
14521							
14522				:+			
14523				:-			
14524				ENSURE GPR'S STILL ZERO			
14525							
14526	047466			IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN			
14527	047466	005700				TST	R0
14528	047470	001004				BNE	\$52100
14529	047472	005701				TST	R1
14530	047474	001002				BNE	\$52100
14531	047476	005702				TST	R2
14532	047500	001402				BEQ	\$52101
14533	047502					\$52100:	
14534	047502			CALL ERROR			
14535	047502	004737	060044			JSR	PC,ERROR

```
14536 047506          ENDIF
14537 047506
14538 047506          IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
14539 047506 005703
14540 047510 001004
14541 047512 005704
14542 047514 001002
14543 047516 005705
14544 047520 001402
14545 047522
14546 047522          CALL ERROR
14547 047522 004737 060044
14548 047526          ENDIF
14549 047526
14550
14551
14552          :+          CHECK RESULTS
14553          :-
14554
14555 047526          LET R0 := #DS.DST
14556 047526 012700 064400
14557 047532          LET R1 := #DS.EXP          ;EXPECTED DATA
14558 047532 012701 064442
14559 047536          LET R2 := TADDPI+10 CLR.BY #70000          ;GET STRING LENGTH
14560 047536 013702 066002
14561 047542 042702 070000
14562 047546          INLINE <ASR R2>          ;LENGTH/2 (PACKING FACTOR)
14563 047546 006202
14564 047550          LET R2 := R2 + #DS.EXP          ;R2 POINTS TO MSD/SIGN
14565 047550 062702 064442
14566 047554          LET (R2) :B= (R2) SET.BY #17          ;MAKE IT UNSIGNED
14567 047554 152712 000017
14568 047560          LET ERRFLG := #0
14569 047560 005037 000456
14570 047564          WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO
14571 047564
14572 047564 020027 064437
14573 047570 001410
14574 047572 005737 000456
14575 047576 001005
14576 047600          IFB (R0)+ NE (R1)+ THEN
14577 047600 122021
14578 047602 001402
14579 047604          CALL ERROR
14580 047604 004737 060044
14581 047610          ENDIF
14582 047610
14583 047610          ENDDO
14584 047610 000765
14585 047612
14586
14587

$52101:
TST R3
BNE $52102
TST R4
BNE $52102
TST R5
BEQ $52103
$52102:
JSR PC,ERROR
$52103:

MOV #DS.DST,R0
MOV #DS.EXP,R1
MOV TADDPI+10,R2
BIC #70000,R2
ASR R2
ADD #DS.EXP,R2
BISB #17,(R2)
CLR ERRFLG
$52104:
CMP R0,#DS.DST+37
BEQ $52105
TST ERRFLG
BNE $52105
CMPB (R0)+,(R1)+
BEQ $52106
JSR PC,ERROR
$52106:
BR $52104
$52105:
```

```

14588 .SBTTL CMPP
14589 :*****
14590 :
14591 : CMPP TEST
14592 :
14593 :*****
14594 :
14595 047612 INLINE <XCMP: >
14596 047612 XCMPP:
14597 047612 INLINE <SCOPE > SCOPE
14598 047612 000004 ;ITERATION COUNT
14599 047614 LET $TIMES := #20 ;GET TEST COUNT MOV #20,$TIMES
14600 047614 012737 000020 000404 ;POINT TO FIRST TEST CASE MOV #TCMP,COUNT
14601 047622 LET COUNT := TCMP ;POINT TO FIRST TEST CASE MOV #TCMP+2,TSTCAS
14602 047622 013737 064502 000470
14603 047630 LET TSTCAS := #TCMP+2
14604 047630 012737 064504 000544
14605 ;REPEAT
14606 047636 INLINE <$5072: >
14607 047636 $5072:
14608 047636 CALL CLRDSB
14609 047636 004737 057040 JSR PC,CLRDSB
14610 047642 LET CC.EXP :- #0
14611 047642 005037 000522 CLR CC.EXP
14612 047646 LET SIGN0 := #0
14613 047646 005037 000450 CLR SIGN0
14614 047652 LET DATTYP :- #0
14615 047652 005037 000464 CLR DATTYP
14616
14617 ;+
14618 ; GET SOURCE DESCRIPTORS FROM TABLE
14619 ;-
14620
14621 047656 LET R0 := TSTCAS
14622 047656 013700 000544 MOV TSTCAS,R0
14623 047662 LET DATA00 := (R0)+ ;SRC1 LENGTH
14624 047662 012037 000562 MOV (R0)+,DATA00
14625 047666 LET DATA01 := (R0)+ ;SRC1 POINTER
14626 047666 012037 000564 MOV (R0)+,DATA01
14627 047672 LET DATA02 := (R0)+ ;SRC2 LENGTH
14628 047672 012037 000566 MOV (R0)+,DATA02
14629 047676 LET DATA03 := (R0) ;SRC2 POINTER
14630 047676 011037 000570 MOV (R0),DATA03
14631
14632 ;+
14633 ; LOAD SRC1 AND SRC2 INTO DSBUF1 AND DSBUF2
14634 ; IN NON-PACKED FORMAT, FOR NOW.
14635 ;-
14636
14637 047702 LET R2 := DATA01 + #2 ;GET SRC1 ADDRESS
14638 047702 013702 000564 MOV DATA01,R2
14639 047706 062702 000002 ADD #2,R2
14640 047712 LET R3 := #DSBUF1
14641 047712 012703 064274 MOV #DSBUF1,R3
14642 047716 LET DSTMP0 :B= (R2) ;SAVE SRC1 MSD
14643 047716 111237 000554 MOVB (R2),DSTMP0
    
```

```

14644 047722          DECR R1 FROM DATA00 TO #0 BY #1
14645 047722 013701 000562          MOV DATA00,R1
14646 047726 000401          BR $52107
14647 047730          $52110:
14648 047730 005301          DEC R1
14649 047732          $52107:
14650 047732 005701          TST R1
14651 047734 002402          BLT $52111
14652 047736          LET (R3)+ :B= (R2)+
14653 047736 112223          MOVB (R2)+,(R3)+
14654 047740          ENDDEC
14655 047740 000773          BR $52110
14656 047742          $52111:
14657 047742          LET DSTMP1 :B= -(R2)          ;SAVE SRC1 LSD
14658 047742 114237 000555          MOVB -(R2),DSTMP1
14659 047746          LET R2 :- DATA03 + #2          ;GET SRC2 ADDRESS
14660 047746 013702 000570          MOV DATA03,R2
14661 047752 062702 000002          ADD #2,R2
14662 047756          LET R3 := #DSBUF2
14663 047756 012703 064336          MOV #DSBUF2,R3
14664 047762          LET DSTMP2 :B= (R2)          ;SAVE SRC2 MSD
14665 047762 111237 000556          MOVB (R2),DSTMP2
14666 047766          DECR R1 FROM DATA02 TO #0 BY #1
14667 047766 013701 000566          MOV DATA02,R1
14668 047772 000401          BR $52112
14669 047774          $52113:
14670 047774 005301          DEC R1
14671 047776          $52112:
14672 047776 005701          TST R1
14673 050000 002402          BLT $52114
14674 050002          LET (R3)+ :B= (R2)+
14675 050002 112223          MOVB (R2)+,(R3)+
14676 050004          ENDDEC
14677 050004 000773          BR $52113
14678 050006          $52114:
14679 050006          LET DSTMP3 :B= -(R2)          ;SAVE SRC2 LSD
14680 050006 114237 000557          MOVB -(R2),DSTMP3
14681
14682          ;+
14683          ; COMPARE SRC1 AND SRC2, AND SET N AND 7 BITS
14684          ; IN CC.EXP ACCORDINGLY.
14685          ;-
14686
14687 050012          IF DATA00 EQ #0 AND DATA02 EQ #0 THEN          ;SRC1 LENGTH AND SRC2 LENGTH = 0
14688 050012 005737 000562          TST DATA00
14689 050016 001007          BNE $52115
14690 050020 005737 000566          TST DATA02
14691 050024 001004          BNE $52115
14692 050026          LET CC.EXP := CC.EXP SET.BY #ZBIT
14693 050026 052737 000004 000522          BIS #ZBIT,CC.EXP
14694 050034          ELSE
14695 050034 000424          BR $52116
14696 050036          $52115:
14697 050036          IF DATA00 EQ #0 AND @DATA03 EQ #0 THEN          ;SRC1 LENGTH = 0 AND SRC2 SIGN IS +
14698 050036 005737 000562          TST DATA00
14699 050042 001007          BNE $52117

```

```

14700 050044 005777 130520          TST @DATA03
14701 050050 001004          BNE $52117
14702 050052          LET CC.EXP := CC.EXP SET.BY #NBIT
14703 050052 052737 000010 000522  BIS #NBIT,CC.EXP
14704 050060          ELSE
14705 050060 000412          BR $52120
14706 050062          $52117:
14707 050062          IF DATA02 EQ #0 AND @DATA01 EQ #-1 THEN ;SRC2 LENGTH = 0 & SRC1 SIGN IS -
14708 050062 005737 000566          TST DATA02
14709 050066 001007          BNE $52121
14710 050070 027727 130470 177777  CMP @DATA01,#-1
14711 050076 001003          BNE $52121
14712 050100          LET CC.EXP := CC.EXP SET.BY #NBIT
14713 050100 052737 000010 000522  BIS #NBIT,CC.EXP
14714 050106          ENDIF
14715 050106          $52121:
14716 050106          ENDIF
14717 050106          $52120:
14718 050106          ENDIF
14719 050106          $52116:
14720 050106          IF DATA00 NE #0 AND DATA02 NE #0 THEN ;SRC1 LENGTH AND SRC2 LENGTH > 0
14721 050106 005737 000562          TST DATA00
14722 050112 001520          BEQ $52122
14723 050114 005737 000566          TST DATA02
14724 050120 001515          BEQ $52122
14725 050122          IF @DATA01 NE @DATA03 THEN ;SRC1 SIGN AND SRC2 SIGN ARE DIFFERENT
14726 050122 027777 130436 130440  CMP @DATA01,@DATA03
14727 050130 001410          BEQ $52123
14728 050132          IF @DATA01 EQ #-1 THEN ;SRC1 SIGN IS -
14729 050132 027727 130426 177777  CMP @DATA01,#-1
14730 050140 001003          BNE $52124
14731 050142          LET CC.EXP := CC.EXP SET.BY #NBIT
14732 050142 052737 000010 000522  BIS #NBIT,CC.EXP
14733 050150          ENDIF
14734 050150          $52124:
14735 050150          ELSE ;SRC1 SIGN AND SRC2 SIGN ARE EQUAL
14736 050150 000501          BR $52125
14737 050152          $52123:
14738 050152          LET SIGN0 := @DATA01 ;SAVE THE COMMON SIGN
14739 050152 017737 130406 000450  MOV @DATA01,SIGN0
14740 050160          IF DATA00 LO DATA02 AND SIGN0 EQ #0 THEN ;SRC1 LENGTH < SRC2 LENGTH AND SIGNS
14741 050160 023737 000562 000566  CMP DATA00,DATA02
14742 050166 103007          BHIS $52126
14743 050170 005737 000450          TST SIGN0
14744 050174 001004          BNE $52126
14745 050176          LET CC.EXP := CC.EXP SET.BY #NBIT
14746 050176 052737 000010 000522  BIS #NBIT,CC.EXP
14747 050204          ELSE
14748 050204 000413          BR $52127
14749 050206          $52126:
14750 050206          IF DATA00 HI DATA02 AND SIGN0 EQ #-1 THEN ;SRC1 LENGTH > SRC2 LENGTH AND
14751 050206 023737 000562 000566  CMP DATA00,DATA02
14752 050214 101407          BLOS $52130
14753 050216 023727 000450 177777  CMP SIGN0,#-1
14754 050224 001003          BNE $52130
14755 050226          LET CC.EXP := CC.EXP SET.BY #NBIT

```

```

14756 050226 052737 000010 000522          BIS      #NBIT,CC.EXP
14757 050234          ENDIF          $52130:
14758 050234          ENDIF          $52127:
14759 050234          IF DATA00 EQ DATA02 THEN ;SRC1 LENGTH AND SRC2 LENGTH ARE EQUAL
14760 050234          ;SRC1 LENGTH AND SRC2 LENGTH ARE EQUAL
14761 050234          CMP      DATA00,DATA02
14762 050234 023737 000562 000566          BNE      $52131
14763 050242 001044
14764
14765          :+
14766          : COMPARE SRC1 AND SRC2 BYTE BY BYTE
14767          :-
14768
14769 050244          LET R0 := DATA00          ;SRC BUFFER LENGTH
14770 050244 013700 000562          MOV      DATA00,R0
14771 050250          LET R1 := #DSBUF1          ;SRC1 BUFFER
14772 050250 012701 064274          MOV      #DSBUF1,R1
14773 050254          LET R2 := #DSBUF2          ;SRC2 BUFFER
14774 050254 012702 064336          MOV      #DSBUF2,R2
14775 050260          WHILEB R0 NE #0 AND (R1)+ EQ (R2)+ DO
14776 050260          $52132:
14777 050260 105700          TSTB    R0
14778 050262 001404          BEQ    $52133
14779 050264 022122          CMP    (R1)+,(R2)+
14780 050266 001002          BNE    $52133
14781 050270          LET R0 := R0 - #1          ;DECREMENT COUNTER
14782 050270 005300          DEC    R0
14783 050272          ENDDO
14784 050272 000772          BR     $52132
14785 050274          $52133:
14786 050274          IFB -(R1) NE -(R2) THEN ;SOURCES WERE NOT =
14787 050274 124142          CMPB   -(R1),-(R2)
14788 050276 001423          BEQ    $52134
14789 050300          IFB (R1) LO (R2) AND SIGN0 EQ #0 THEN
14790 050300 121112          CMPB   (R1),(R2)
14791 050302 103007          BHIS   $52135
14792 050304 005737 000450          TST    SIGN0
14793 050310 001004          BNE    $52135
14794 050312          LET CC.EXP := CC.EXP SET.BY #NBIT
14795 050312 052737 000010 000522          BIS    #NBIT,CC.EXP
14796 050320          ELSE
14797 050320 000411          BR     $52136
14798 050322          $52135:
14799 050322          IFB (R1) HI (R2) AND SIGN0 EQ #-1 THEN
14800 050322 121112          CMPB   (R1),(R2)
14801 050324 101407          BLOS   $52137
14802 050326 023727 000450 177777          CMP    SIGN0,#-1
14803 050334 001003          BNE    $52137
14804 050336          LET CC.EXP := CC.EXP SET.BY #NBIT
14805 050336 052737 000010 000522          BIS    #NBIT,CC.EXP
14806 050344          ENDIF
14807 050344          ENDIF
14808 050344          ENDIF
14809 050344          ELSE
14810 050344          ;SOURCES WERE EQUAL
14811 050344 000403          BR     $52140
    
```

14812	050346						\$52134:
14813	050346				LET CC.EXP := CC.EXP SET.BY #ZBIT		
14814	050346	052737	000004	000522			BIS #ZBIT,CC.EXP
14815	050354				ENDIF		
14816	050354						\$52140:
14817	050354				ENDIF		
14818	050354						\$52131:
14819	050354				ENDIF		
14820	050354						\$52125:
14821	050354				ENDIF		
14822	050354						\$52122:
14823							
14824							
14825					:+		
14826					: PERFORM THE CMPP INSTRUCTION WITH THE SIGNED		
14827					: PACKED DATA TYPE. (UNSIGNED PACKED WILL BE		
14828					: TESTED IN THE INLINE PORTION OF THE TEST).		
14829					: CHECK THE RESULTS AND REPORT ANY ERRORS.		
14830					:-		
14831	050354				CALL CLRDSB		
14832	050354	004737	057040			JSR PC,CLRDSB	
14833					:LOAD DSBUF1 WITH SRC1 DECIMAL STRING AND SIGN,		IN PACKED FORMAT.
14834	050360				LET R0 := #DSBUF1		
14835	050360	012700	064274			MOV #DSBUF1,R0	
14836	050364				LET R1 := DATA01 + #2		
14837	050364	013701	000564			MOV DATA01,R1	
14838	050370	062701	000002			ADD #2,R1	
14839	050374				LET R2 := DATA00	:DECIMAL STRING	LENGTH
14840	050374	013702	000562			MOV DATA00,R2	
14841	050400				IF R2 EQ #0 THEN	:LENGTH = 0	
14842	050400	005702				TST R2	
14843	050402	001011				BNE \$52141	
14844	050404				IF @DATA01 EQ #0 THEN	:SIGN IS +	
14845	050404	005777	130154			TST @DATA01	
14846	050410	001003				BNE \$52142	
14847	050412				LET (R0) :B= #14		
14848	050412	112710	000014			MOVB #14,(R0)	
14849	050416				ELSE	:SIGN IS -	
14850	050416	000402				BR \$52143	
14851	050420					\$52142:	
14852	050420				LET (R0) :B= #15		
14853	050420	112710	000015			MOVB #15,(R0)	
14854	050424				ENDIF		
14855	050424					\$52143:	
14856	050424				ELSE		
14857	050424	000434				BR \$52144	
14858	050426					\$52141:	
14859	050426				IF #BIT00 NOTSET IN R2 THEN	:LENGTH IS EVEN AND > 0	
14860	050426	032702	000001			BIT #BIT00,R2	
14861	050432	001002				BNE \$52145	
14862	050434				LET (R0)+ :B= (R1)+	:MOVE THE EVEN NIBBLE	
14863	050434	112120				MOVB (R1)+,(R0)+	
14864	050436				LET R2 := R2 - #1	:DECREMENT COUNTER	
14865	050436	005302				DEC R2	
14866	050440				ENDIF		
14867	050440					\$52145:	

14868	050440			WHILE R2 NE #1 DO	
14869	050440				\$52146:
14870	050440	020227	000001		CMP R2,#1
14871	050444	001410			BEQ \$52147
14872	050446			LET R3 :B= (R1)+	;GET A BCD
14873	050446	112103			MOVB (R1)+,R3
14874	050450			INLINE <ASH #4,R3>	;SHIFT IT TO THE HIGH NIBBLE
14875	050450	072327	000004		ASH #4,R3
14876	050454			LET R3 :B= R3 SET.BY (R1)+	;PUT NEXT BCD INTO LOWER NIBBLE
14877	050454	152103			BISB (R1)+,R3
14878	050456			LET (R0)+ :B= R3	
14879	050456	110320			MOVB R3,(R0)+
14880	050460			LET R2 := R2 - #2	;DECREMENT COUNTER
14881	050460	162702	000002		SUB #2,R2
14882	050464			ENDDO	
14883	050464	000765			BR \$52146
14884	050466				\$52147:
14885	050466			LET R3 :B= (R1)	;GET LAST BCD
14886	050466	111103			MOVB (R1),R3
14887	050470			INLINE <ASH #4,R3>	
14888	050470	072327	000004		ASH #4,R3
14889	050474			IF @DATA01 EQ #0 THEN	;SIGN IS +
14890	050474	005777	130064		TST @DATA01
14891	050500	001003			BNE \$52150
14892	050502			LET R3 :B= R3 SET.BY #14	
14893	050502	152703	000014		BISB #14,R3
14894	050506			ELSE	;SIGN IS -
14895	050506	000402			BR \$52151
14896	050510				\$52150:
14897	050510			LET R3 :B= R3 SET.BY #15	
14898	050510	152703	000015		BISB #15,R3
14899	050514			ENDIF	
14900	050514			LET (R0) :B= R3	;PUT MSD AND SIGN INTO THE BUFFER
14901	050514				MOVB R3,(R0)
14902	050514	110310		ENDIF	
14903	050516				\$52151:
14904	050516				\$52144:
14905				;LOAD DSBUF2 WITH SRC2 DECIMAL STRING AND SIGN, IN PACKED FORMAT.	
14906	050516			LET R0 := #DSBUF2	
14907	050516	012700	064336		MOV #DSBUF2,R0
14908	050522			LET R1 := DATA03 + #2	
14909	050522	013701	000570		MOV DATA03,R1
14910	050526	062701	000002		ADD #2,R1
14911	050532			LET R2 := DATA02	;DECIMAL STRING LENGTH
14912	050532	013702	000566		MOV DATA02,R2
14913	050536			IF R2 EQ #0 THEN	;LENGTH = 0
14914	050536	005702			TST R2
14915	050540	001011			BNE \$52152
14916	050542			IF @DATA03 EQ #0 THEN	;SIGN IS +
14917	050542	005777	130022		TST @DATA03
14918	050546	001003			BNE \$52153
14919	050550			LET (R0) :B= #14	
14920	050550	112710	000014		MOVB #14,(R0)
14921	050554			ELSE	;SIGN IS -
14922	050554	000402			BR \$52154
14923	050556				\$52153:

```

14924 050556          LET (R0) :B= #15
14925 050556 112710 000015          MOVB #15,(R0)
14926 050562          ENDIF
14927 050562          $52154:
14928 050562          ELSE
14929 050562 000434          BR $52155
14930 050564          $52152:
14931 050564          IF #BIT00 NOTSET IN R2 THEN ;LENGTH IS EVEN AND > 0
14932 050564 032702 000001          BIT #BIT00,R2
14933 050570 001002          BNE $52156
14934 050572          LET (R0)+ :B= (R1)+ ;MOVE THE EVEN NIBBLE
14935 050572 112120          MOVB (R1)+,(R0)+
14936 050574          LET R2 := R2 - #1 ;DECREMENT COUNTER
14937 050574 005302          DEC R2
14938 050576          ENDIF
14939 050576          $52156:
14940 050576          WHILE R2 NE #1 DO
14941 050576          $52157:
14942 050576 020227 000001          CMP R2,#1
14943 050602 001410          BEQ $52160
14944 050604          LET R3 :B= (R1)+ ;GET A BCD
14945 050604 112103          MOVB (R1)+,R3
14946 050606          INLINE <ASH #4,R3> ;SHIFT IT TO THE HIGH NIBBLE
14947 050606 072327 000004          ASH #4,R3
14948 050612          LET R3 :B= R3 SET.BY (R1)+ ;PUT NEXT BCD INTO LOWER NIBBLE
14949 050612 152103          BISB (R1)+,R3
14950 050614          LET (R0)+ :B= R3
14951 050614 110320          MOVB R3,(R0)+
14952 050616          LET R2 := R2 - #2 ;DECREMENT COUNTER
14953 050616 162702 000002          SUB #2,R2
14954 050622          ENDDO
14955 050622 000765          BR $52157
14956 050624          $52160:
14957 050624          LET R3 :B= (R1)
14958 050624 111103          MOVB (R1),R3
14959 050626          INLINE <ASH #4,R3>
14960 050626 072327 000004          ASH #4,R3
14961 050632          IF @DATA03 EQ #0 THEN ;SIGN IS +
14962 050632 005777 127732          TST @DATA03
14963 050636 001003          BNE $52161
14964 050640          LET R3 :B= R3 SET.BY #14
14965 050640 152703 000014          BISB #14,R3
14966 050644          ELSE ;SIGN IS -
14967 050644 000402          BR $52162
14968 050646          $52161:
14969 050646          LET R3 :B= R3 SET.BY #15
14970 050646 152703 000015          BISB #15,R3
14971 050652          ENDIF
14972 050652          $52162:
14973 050652          LET (R0) :B= R3 ;PUT MSD AND SIGN INTO THE BUFFER
14974 050652 110310          MOVB R3,(R0)
14975 050654          ENDIF
14976 050654          $52155:
14977
14978
14979          ;+
          ; LOAD THE GPR'S AND DO THE CMPP INSTRUCTION

```

```

14980
14981
14982 050654          LET R0 := #60000 SET.BY DATA00 ;SRC1.DSCR
14983 050654 012700 060000          MOV #60000,R0
14984 050660 053700 000562          BIS DATA00,R0
14985 050664          LET R1 := #DSBUF1          ; ""
14986 050664 012701 064274          MOV #DSBUF1,R1
14987 050670          LET R2 := #60000 SET.BY DATA02 ;SRC2.DSCR
14988 050670 012702 060000          MOV #60000,R2
14989 050674 053702 000566          BIS DATA02,R2
14990 050700          LET R3 := #DSBUF2          ; ""
14991 050700 012703 064336          MOV #DSBUF2,R3
14992 050704          LET R4 := #0
14993 050704 005004          CLR R4
14994 050706          LET R5 := #0
14995 050706 005005          CLR R5
14996 050710          INLINE <CMPP>
14997 050710 076072          CMPP
14998
14999
15000
15001
15002
15003
15004 050712          :+
15005 050712 013737 177776 000520      : CHECK THE RESULTS AND REPORT ANY ERRORS
15006 050720          :-
15007 050720 042737 177760 000520      LET CC := PSW
15008 050726 023737 000520 000522      MOV PSW,CC
15009 050734 001402          LET CC := CC CLR.BY #177760
15010 050736          BIC #177760,CC
15011 050736 004737 060044          IF CC NE CC.EXP THEN
15012 050742          CALL ERROR
15013 050742          CMP CC,CC.EXP
15014 050742          BEQ $52163
15015 050742 005700          JSR PC,ERROR
15016 050744 001004          ENDF
15017 050746 005701          IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN
15018 050750 001002          $52163:
15019 050752 005702          TST R0
15020 050754 001402          BNE $52164
15021 050756          TST R1
15022 050756          BNE $52164
15023 050756 004737 060044          TST R2
15024 050762          BEQ $52165
15025 050762          $52164:
15026 050762          CALL ERROR
15027 050762 005703          JSR PC,ERROR
15028 050764 001004          ENDF
15029 050766 005704          IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN
15030 050770 001002          $52165:
15031 050772 005705          TST R3
15032 050774 001402          BNE $52166
15033 050776          TST R4
15034 050776          BNE $52166
15035 050776 004737 060044          TST R5
15036          BEQ $52167
15037          $52166:
15038          CALL ERROR
15039          JSR PC,ERROR
    
```

15036	051002			ENDIF		\$52167:
15037	051002					
15038	051002			LET TSTCAS := TSTCAS + #10		ADD #10,TSTCAS
15039	051002	062737	000010	000544		
15040	051010			LET COUNT := COUNT - #1		DEC COUNT
15041	051010	005337	000470			
15042				:UNTIL COUNT EQ #0		
15043	051014			IF COUNT NE #0 THEN		TST COUNT
15044	051014	005737	000470			BEQ \$52170
15045	051020	001402				
15046	051022			INLINE <JMP \$5072>		JMP \$5072
15047	051022	000137	047636			
15048	051026			ENDIF		\$52170:
15049	051026					
15050						
15051						
15052						
15053				:+		
15054				: DO ONE INLINE CASE - CMPP1. USE UNSIGNED PACKED DATA TYPE.		
15055				:-		
15056						
15057	051026			LET R0 := #-1		
15058	051026	012700	177777			MOV #-1,R0
15059	051032			LET R1 := #-1		
15060	051032	012701	177777			MOV #-1,R1
15061	051036			LET R2 := #-1		
15062	051036	012702	177777			MOV #-1,R2
15063	051042			LET R3 := #-1		
15064	051042	012703	177777			MOV #-1,R3
15065	051046			LET R4 := #-1		
15066	051046	012704	177777			MOV #-1,R4
15067	051052			LET R5 := #-1		
15068	051052	012705	177777			MOV #-1,R5
15069	051056			INLINE <CMPP1>		CMPP1
15070	051056	076172				
15071	051060			INLINE <.WORD TCMPP1>		.WORD TCMPP1
15072	051060	051174				
15073	051062			INLINE <.WORD TCMPP1+4>		.WORD TCMPP1+4
15074	051062	051200				
15075	051064			LET CC := PSW		
15076	051064	013737	177776	000520		MOV PSW,CC
15077	051072			LET CC := CC CLR.BY #177760		
15078	051072	042737	177760	000520		BIC #177760,CC
15079	051100			IF CC NE #4 THEN		
15080	051100	023727	000520	000004		CMP CC,#4
15081	051106	001402				BEQ \$52171
15082	051110			CALL ERROR		
15083	051110	004737	060044			JSR PC,ERROR
15084	051114			ENDIF		\$52171:
15085	051114					
15086	051114			IF R0 NE #-1 OR R1 NE #-1 OR R2 NE #-1 THEN		
15087	051114	020027	177777			CMP R0,#-1
15088	051120	001006				BNE \$52172
15089	051122	020127	177777			CMP R1,#-1
15090	051126	001003				BNE \$52172
15091	051130	020227	177777			CMP R2,#-1


```

15127      .SBTTL ASHP
15128      :*****
15129      :
15130      : ASHP TEST
15131      :
15132      :*****
15133
15134      051210      INLINE <XASHP:>
15135      051210
15136      051210
15137      051210      000004
15138      051212
15139      051212      012737      000020      000404
15140      051220
15141      051220      013737      064750      000470
15142      051226
15143      051226      012737      064752      000544
15144      051234
15145      051234      005037      000450
15146
15147      051240
15148      051240
15149      051240
15150      051240      004737      057040
15151      051244
15152      051244      005037      000522
15153
15154      :+
15155      : GET TEST DATA FROM TABLE
15156      :-
15157
15158      051250
15159      051250      013700      000544
15160      051254
15161      051254      012037      000562
15162      051260
15163      051260      012037      000564
15164      051264
15165      051264      011037      000566
15166
15167      :+
15168      :+
15169      : PUT THE TEST STRING INTO DSBUF1 IN SIGNED PACKED FORMAT. INCLUDE
15170      : THE SIGN.
15171      :-
15172
15173      051270
15174      051270      012700      064274
15175      051274
15176      051274      013701      000562
15177      051300
15178      051300      005701
15179      051302      001011
15180      051304
15181      051304      005737      000450
15182      051310      001003
    
```

```

XASHP:
SCOPE
:ITERATION COUNT
MOV #20,$TIMES
:GET TEST COUNT
MOV TASH,COUNT
:POINT TO FIRST TEST CASE
MOV #TASH+2,TSTCAS
:INITIALIZE THE SIGN WORD
CLR SIGNO

$5076:
:CLEAR DECIMAL STRING BUFFERS
JSR PC,CLRDSB
CLR CC.EXP

MOV TSTCAS,R0
:(R0)+,DATA00
:(R0)+,DATA01
:(R0)+,DATA02
MOV #DSBUF1,R0
MOV DATA00,R1
:SRC.LEN = 0
TST R1
BNE $52176
:SIGN IS +
TST SIGNO
BNE $52177
    
```

```
15183 051312          LET (R0) :B= #14
15184 051312 112710 000014
15185 051316          ELSE
15186 051316 000402          ;SIGN IS -
15187 051320          BR          $52200
15188 051320          LET (R0) :B= #15          $52177:
15189 051320 112710 000015          MOVB          #15,(R0)
15190 051324          ENDIF          $52200:
15191 051324          ELSE
15192 051324          BR          $52201
15193 051324 000426          $52176:
15194 051326          IF #BIT00 NOTSET IN R1 THEN ;SRC.LEN IS EVEN AND > 0
15195 051326 032701 000001          BIT          #BIT00,R1
15196 051326 001003          BNE          $52202
15197 051332          LET (R0)+ :B= #4
15198 051334 112720 000004          MOVB          #4,(R0)+
15199 051334          LET R1 := R1 - #1          ;DECREMENT COUNT
15200 051340          ENDIF          DEC          R1
15201 051340 005301          WHILE R1 NE #1 DO          $52202:
15202 051342          $52203:
15203 051342 020127 000001          CMP          R1,#1
15204 051342 001405          BEQ          $52204
15205 051346          ;MOVE TWO NIBBLES OF BCD
15206 051350          LET (R0)+ :B= #104          MOVB          #104,(R0)+
15207 051350 112720 000104          ;DECREMENT COUNT
15208 051354          LET R1 := R1 - #2          SUB          #2,R1
15209 051354 162701 000002          ENDDO
15210 051360          BR          $52203
15211 051360 000770          $52204:
15212 051362          IF SIGN0 EQ #0 THEN          ;SIGN IS +
15213 051362 005737 000450          TST          SIGN0
15214 051362 001003          BNE          $52205
15215 051370          LET (R0) :B- #114
15216 051370 112710 000114          ;SIGN IS -
15217 051374          ELSE          BR          $52206
15218 051374 000402          $52205:
15219 051376          LET (R0) :B #115          MOVB          #115,(R0)
15220 051376 112710 000115          ENDIF
15221 051402          ENDIF          $52206:
15222 051402          $52201:
15223 051402
15224 051402
15225 051402
15226 051402
15227 051402
15228 051402
15229
15230          ;+
15231          ; LOAD DS.EXP AND CC.EXP WITH EXPECTED RESULTS.
15232          ;-
15233
15234          ;GET SHIFT COUNT MAGNITUDE AND PUT IT INTO TEMPO.
15235
15236 051402          LET TEMPO := DATA02
15237 051402 013737 000566 000476          IF #BIT07 SET IN TEMPO THEN
15238 051410          MOV          DATA02,TEMPO
          ;NEGATIVE SHIFT COUNT
```

15239	051410	032737	000200	000476		BIT	#BIT07,TEMPO
15240	051416	001402				BEQ	\$52207
15241	051420				LET TEMPO := NEGATE TEMPO		
15242	051420	005437	000476		ENDIF	NEG	TEMPO
15243	051424						
15244	051424				LET TEMPO := TEMPO CLR.BY #177600	\$52207:	
15245	051424						
15246	051424	042737	177600	000476		BIC	#177600,TEMPO
15247							
15248					:TAKE CARE OF EXPECTED NBIT		
15249							
15250	051432				IF SIGN0 EQ #-1 AND DATA00 NE #0 THEN		
15251	051432	023727	000450	177777		CMP	SIGN0,#-1
15252	051440	001006				BNE	\$52210
15253	051442	005737	000562			TST	DATA00
15254	051446	001403				BEQ	\$52210
15255	051450				LET CC.EXP := CC.EXP SET.BY #NBIT		
15256	051450	052737	000010	000522	ENDIF	BIS	#NBIT,CC.EXP
15257	051456						
15258	051456					\$52210:	
15259							
15260					:PUT THE SIGN NIBBLE INTO THE EXPECTED DESTINATION		
15261							
15262	051456				LET R1 := DATA01 ;DST.LEN		
15263	051456	013701	000564			MOV	DATA01,R1
15264	051462				INLINE <ASR R1>		
15265	051462	006201				ASR	R1
15266	051464				LET R1 := R1 + #DS.EXP		
15267	051464	062701	064442			ADD	#DS.EXP,R1
15268	051470				IF SIGN0 EQ #0 OR DATA00 EQ #0 THEN		
15269	051470	005737	000450			TST	SIGN0
15270	051474	001403				BEQ	\$52211
15271	051476	005737	000562			TST	DATA00
15272	051502	001003				BNE	\$52212
15273	051504					\$52211:	
15274	051504				LET (R1) :B= (R1) SET.BY #14 ;POSITIVE SIGN		
15275	051504	152711	000014			BISB	#14,(R1)
15276	051510				ELSE		
15277	051510	000402				BR	\$52213
15278	051512					\$52212:	
15279	051512				LET (R1) :B= (R1) SET.BY #15 ;NEGATIVE SIGN		
15280	051512	152711	000015			BISB	#15,(R1)
15281	051516				ENDIF		
15282	051516					\$52213:	
15283							
15284					:TAKE CARE OF ZERO LENGTH SOURCE OR DESTINATION		
15285							
15286	051516				IF DATA00 EQ #0 OR DATA01 EQ #0 THEN		
15287	051516	005737	000562			TST	DATA00
15288	051522	001403				BEQ	\$52214
15289	051524	005737	000564			TST	DATA01
15290	051530	001003				BNE	\$52215
15291	051532					\$52214:	
15292	051532				LET CC.EXP := CC.EXP SET.BY #ZBIT		
15293	051532	052737	000004	000522	ENDIF	BIS	#ZBIT,CC.EXP
15294	051540						

15295	051540							\$52215:
15296	051540							
15297	051540	005737	000562					IF DATA00 NE #0 AND DATA01 NE #0 AND #BIT07 SET IN DATA02 THEN
15298	051544	001531						TST DATA00
15299	051546	005737	000564					BEQ \$52216
15300	051552	001526						TST DATA01
15301	051554	032737	000200	000566				BEQ \$52216
15302	051562	001522						BIT #BIT07,DATA02
15303								BEQ \$52216
15304								
15305								;TAKE CARE OF NEGATIVE SHIFT
15306	051564							LET R0 := DATA00 - TEMPO ;SRC.LEN - SHIFT COUNT
15307	051564	013700	000562					MOV DATA00,R0
15308	051570	163700	000476					SUB TEMPO,R0
15309	051574							;DST.LEN
15310	051574	013701	000564					MOV DATA01,R1
15311	051600							ASR R1
15312	051600	006201						LET R1 := R1 + #DS.EXP
15313	051602							ADD #DS.EXP,R1
15314	051602	062701	064442					LET R2 := #1
15315	051606							MOV #1,R2
15316	051606	012702	000001					MOV DATA01,R3
15317	051612							MOV DATA01,R3
15318	051612	013703	000564					TST R0
15319	051616							BNE \$52217
15320	051616	005700						
15321	051620	001024						IF DATA02 HIS #3000 THEN
15322	051622							
15323	051622	023727	000566	003000				CMP DATA02,#3000
15324	051630	103403						BLO \$52220
15325	051632							;ROUNDING
15326	051632	152711	000020					BISB #20,(R1)
15327	051636							BR \$52221
15328	051636	000415						\$52220:
15329	051640							BIS #ZBIT,CC.EXP
15330	051640							LET CC.EXP := CC.EXP SET.BY #ZBIT
15331	051640	052737	000004	000522				BIC #NBIT,CC.EXP
15332	051646							LET CC.EXP := CC.EXP CLR.BY #NBIT
15333	051646	042737	000010	000522				MOV DATA01,R5
15334	051654							LET R5 := DATA01
15335	051654	013705	000564					ASR R5
15336	051660							INLINE <ASR R5>
15337	051660	006205						LET R5 := R5 + #DS.EXP
15338	051662							ADD #DS.EXP,R5
15339	051662	062705	064442					LET (R5) := (R5) CLR.BY #1
15340	051666							BICB #1,(R5)
15341	051666	142715	000001					\$52221:
15342	051672							ENDIF
15343	051672							\$52217:
15344	051672							ENDIF
15345	051672							IF R0 LT #0 THEN
15346	051672							TST R0
15347	051672	005700						BGE \$52222
15348	051674	002015						
15349	051676							LET CC.EXP := CC.EXP SET.BY #ZBIT
15350	051676	052737	000004	000522				BIS #ZBIT,CC.EXP

15351	051704				LET CC.EXP := CC.EXP CLR.BY #NBIT		
15352	051704	042737	000010	000522		BIC	#NBIT,CC.EXP
15353	051712				LET R5 := DATA01	MOV	DATA01,R5
15354	051712	013705	000564		INLINE <ASR R5>	ASR	R5
15355	051716						
15356	051716	006205			LET R5 := R5 + #DS.EXP	ADD	#DS.EXP,R5
15357	051720						
15358	051720	062705	064442		LET (R5) :B= (R5) CLR.BY #1	BICB	#1,(R5)
15359	051724						
15360	051724	142715	000001		ENDIF	\$52222:	
15361	051730						
15362	051730				WHILE R2 LE R0 AND R3 HI #0 DO	\$52223:	
15363	051730					CMP	R2,R0
15364	051730					BGT	\$52224
15365	051730	020200				TST	R3
15366	051732	003027				BLOS	\$52224
15367	051734	005703					
15368	051736	101425			IF #BIT00 SET IN R2 THEN	BIT	#BIT00,R2
15369	051740					BEQ	\$52225
15370	051740	032702	000001		LET (R1) :B= (R1) SET.BY #100	BISB	#100,(R1)
15371	051744	001415					
15372	051746				IF R2 EQ #1 AND DATA02 HIS #3000 THEN	CMP	R2,#1
15373	051746	152711	000100			BNE	\$52226
15374	051752					CMP	DATA02,#3000
15375	051752	020227	000001			BLO	\$52226
15376	051756	001006					
15377	051760	023727	000566	003000	LET (R1) :B= (R1) SET.BY #20	:ROUNDING	
15378	051766	103402				BISB	#20,(R1)
15379	051770				ENDIF	\$52226:	
15380	051770	152711	000020		LET R1 := R1 - #1	DEC	R1
15381	051774						
15382	051774				ELSE	BR	\$52227
15383	051774					\$52225:	
15384	051774	005301			LET (R1) :B= (R1) SET.BY #4	BISB	#4,(R1)
15385	051776						
15386	051776	000402			ENDIF	\$52227:	
15387	052000				LET R2 := R2 + #1	INC	R2
15388	052000				LET R3 := R3 - #1	DEC	R3
15389	052000	152711	000004		ENDDO	BR	\$52223
15390	052004					\$52224:	
15391	052004				IF R3 EQ #0 AND R2 LE R0 THEN	TST	R3
15392	052004					BNE	\$52230
15393	052004	005202				CMP	R2,R0
15394	052006					BGT	\$52230
15395	052006	005303			LET CC.EXP := CC.EXP SET.BY #VBIT	BIS	#VBIT,CC.EXP
15396	052010						
15397	052010	000747			ENDIF		
15398	052012						
15399	052012						
15400	052012	005703					
15401	052014	001005					
15402	052016	020200					
15403	052020	003003					
15404	052022						
15405	052022	052737	000002	000522			
15406	052030						

15407 052030
15408 052030
15409 052030

ENDIF

\$52230:

\$52216:

15410	052030			
15411	052030	005737	000562	
15412	052034	001506		
15413	052036	005737	000564	
15414	05204	001503		
15415	052044	032737	000200	000566
15416	052052	001077		
15417				
15418				
15419				
15420				
15421	052054			
15422	052054	032737	000200	000566
15423	052062	001073		
15424	052064			
15425	052064	013737	000562	000500
15426	052072	063737	000476	000500
15427	052100			
15428	052100	023737	000500	000564
15429	052106	101403		
15430	052110			
15431	052110	052737	000002	000522
15432	052116			
15433	052116			
15434	052116			
15435	052116	023737	000476	000564
15436	052124	103407		
15437	052126			
15438	052126	052737	000004	000522
15439	052134			
15440	052134	042737	000010	000522
15441	052142			
15442	052142	000443		
15443	052144			
15444	052144			
15445	052144	013704	000476	
15446	052150	005204		
15447	052152			
15448	052152	006204		
15449	052154			
15450	052154	013701	000564	
15451	052160			
15452	052160	006201		
15453	052162			
15454	052162	062701	064442	
15455	052166	160401		
15456	052170			
15457	052170	013700	000476	
15458	052174			
15459	052174	013702	000562	
15460	052200			
15461	052200	013703	000564	
15462	052204	163703	000476	
15463	052210			
15464	052210			
15465	052210	005702		

IF DATA00 NE #0 AND DATA01 NE #0 AND #BIT07 NOTSETIN DATA02 THEN

```
TST DATA00
BEQ $52231
TST DATA01
BEQ $52231
BIT #BIT07,DATA02
BNE $52231
```

;TAKE CARE OF POSITIVE SHIFT

IF #BIT07 NOTSETIN DATA02 THEN

```
BIT #BIT07,DATA02
BNE $52232
;SRC.LEN + SHIFT COUNT
MOV DATA00,TEMP1
ADD TEMPO,TEMP1
```

LET TEMP1 := DATA00 + TEMPO

IF TEMP1 HI DATA01 THEN

```
CMP TEMP1,DATA01
BLOS $52233
```

LET CC.EXP := CC.EXP SET.BY #VBIT

```
BIS #VBIT,CC.EXP
```

ENDIF

\$52233:

IF TEMPO HIS DATA01 THEN

```
CMP TEMPO,DATA01
BLO $52234
```

LET CC.EXP := CC.EXP SET.BY #ZBIT

```
BIS #ZBIT,CC.EXP
```

LET CC.EXP := CC.EXP CLR.BY #NBIT

```
BIC #NBIT,CC.EXP
```

ELSE

```
BR $52235
```

LET R4 := TEMPO + #1

```
$52234:
;SHIFT COUNT + 1
MOV TEMPO,R4
INC R4
```

INLINE <ASR R4>

```
ASR R4
```

LET R1 := DATA01

```
;DST.LEN
MOV DATA01,R1
```

INLINE <ASR R1>

```
ASR R1
```

LET R1 := R1 + #DS.EXP - R4

```
ADD #DS.EXP,R1
SUB R4,R1
```

LET R0 := TEMPO

```
;SHIFT COUNT
MOV TEMPO,R0
```

LET R2 := DATA00

```
;SRC.LEN
MOV DATA00,R2
```

LET R3 := DATA01 - TEMPO

```
;DST.LEN - SHIFT COUNT
MOV DATA01,R3
SUB TEMPO,R3
```

WHILE R2 NE #0 AND R3 NE #0 DO

```
$52236:
TST R2
```

15466	052212	001417			BEQ	\$52237
15467	052214	005703			TST	R3
15468	052216	001415			BEQ	\$52237
15469	052220			IF #BIT00 NOTSET IN R0 THEN		
15470	052220	032700	000001		BIT	#BIT00,R0
15471	052224	001004			BNE	\$52240
15472	052226			LET (R1) :B= (R1) SET.BY #100		
15473	052226	152711	000100		BISB	#100,(R1)
15474	052232			LET R1 := R1 - #1		
15475	052232	005301			DEC	R1
15476	052234			ELSE		
15477	052234	000402			BR	\$52241
15478	052236				\$52240:	
15479	052236			LET (R1) :B= (R1) SET.BY #4		
15480	052236	152711	000004		BISB	#4,(R1)
15481	052242			ENDIF		
15482	052242				\$52241:	
15483	052242			LET R0 := COMP R0	;TOGGLE	SHIFT COUNT
15484	052242	005100			COM	R0
15485	052244			LET R2 := R2 - #1		
15486	052244	005302			DEC	R2
15487	052246			LET R3 := R3 - #1		
15488	052246	005303			DEC	R3
15489	052250			ENDDO		
15490	052250	000757			BR	\$52236
15491	052252				\$52237:	
15492	052252			ENDIF		
15493	052252				\$52235:	
15494	052252			ENDIF		
15495	052252				\$52232:	
15496	052252			ENDIF		
15497	052252				\$52231:	
15498						
15499						
15500				:+		
15501				: DO THE ASHP INSTRUCTION USING SIGNED PACKED (UNSIGNED		
15502				: PACKED WILL BE TESTED IN THE INLINE PORTION OF THE TEST).		
15503				: CHECK THE RESULTS AND REPORT ANY ERRORS.		
15504				: -		
15505	052252			LET R0 := DATA00 SET.BY #60000		
15506	052252	013700	000562		MOV	DATA00,R0
15507	052256	052700	060000		BIS	#60000,R0
15508	052262			LET R1 := #DSBUF1		
15509	052262	012701	064274		MOV	#DSBUF1,R1
15510	052266			LET R2 := DATA01 SET.BY #60000		
15511	052266	013702	000564		MOV	DATA01,R2
15512	052272	052702	060000		BIS	#60000,R2
15513	052276			LET R3 := #DS.DST		
15514	052276	012703	064400		MOV	#DS.DST,R3
15515	052302			LET R4 := DATA02		
15516	052302	013704	000566		MOV	DATA02,R4
15517	052306			LET R5 := #0		
15518	052306	005005			CLR	R5
15519	052310			INLINE <ASHP>		
15520	052310	076076			ASHP	
15521				;CHECK RESULTS		

15522	052312				LET CC := PSW		
15523	052312	013737	177776	000520		MOV	PSW,CC
15524	052320				LET CC := CC CLR.BY #177760		
15525	052320	042737	177760	000520		BIC	#177760,CC
15526	052326				IF CC NE CC.EXP THEN		
15527	052326	023737	000520	000522		CMP	CC,CC.EXP
15528	052334	001402				BEQ	\$52242
15529	052336				CALL ERROR		
15530	052336	004737	060044			JSR	PC,ERROR
15531	052342				ENDIF		
15532	052342					\$52242:	
15533	052342				IF R0 NE #0 OR R1 NE #0 THEN		
15534	052342	005700				TST	R0
15535	052344	001002				BNE	\$52243
15536	052346	005701				TST	R1
15537	052350	001402				BEQ	\$52244
15538	052352					\$52243:	
15539	052352				CALL ERROR		
15540	052352	004737	060044			JSR	PC,ERROR
15541	052356				ENDIF		
15542	052356					\$52244:	
15543	052356				IF R4 NE #0 OR R5 NE #0 THEN		
15544	052356	005704				TST	R4
15545	052360	001002				BNE	\$52245
15546	052362	005705				TST	R5
15547	052364	001402				BEQ	\$52246
15548	052366					\$52245:	
15549	052366				CALL ERROR		
15550	052366	004737	060044			JSR	PC,ERROR
15551	052372				ENDIF		
15552	052372					\$52246:	
15553	052372				LET TEMPO :- DATA01 SET.BY #60000		
15554	052372	013737	000564	000476		MOV	DATA01,TEMPO
15555	052400	052737	060000	000476		BIS	#60000,TEMPO
15556	052406				IF R2 NE TEMPO THEN		
15557	052406	020237	000476			CMP	R2,TEMPO
15558	052412	001402				BEQ	\$52247
15559	052414				CALL ERROR		
15560	052414	004737	060044			JSR	PC,ERROR
15561	052420				ENDIF		
15562	052420					\$52247:	
15563	052420				IF R3 NE #DS.DST THEN		
15564	052420	020327	064400			CMP	R3,#DS.DST
15565	052424	001402				BEQ	\$52250
15566	052426				CALL ERROR		
15567	052426	004737	060044			JSR	PC,ERROR
15568	052432				ENDIF		
15569	052432					\$52250:	
15570					;SEE IF DS.DST = DS.EXP		
15571							
15572	052432				LET R0 :- #40		
15573	052432	012700	000040			MOV	#40,R0
15574	052436				LET R1 := #DS.EXP		
15575	052436	012701	064442			MOV	#DS.EXP,R1
15576	052442				LET R2 := #DS.DST		
15577	052442	012702	064400			MOV	#DS.DST,R2

15578	052446			LET ERRFLG := #0		CLR	ERRFLG
15579	052446	005037	000456				
15580	052452			WHILE RO HI #0 AND ERRFLG EQ #0 DO			
15581	052452					\$52251:	
15582	052452	005700				TST	RO
15583	052454	101412				BLOS	\$52252
15584	052456	005737	000456			TST	ERRFLG
15585	052462	001007				BNE	\$52252
15586	052464			IFB (R1)+ NE (R2)+ THEN			
15587	052464	122122				CMPB	(R1)+ (R2)+
15588	052466	001403				BEQ	\$52253
15589	052470			LET ERRFLG := #-1			
15590	052470	012737	177777	000456		MOV	#-1,ERRFLG
15591	052476			ENDIF			
15592	052476					\$52253:	
15593	052476			LET RO := RO - #1			
15594	052476	005300				DEC	RO
15595	052500			ENDDO			
15596	052500	000764				BR	\$52251
15597	052502					\$52252:	
15598	052502			IF ERRFLG NE #0 THEN			
15599	052502	005737	000456			TST	ERRFLG
15600	052506	001402				BEQ	\$52254
15601	052510			CALL ERROR		;DS.DST AND DS.EXP DO NOT MATCH	
15602	052510	004737	060044			JSR	PC,ERROR
15603	052514			ENDIF			
15604	052514					\$52254:	
15605	052514			LET SIGNO : COMP SIGNO		;REPEAT THE TEST CASE WITH A NEGATIVE	
15606	052514	005137	000450			COM	SIGNO
15607						; SIGN BEFORE MOVING ON TO THE NEXT TEST CASE	
15608	052520			IF SIGNO EQ #0 THEN			
15609	052520	005737	000450			TST	SIGNO
15610	052524	001005				BNE	\$52255
15611	052526			LET TSTCAS := TSTCAS + #6			
15612	052526	062737	000006	000544		ADD	#6,TSTCAS
15613	052534			LET COUNT := COUNT - #1			
15614	052534	005337	000470			DEC	COUNT
15615	052540			ENDIF			
15616	052540					\$52255:	
15617				:UNTIL COUNT EQ #0			
15618	052540			IF COUNT NE #0 THEN			
15619	052540	005737	000470			TST	COUNT
15620	052544	001402				BEQ	\$52256
15621	052546			INLINE <JMP \$5076>			
15622	052546	000137	051240			JMP	\$5076
15623	052552			ENDIF			
15624	052552					\$52256:	
15625							
15626							
15627							
15628				:+			
15629				: DO ONE INLINE CASE - ASHPI. USE UNSIGNED PACKED.			
15630				:-			
15631							
15632	052552			CALL CLRDSB			
15633	052552	004737	057040			JSR	PC,CLRDSB

15634	052556			LET DSBUF1 :B= #231	:9/9		
15635	052556	112737	000231	064274		MOVB	#231,DSBUF1
15636	052564			LET DSBUF1+1 :B= #231	:9/9		
15637	052564	112737	000231	064275		MOVB	#231,DSBUF1+1
15638	052572			LET DSBUF1+2 :B= #237	:9/17		
15639	052572	112737	000237	064276		MOVB	#237,DSBUF1+2
15640	052600			CALL CLRREG			
15641	052600	004737	056604			JSR	PC,CLRREG
15642	052604			INLINE <ASHP1>			
15643	052604	076176				ASHP1	
15644	052606			INLINE <.WORD TASHPI>			
15645	052606	052776				.WORD	TASHPI
15646	052610			INLINE <.WORD TASHPI+4>			
15647	052610	053002				.WORD	TASHPI+4
15648	052612			INLINE <.WORD 375>		;SHIFT COUNT = -3, RND	DIGIT = 0
15649	052612	000375				.WORD	375
15650	052614			LET CC := PSW			
15651	052614	013737	177776	000520		MOV	PSW,CC
15652	052622			LET CC := CC CLR.BY #177760			
15653	052622	042737	177760	000520		BIC	#177760,CC
15654	052630			IF CC NE #0 THEN		;VBIT SHOULD BE	
15655	052630	005737	000520			SET	
15656	052634	001402				TST	CC
15657	052636			CALL ERROR		BEQ	\$52257
15658	052636	004737	060044			JSR	PC,ERROR
15659	052642			ENDIF			
15660	052642					\$52257:	
15661	052642			IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN			
15662	052642	005700				TST	R0
15663	052644	001004				BNE	\$52260
15664	052646	005701				TST	R1
15665	052650	001002				BNE	\$52260
15666	052652	005702				TST	R2
15667	052654	001402				BEQ	\$52261
15668	052656					\$52260:	
15669	052656			CALL ERROR			
15670	052656	004737	060044			JSR	PC,ERROR
15671	052662			ENDIF			
15672	052662					\$52261:	
15673	052662			IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN			
15674	052662	005703				TST	R3
15675	052664	001004				BNE	\$52262
15676	052666	005704				TST	R4
15677	052670	001002				BNE	\$52262
15678	052672	005705				TST	R5
15679	052674	001402				BEQ	\$52263
15680	052676					\$52262:	
15681	052676			CALL ERROR			
15682	052676	004737	060044			JSR	PC,ERROR
15683	052702			ENDIF			
15684	052702					\$52263:	
15685				;CHECK DS.DST			
15686	052702			LET R0 := #DS.DST			
15687	052702	012700	064400			MOV	#DS.DST,R0
15688	052706			IFB (R0)+ NE #11 THEN		;FIRST BYTE = 0/9	
15689	052706	122027	000011			CMPB	(R0)+,#11

15690	052712	001402				BEQ	\$52264
15691	052714			CALL ERROR			
15692	052714	004737	060044			JSR	PC,ERROR
15693	052720			ENDIF			
15694	052720					\$52264:	
15695	052720			IFB (R0)+ NE #237 THEN		;SECOND BYTE = 9/17	
15696	052720	122027	000237			CMPB	(R0)+,#237
15697	052724	001402				BEQ	\$52265
15698	052726			CALL ERROR			
15699	052726	004737	060044			JSR	PC,ERROR
15700	052732			ENDIF			
15701	052732					\$52265:	
15702	052732			LET ERRFLG := #0			
15703	052732	005037	000456			CLR	ERRFLG
15704	052736			WHILE ERRFLG NE #0 AND R0 NE #DS.DST+37 DO			
15705	052736					\$52266:	
15706	052736	005737	000456			TST	ERRFLG
15707	052742	001413				BEQ	\$52267
15708	052744	020027	064437			CMP	R0,#DS.DST+37
15709	052750	001410				BEQ	\$52267
15710	052752			IFB (R0)+ NE #0 THEN			
15711	052752	105720				TSTB	(R0)+
15712	052754	001405				BEQ	\$52270
15713	052756			LET ERRFLG := #-1			
15714	052756	012737	177777	000456		MOV	#-1,ERRFLG
15715	052764			CALL ERROR			
15716	052764	004737	060044			JSR	PC,ERROR
15717	052770			ENDIF			
15718	052770					\$52270:	
15719	052770			ENDDO			
15720	052770	000762				BR	\$52266
15721	052772					\$52267:	
15722	052772			INLINE <JMP CIS7END>			
15723	052772	000137	053006			JMP	CIS7END
15724							
15725							
15726							
15727							
15728				:+			
15729				: INLINE TEST CASE			
15730				:-			
15731							
15732	052776	070005		TASHPI: .WORD 070005		:SRC.DSCR, UNSIGNED PACKED, SRC.LEN = 5	
15733	053000	064274		.WORD DSBUF1		:SRC.DSCR	
15734	053002	070002		.WORD 070002		:DST.DSCR, UNSIGNED PACKED, DST.LEN = 2	
15735	053004	064400		.WORD DS.DST		:DST.DSCR	
15736							
15737	053006			INLINE <CIS7END:>			
15738	053006					CIS7END:	
15739							
15740							

```

15741 :*****
15742 :*****
15743 :++
15744 :          C I S 8      M O D U L E
15745 :
15746 :   THIS MODULE TESTS THE PACKED MULTIPLY INSTRUCTION: MULP.
15747 :
15748 :-----
15749 :*****
15750 :*****
15751 :
15752 :
15753 :
15754 :
15755 :.SBTTL  CIS8 INITIALIZATION
15756 :*****
15757 :
15758 :   CIS8 INITIALIZATION
15759 :
15760 :*****
15761 :
15762 :;IF SWR<5:0> ARE CLEARED OR IF SWR<4> IS SET THEN EXECUTE
15763 :; THE CIS8 MODULE; OTHERWISE, SKIP TO THE NEXT CIS MODULE.
15764 :
15765 053006      INLINE  <SCOPE>
15766 053006 000004      SCOPE
15767
15768 053010      LET TEMPO := @SWR CLR.BY #177700
15769 053010 017737 125412 000476      MOV @SWR,TEMPO
15770 053016 042737 177700 000476      BIC #177700,TEMPO
15771 053024      IF TEMPO NE #0 THEN
15772 053024 005737 000476      TST TEMPO
15773 053030 001406      BEQ $52271
15774 053032      IF #BIT4 NOTSETIN TEMPO THEN
15775 053032 032737 00002C 000476      BIT #BIT4,TEMPO
15776 053040 001002      BNE $52272
15777 053042      INLINE  <JMP  CIS8END>
15778 053042 000137 054222      JMP  CIS8END
15779 053046      ENDIF
15780 053046      $52272:
15781 053046      ENDIF
15782 053046      $52271:
15783 053046      LET CTL := #8.          ;CONTROL CHIP NUMBER
15784 053046 012737 000010 000546      MOV #8.,CTL
15785 053054      LET ERRFLG := #0      ;INITIALIZE ERROR FLAG
15786 053054 005037 000456      CLR  ERRFLG
15787 053060      LET PSW := #340
15788 053060 012737 000340 177776      MOV #340,PSW
15789 053066      LET ERR := #1
15790 053066 012737 000001 000552      MOV #1,ERR
    
```

```

15791
15792      .SBTTL  MULP
15793      :*****
15794      :+
15795      :      MULP TEST
15796      :-
15797      :*****
15798
15799      053074      INLINE <XMULP:>
15800      053074
15801      053074
15802      053074      000004
15803      053076
15804      053076      012737      000020      000404
15805      053104
15806      053104      013737      055462      000470
15807      053112
15808      053112      012737      055464      000544
15809
15810      :+
15811      :      CLR R0.EXP TO R3.EXP
15812      :-
15813
15814      053120      INCR  R0 FROM #0 TO #6 BY #2
15815      053120      005000
15816      053122      000402
15817      053124
15818      053124      062700      000002
15819      053130
15820      053130      020027      000006
15821      053134      003003
15822      053136      INLINE <CLR R0.EXP(R0)>
15823      053136      005060      000524
15824      053142      ENDINC
15825      053142      000770
15826      053144
15827
15828      053144      ;REPEAT OUTER LOOP FOR TEST CASES
15829      053144      INLINE <$5074:>
15830      053144
15831      053144      004737      057040
15832
15833      :+
15834      :      LOAD DATAXX AND CC.EXP FROM TEST CASE
15835      :-
15836
15837      053150      LET R0 := TSTCAS
15838      053150      013700      000544
15839      053154
15840      053154      012037      000562
15841      053160
15842      053160      012037      000564
15843      053164
15844      053164      012037      000566
15845      053170
15846      053170      012037      000570
    
```

```

XMULP:
SCOPE
;ITERATION COUNT
MOV #20,$TIMES
;# OF TEST CASES
MOV TAB89,COUNT
;POINT TO FIRST TEST CASE
MOV #TAB89+2,TSTCAS

CLR R0
BR $52273
$52274:
ADD #2,R0
$52273:
CMP R0,#6
BGT $52275

CLR R0.EXP(R0)
BR $52274
$52275:

$5074:
JSR PC,CLRDSB

MOV TSTCAS,R0
MOV (R0)+,DATA00
MOV (R0)+,DATA01
MOV (R0)+,DATA02
MOV (R0)+,DATA03
    
```

```

15847 053174          LET DATA04 := (R0)+          ;DEST LENGTH
15848 053174 012037 000572          ;DEST LENGTH
15849 053200          LET DATA05 := (R0)+          ;POINTER TO DS.EXP DATA (PRODUCT)
15850 053200 012037 000574          ;POINTER TO DS.EXP DATA (PRODUCT)
15851 053204          LET CC.EXP := (R0)+          ;CC.EXP (PRODUCT)
15852 053204 012037 000522          ;CC.EXP (PRODUCT)
15853
15854          ;+
15855          ;-          PACK BUFFERS FOR SRC1, SRC2, & DEST
15856          ;-
15857
15858 053210          LET R5 := #CHRSET          ;R5 IS OPERAND STACK FOR ROUTINE
15859 053210 012705 063672          ;R5 IS OPERAND STACK FOR ROUTINE
15860 053214          CALL PACK IN <#DSBUF1, DATA01, DATA00> ;SRC1
15861 053214 010546          ;SRC1
15862 053216 010605          ;SRC1
15863 053220 162706 000006          ;SRC1
15864 053224 013745 000562          ;SRC1
15865 053230 013745 000564          ;SRC1
15866 053234 012745 064274          ;SRC1
15867 053240 004737 057420          ;SRC1
15868 053244 062706 000006          ;SRC1
15869 053250 012605          ;SRC1
15870 053252          CALL PACK IN <#DSBUF2, DATA03, DATA02> ;SRC2
15871 053252 010546          ;SRC2
15872 053254 010605          ;SRC2
15873 053256 162706 000006          ;SRC2
15874 053262 013745 000566          ;SRC2
15875 053266 013745 000570          ;SRC2
15876 053272 012745 064336          ;SRC2
15877 053276 004737 057420          ;SRC2
15878 053302 062706 000006          ;SRC2
15879 053306 012605          ;SRC2
15880 053310          CALL PACK IN <#DS.EXP, DATA05, DATA04> ;DEST
15881 053310 010546          ;DEST
15882 053312 010605          ;DEST
15883 053314 162706 000006          ;DEST
15884 053320 013745 000572          ;DEST
15885 053324 013745 000574          ;DEST
15886 053330 012745 064442          ;DEST
15887 053334 004737 057420          ;DEST
15888 053340 062706 000006          ;DEST
15889 053344 012605          ;DEST
15890
15891          ;+
15892          ;-          LOAD GPR'S FROM DATAXX & ADD DATA TYPE
15893          ;-
15894
15895 053346          LET R0 := DATA00 SET.BY #60000
15896 053346 013700 000562          ;DEST LENGTH
15897 053352 052700 060000          ;DEST LENGTH
15898 053356          LET R1 := #DSBUF1
15899 053356 012701 064274          ;DEST LENGTH
15900 053362          LET R2 := DATA02 SET.BY #60000
15901 053362 013702 000566          ;DEST LENGTH
15902 053366 052702 060000          ;DEST LENGTH

```

15903	053372			LET R3 := #DSBUF2	
15904	053372	012703	064336		MOV #DSBUF2,R3
15905	053376			LET R4 := DATA04 SET.BY #60000	
15906	053376	013704	000572		MOV DATA04,R4
15907	053402	052704	060000		BIS #60000,R4
15908	053406			LET R5 :- #DS.DST	
15909	053406	012705	064400		MOV #DS.DST,R5
15910					
15911				:+	
15912				FINISH SET UP & DO CIS INSTRUCTION	
15913				:-	
15914					
15915					
15916	053412			LET R4.EXP := R4	
15917	053412	010437	000534		MOV R4,R4.EXP
15918	053416			LET R5.EXP := R5	
15919	053416	010537	000536		MOV R5,R5.EXP
15920	053422			INLINE <MULP>	
15921	053422	076074			MULP
15922					
15923				:+	
15924				CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.	
15925				:-	
15926					
15927	053424			LET CC := PSW	
15928	053424	013737	177776 000520		MOV PSW,CC
15929	053432			LET CC := CC CLR.BY #177760	
15930	053432	042737	177760 000520		BIC #177760,CC
15931	053440			IF CC NE CC.EXP THEN	
15932	053440	023737	000520 000522		CMP CC,CC.EXP
15933	053446	001402			BEQ \$52276
15934	053450			CALL ERROR	
15935	053450	004737	060044		JSR PC,ERROR
15936	053454			ENDIF	
15937	053454				\$52276:
15938	053454			IF R0 NE R0.EXP THEN	
15939	053454	020037	000524		CMP R0,R0.EXP
15940	053460	001402			BEQ \$52277
15941	053462			CALL ERROR	
15942	053462	004737	060044		JSR PC,ERROR
15943	053466			ENDIF	
15944	053466				\$52277:
15945	053466			IF R1 NE R1.EXP THEN	
15946	053466	020137	000526		CMP R1,R1.EXP
15947	053472	001402			BEQ \$52300
15948	053474			CALL ERROR	
15949	053474	004737	060044		JSR PC,ERROR
15950	053500			ENDIF	
15951	053500				\$52300:
15952	053500			IF R2 NE R2.EXP THEN	
15953	053500	020237	000530		CMP R2,R2.EXP
15954	053504	001402			BEQ \$52301
15955	053506			CALL ERROR	
15956	053506	004737	060044		JSR PC,ERROR
15957	053512			ENDIF	
15958	053512				\$52301:

15959	053512			IF R3 NE R3.EXP THEN		
15960	053512	020337	000532		CMP	R3,R3.EXP
15961	053516	001402			BEQ	\$52302
15962	053520			CALL ERROR		
15963	053520	004737	060044		JSR	PC,ERROR
15964	053524			ENDIF		
15965	053524				\$52302:	
15966	053524			IF R4 NE R4.EXP THEN		
15967	053524	020437	000534		CMP	R4,R4.EXP
15968	053530	001402			BEQ	\$52303
15969	053532			CALL ERROR		
15970	053532	004737	060044		JSR	PC,ERROR
15971	053536			ENDIF		
15972	053536				\$52303:	
15973	053536			IF R5 NE R5.EXP THEN		
15974	053536	020537	000536		CMP	R5,R5.EXP
15975	053542	001402			BEQ	\$52304
15976	053544			CALL ERROR		
15977	053544	004737	060044		JSR	PC,ERROR
15978	053550			ENDIF		
15979	053550				\$52304:	
15980						
15981						
15982				:+		
15983				: :	CHECK RESULTS VS EXPECTED RESULTS	
15984				:-		
15985						
15986	053550			LET R0 := #DS.DST		
15987	053550	012700	064400		MOV	#DS.DST,R0
15988	053554			LET R1 := #DS.EXP		
15989	053554	012701	064442		MOV	#DS.EXP,R1
15990	053560			LET ERRFLG := #0		
15991	053560	005037	000456		CLR	ERRFLG
15992	053564			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO		
15993	053564				\$52305:	
15994	053564	020027	064440		CMP	R0,#DS.DST+40
15995	053570	001410			BEQ	\$52306
15996	053572	005737	000456		TST	ERRFLG
15997	053576	001005			BNE	\$52306
15998	053600			IFB (R0)+ NE (R1)+ THEN		
15999	053600	122021			CMPB	(R0)+,(R1)+
16000	053602	001402			BEQ	\$52307
16001	053604			CALL ERROR		
16002	053604	004737	060044		JSR	PC,ERROR
16003	053610			ENDIF		
16004	053610				\$52307:	
16005	053610			ENDDO		
16006	053610	000765			BR	\$52305
16007	053612				\$52306:	
16008	053612			LET TSTCAS := TSTCAS + #22		
16009	053612	062737	000022 000544		ADD	#22,TSTCAS
16010	053620			LET COUNT := COUNT - #1		
16011	053620	005337	000470		DEC	COUNT
16012				:UNTIL COUNT EQ #0		
16013	053624			IF COUNT NE #0 THEN		
16014	053624	005737	000470		TST	COUNT

16015	053630	001402			BEQ	\$52310
16016	053632			INLINE <JMP \$5074>		
16017	053632	000137	053144		JMP	\$5074
16018	053636			ENDIF		
16019	053636				\$52310:	
16020						
16021				:+		
16022				:+ INLINE CASE - MULPI, USING UNSIGNED PACKED DATA		
16023				:+		
16024				:+		
16025	053636			CALL CLRDSB		
16026	053636	004737	057040		JSR	PC,CLRDSB
16027						
16028				:+		
16029				:+ LOAD DATAXX AND CC.EXP FROM TEST CASE		
16030				:+		
16031				:+		
16032	053642			LET R0 := #TAB89+2		
16033	053642	012700	055464		MOV	#TAB89+2,R0
16034	053646			LET DATA00 := (R0)+		
16035	053646	012037	000562		MOV	(R0)+,DATA00
16036	053652			LET DATA01 := (R0)+		
16037	053652	012037	000564		MOV	(R0)+,DATA01
16038	053656			LET DATA02 := (R0)+		
16039	053656	012037	000566		MOV	(R0)+,DATA02
16040	053662			LET DATA03 := (R0)+		
16041	053662	012037	000570		MOV	(R0)+,DATA03
16042	053666			LET DATA04 := (R0)+		
16043	053666	012037	000572		MOV	(R0)+,DATA04
16044	053672			LET DATA05 := (R0)+		
16045	053672	012037	000574		MOV	(R0)+,DATA05
16046						
16047				:+		
16048				:+ PACK BUFFERS FOR SRC1, SRC2, & DEST		
16049				:+		
16050				:+		
16051	053676			LET R5 := #CHRSET		
16052	053676	012705	063672		MOV	#CHRSET,R5
16053	053702			CALL PACK IN <#DSBUF1,DATA01,DATA00>		
16054	053702	010546			MOV	R5,-(SP)
16055	053704	010605			MOV	SP,R5
16056	053706	162706	000006		SUB	#6,SP
16057	053712	013745	000562		MOV	DATA00,-(R5)
16058	053716	013745	000564		MOV	DATA01,-(R5)
16059	053722	012745	064274		MOV	#DSBUF1,-(R5)
16060	053726	004737	057420		JSR	PC,PACK
16061	053732	062706	000006		ADD	#6,SP
16062	053736	012605			MOV	(SP)+,R5
16063	053740			CALL PACK IN <#DSBUF2,DATA03,DATA02>		
16064	053740	010546			MOV	R5,-(SP)
16065	053742	010605			MOV	SP,R5
16066	053744	162706	000006		SUB	#6,SP
16067	053750	013745	000566		MOV	DATA02,-(R5)
16068	053754	013745	000570		MOV	DATA03,-(R5)
16069	053760	012745	064336		MOV	#DSBUF2,-(R5)
16070	053764	004737	057420		JSR	PC,PACK

16071	053770	062706	000006			ADD	#6,SP
16072	053774	012605				MOV	(SP)+,R5
16073	053776			CALL PACK IN <#DS.EXP,DATA05,DATA04>	;DEST		
16074	053776	010546				MOV	R5,-(SP)
16075	054000	010605				MOV	SP,R5
16076	054002	162706	000006			SUB	#6,SP
16077	054006	013745	000572			MOV	DATA04,-(R5)
16078	054012	013745	000574			MOV	DATA05,-(R5)
16079	054016	012745	064442			MOV	#DS.EXP,-(R5)
16080	054022	004737	057420			JSR	PC,PACK
16081	054026	062706	000006			ADD	#6,SP
16082	054032	012605				MOV	(SP)+,R5
16083	054034			CALL CLRREG	;ENSURE GPR'S 0-5 ARE ZERO	JSR	PC,CLRREG
16084	054034	004737	056604				
16085							
16086	054040			INLINE <MULPI>			
16087	054040	076174				MULPI	
16088	054042			INLINE <.WORD TMULPI>			
16089	054042	056264				.WORD TMULPI	
16090	054044			INLINE <.WORD TMULPI+4>			
16091	054044	056270				.WORD TMULPI+4	
16092	054046			INLINE <.WORD TMULPI+10>			
16093	054046	056274				.WORD TMULPI+10	
16094	054050			LET CC := PSW			
16095	054050	013737	177776	000520		MOV	PSW,CC
16096	054056			LET CC := CC CLR.BY #177760			
16097	054056	042737	177760	000520		BIC	#177760,CC
16098	054064			IF CC NE #0 THEN			
16099	054064	005737	000520			TST	CC
16100	054070	001402				BEQ	\$52311
16101	054072			CALL ERROR		JSR	PC,ERROR
16102	054072	004737	060044				
16103	054076			ENDIF			
16104	054076					\$52311:	
16105							
16106				:+			
16107				:-	ENSURE GPR'S STILL ZERO		
16108							
16109							
16110	054076			IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN			
16111	054076	005700				TST	R0
16112	054100	001004				BNE	\$52312
16113	054102	005701				TST	R1
16114	054104	001002				BNE	\$52312
16115	054106	005702				TST	R2
16116	054110	001402				BEQ	\$52313
16117	054112					\$52312:	
16118	054112			CALL ERROR			
16119	054112	004737	060044			JSR	PC,ERROR
16120	054116			ENDIF			
16121	054116					\$52313:	
16122	054116			IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN			
16123	054116	005703				TST	R3
16124	054120	001004				BNE	\$52314
16125	054122	005704				TST	R4
16126	054124	001002				BNE	\$52314

16127	054126	005705				TST	R5
16128	054130	001402				BEQ	\$52315
16129	054132					\$52314:	
16130	054132			CALL ERROR			
16131	054132	004737	060044			JSR	PC,ERROR
16132	054136			ENDIF			
16133	054136					\$52315:	
16134							
16135				:+			
16136				:+ CHECK RESULTS			
16137				:+			
16138				:-			
16139	054136			LET R0 := #DS.DST			
16140	054136	012700	064400			MOV	#DS.DST,R0
16141	054142			LET R1 := #DS.EXP	;EXPECTED DATA		
16142	054142	012701	064442			MOV	#DS.EXP,R1
16143	054146			LET R2 := TMULPI+10 CLR.BY #70000	;GET STRING LENGTH		
16144	054146	013702	056274			MOV	TMULPI+10,R2
16145	054152	042702	070000			BIC	#70000,R2
16146	054156			INLINE <ASR R2>	;LENGTH/2 (PACKING FACTOR)		
16147	054156	006202				ASR	R2
16148	054160			LET R2 := R2 + #DS.EXP	;R2 POINTS TO MSD/SIGN		
16149	054160	062702	064442			ADD	#DS.EXP,R2
16150	054164			LET (R2) :B= (R2) SET.BY #17	;MAKE IT UNSIGNED		
16151	054164	152712	000017			BISB	#17,(R2)
16152	054170			LET ERRFLG := #0			
16153	054170	005037	000456			CLR	ERRFLG
16154	054174			WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO			
16155	054174					\$52316:	
16156	054174	020027	064437			CMPL	R0,#DS.DST+37
16157	054200	001410				BEQ	\$52317
16158	054202	005737	000456			TST	ERRFLG
16159	054206	001005				BNE	\$52317
16160	054210			IFB (R0)+ NE (R1)+ THEN			
16161	054210	122021				CMPL	(R0)+,(R1)+
16162	054212	001402				BEQ	\$52320
16163	054214			CALL ERROR			
16164	054214	004737	060044			JSR	PC,ERROR
16165	054220			ENDIF			
16166	054220					\$52320:	
16167	054220			ENDDO			
16168	054220	000765				BR	\$52316
16169	054222					\$52317:	
16170							
16171	054222			INLINE <CISBEND:>			
16172	054222					CISBEND:	
16173							

16174
 16175
 16176
 16177
 16178
 16179
 16180
 16181
 16182
 16183
 16184
 16185
 16186
 16187
 16188
 16189
 16190
 16191
 16192
 16193
 16194
 16195
 16196
 16197
 16198
 16199
 16200
 16201
 16202
 16203
 16204
 16205
 16206
 16207
 16208
 16209
 16210
 16211
 16212
 16213
 16214
 16215
 16216
 16217
 16218
 16219
 16220
 16221
 16222
 16223
 16224

```

*****
*****
:++
:          C I S 9   M O D U L E
:
:  THIS MODULE TESTS THE PACKED DIVIDE INSTRUCTION: DIVP.
:--
*****
*****
    
```

```

.SBTTL CIS9 INITIALIZATION
*****
:
:  CIS9 INITIALIZATION
:
*****
    
```

```

;IF SWR<5:0> ARE CLEARED OR IF SWR<5> IS SET THEN EXECUTE
; THE CIS9 MODULE; OTHERWISE, SKIP TO END OF PASS.
    
```

```

INLINE <SCOPE>
                                SCU
LET TEMPO := @SWR CLR.BY #177700
                                MOV    @SWR,TEMPO
                                BIC    #177700,TEMPO
IF TEMPO NE #0 THEN
                                TST    TEMPO
                                BEQ    $52321
    IF #BITS NOTSET IN TEMPO THEN
                                BIT    #BITS,TEMPO
                                BNE    $52322
                                INLINE <JMP CIS9END>
                                JMP    CIS9END
    ENDIF
ENDIF
                                $52322:
                                $52321:
LET CTL : #9.                   ;CONTROL CHIP NUMBER
                                MOV    #9.,CTL
LET ERRFLG :- #0                ;INITIALIZE ERROR FLAG
                                CLR    ERRFLG
LET PSW :- #340
                                MOV    #340,PSW
LET ERR :- #1
                                MOV    #1,ERR
    
```

```

054222
054222 000004
054224
054224 017737 124176 000476
054232 042737 177700 000476
054240
054240 005737 000476
054244 001406
054246
054246 032737 000040 000476
054254 001002
054256
054256 000137 056300
054262
054262
054262
054262
054262
054262 012737 000011 000546
054270
054270 005037 000456
054274
054274 012737 000340 177776
054302
054302 012737 000001 000552
    
```

```

16225 .SBTTL DIVP
16226 :*****
16227 :+
16228 :   DIVP TEST
16229 :-
16230 :*****
16231
16232 054310 INLINE <XDIVP:>                                XDIVP:
16233 054310
16234 054310 INLINE <SCOPE>                                SCOPE
16235 054310 00C004
16236 054312 LET $TIMES := #20                        ;ITERATION COUNT
16237 054312 012737 000020 000404 MOV #20,$TIMES
16238 054320 LET COUNT := TAB89                      ;# OF TEST CASES
16239 054320 013737 055462 000470 MOV TAB89,COUNT
16240 054326 LET TSTCAS := #TAB89+2                ;POINT TO FIRST TEST CASE
16241 054326 012737 055464 000544 MOV #TAB89+2,TSTCAS
16242
16243 :+
16244 :   CLR R0.EXT TO R3.EXP
16245 :-
16246
16247 054334 INCR R0 FROM #0 TO #6 BY #2
16248 054334 005000 CLR R0
16249 054336 000402 BR $52323
16250 054340 ADD #2,R0                                $52324:
16251 054340 062700 000002 $52323:
16252 054344 CMP R0,#6
16253 054344 020027 000006 BGT $52325
16254 054350
16255 054352 INLINE <CLR R0.EXP(R0)>
16256 054352 005060 000524 CLR R0.EXP(R0)
16257 054356 ENDINC
16258 054356 000770 BR $52324
16259 054360 $52325:
16260 ;REPEAT OUTER LOOP FOR TEST CASES
16261 054360 INLINE <$5075:>
16262 054360 $5075:
16263 054360 CALL CLRDSB ;CLR DS BUFFERS
16264 054360 004737 057040 JSR PC,CLRDSB
16265
16266 :+
16267 :   LOAD DATAXX AND CC.EXP FROM TEST CASE
16268 :-
16269
16270 054364 LET R0 := TSTCAS
16271 054364 013700 000544 MOV TSTCAS,R0
16272 054370 LET DATA00 := (R0)+ ;SRC1 LFNGTH
16273 054370 012037 000562 MOV (R0)+,DATA00
16274 054374 LET DATA01 := (R0)+ ;POINTER TO SRC1 DATA
16275 054374 012037 000564 MOV (R0)+,DATA01
16276 054400 LET DATA02 := (R0)+ ;SRC2 LENGTH
16277 054400 012037 000566 MOV (R0)+,DATA02
16278 054404 LET DATA03 := (R0)+ ;POINTER TO SRC2 DATA
16279 054404 012037 000570 MOV (R0)+,DATA03
16280 054410 LET DATA04 := (R0)+ ;DEST LENGTH
    
```

16281	054410	012037	000572			MOV	(R0)+,DATA04
16282	054414			LET DATA05 := (R0)+	;POINTER TO DS.EXP DATA	(PRODUCT)	
16283	054414	012037	000574			MOV	(R0)+,DATA05
16284	054420			LET CC.EXP := (R0)+	;CC.EXP (PRODUCT)		
16285	054420	012037	000522			MOV	(R0)+,CC.EXP
16286	054424			LET DATA05 := (R0)+	;POINTER TO DS.EXP DATA	(QUOTIENT)	
16287	054424	012037	000574			MOV	(R0)+,DATA05
16288	054430			LET CC.EXP := (R0)+	;CC.EXP (QUOTIENT)		
16289	054430	012037	000522			MOV	(R0)+,CC.EXP
16290							
16291				:+			
16292				:	PACK BUFFERS FOR SRC1, SRC2, & DEST		
16293				:-			
16294							
16295	054434			LET R5 := #CHRSET	;R5 IS OPERAND STACK FOR ROUTINE		
16296	054434	012705	063672			MOV	#CHRSET,R5
16297	054440			CALL PACK IN <#DSBUF1,DATA01,DATA00>	;SRC1		
16298	054440	010546				MOV	R5,-(SP)
16299	054442	010605				MOV	SP,R5
16300	054444	162706	000006			SUB	#6,SP
16301	054450	013745	000562			MOV	DATA00,-(R5)
16302	054454	013745	000564			MOV	DATA01,-(R5)
16303	054460	012745	064274			MOV	#DSBUF1,-(R5)
16304	054464	004737	057420			JSR	PC,PACK
16305	054470	062706	000006			ADD	#6,SP
16306	054474	012605				MOV	(SP)+,R5
16307	054476			CALL PACK IN <#DSBUF2,DATA03,DATA02>	;SRC2		
16308	054476	010546				MOV	R5,-(SP)
16309	054500	010605				MOV	SP,R5
16310	054502	162706	000006			SUB	#6,SP
16311	054506	013745	000566			MOV	DATA02,-(R5)
16312	054512	013745	000570			MOV	DATA03,-(R5)
16313	054516	012745	064336			MOV	#DSBUF2,-(R5)
16314	054522	004737	057420			JSR	PC,PACK
16315	054526	062706	000006			ADD	#6,SP
16316	054532	012605				MOV	(SP)+,R5
16317	054534			CALL PACK IN <#DS.EXP,DATA05,DATA04>	;DEST		
16318	054534	010546				MOV	R5,-(SP)
16319	054536	010605				MOV	SP,R5
16320	054540	162706	000006			SUB	#6,SP
16321	054544	013745	000572			MOV	DATA04,-(R5)
16322	054550	013745	000574			MOV	DATA05,-(R5)
16323	054554	012745	064442			MOV	#DS.EXP,-(R5)
16324	054560	004737	057420			JSR	PC,PACK
16325	054564	062706	000006			ADD	#6,SP
16326	054570	012605				MOV	(SP)+,R5
16327							
16328							
16329				:+			
16330				:	LOAD GPR'S FROM DATA05 & ADD DATA TYPE		
16331				:-			
16332							
16333	054572			LET R0 :- DATA00 SET.BY #60000			
16334	054572	013700	000562			MOV	DATA00,R0
16335	054576	052700	060000			BIS	#60000,R0
16336	054602			LET R1 :- #DSBUF1			

16337	054602	012701	064274		MOV	#DSBUF1,R1
16338	054606			LET R2 := DATA02 SET.BY #60000		
16339	054606	013702	000566		MOV	DATA02,R2
16340	054612	052702	060000		BIS	#60000,R2
16341	054616			LET R3 := #DSBUF2		
16342	054616	012703	064336		MOV	#DSBUF2,R3
16343	054622			LET R4 := DATA04 SET.BY #60000		
16344	054622	013704	000572		MOV	DATA04,R4
16345	054626	052704	060000		BIS	#60000,R4
16346	054632			LET R5 := #DS.DST		
16347	054632	012705	064400		MOV	#DS.DST,R5
16348						
16349				:+ FINISH SET UP & DO CIS INSTRUCTION		
16350				:+ FINISH SET UP & DO CIS INSTRUCTION		
16351				:+ FINISH SET UP & DO CIS INSTRUCTION		
16352				:+ FINISH SET UP & DO CIS INSTRUCTION		
16353				:+ FINISH SET UP & DO CIS INSTRUCTION		
16354	054636			LET R4.EXP := R4		
16355	054636	010437	000534		MOV	R4,R4.EXP
16356	054642			LET R5.EXP := R5		
16357	054642	010537	000536		MOV	R5,R5.EXP
16358	054646			INLINE <DIVP>		
16359	054646	076075			DIVP	
16360						
16361				:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
16362				:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
16363				:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
16364				:+ CHECK CONDITION CODES AND GPR'S FOR CORRECT RESULTS.		
16365	054650			LET CC := PSW		
16366	054650	013737	177776 000520		MOV	PSW,CC
16367	054656			LET CC := CC CLR.BY #177760		
16368	054656	042737	177760 000520		BIC	#177760,CC
16369	054664			IF CC NE CC.EXP THEN		
16370	054664	023737	000520 000522		CMP	CC,CC.EXP
16371	054672	001402			BEQ	\$52326
16372	054674			CALL ERROR		
16373	054674	004737	060044		JSR	PC,ERROR
16374	054700			ENDIF		
16375	054700				\$52326:	
16376	054700			IF R0 NE R0.EXP THEN		
16377	054700	020037	000524		CMP	R0,R0.EXP
16378	054704	001402			BEQ	\$52327
16379	054706			CALL ERROR		
16380	054706	004737	060044		JSR	PC,ERROR
16381	054712			ENDIF		
16382	054712				\$52327:	
16383	054712			IF R1 NE R1.EXP THEN		
16384	054712	020137	000526		CMP	R1,R1.EXP
16385	054716	001402			BEQ	\$52330
16386	054720			CALL ERROR		
16387	054720	004737	060044		JSR	PC,ERROR
16388	054724			ENDIF		
16389	054724				\$52330:	
16390	054724			IF R2 NE R2.EXP THEN		
16391	054724	020237	000530		CMP	R2,R2.EXP
16392	054730	001402			BEQ	\$52331

16393	054732			CALL ERROR					
16394	054732	004737	060044					JSR	PC,ERROR
16395	054736			ENDIF					
16396	054736								\$52331:
16397	054736			IF R3 NE R3.EXP THEN					
16398	054736	020337	000532					CMP	R3,R3.EXP
16399	054742	001402						BEQ	\$52332
16400	054744			CALL ERROR					
16401	054744	004737	060044					JSR	PC,ERROR
16402	054750			ENDIF					
16403	054750								\$52332:
16404	054750			IF R4 NE R4.EXP THEN					
16405	054750	020437	000534					CMP	R4,R4.EXP
16406	054754	001402						BEQ	\$52333
16407	054756			CALL ERROR					
16408	054756	004737	060044					JSR	PC,ERROR
16409	054762			ENDIF					
16410	054762								\$52333:
16411	054762			IF R5 NE R5.EXP THEN					
16412	054762	020537	000536					CMP	R5,R5.EXP
16413	054766	001402						BEQ	\$52334
16414	054770			CALL ERROR					
16415	054770	004737	060044					JSR	PC,ERROR
16416	054774			ENDIF					
16417	054774								\$52334:
16418									
16419									
16420				:+					
16421				:-	CHECK RESULTS VS EXPECTED RESULTS				
16422									
16423									
16424	054774			LET R0 := #DS.DST					
16425	054774	012700	064400					MOV	#DS.DST,R0
16426	055000			LET R1 := #DS.EXP					
16427	055000	012701	064442					MOV	#DS.EXP,R1
16428	055004			LET ERRFLG := #0					
16429	055004	005037	000456					CLR	ERRFLG
16430	055010			WHILE R0 NE #DS.DST+40 AND ERRFLG EQ #0 DO					
16431	055010								\$52335:
16432	055010	020027	064440					CMP	R0,#DS.DST+40
16433	055014	001410						BEQ	\$52336
16434	055016	005737	000456					TST	ERRFLG
16435	055022	001005						BNE	\$52336
16436	055024			IFB (R0)+ NE (R1)+ THEN					
16437	055024	122021							
16438	055026	001402						CMPB	(R0)+,(R1)+
16439	055030			CALL ERROR				BEQ	\$52337
16440	055030	004737	060044					JSR	PC,ERROR
16441	055034			ENDIF					
16442	055034								\$52337:
16443	055034			ENDDO					
16444	055034	000765						BR	\$52335
16445	055036								\$52336:
16446	055036			LET TSTCAS := TSTCAS + #22					
16447	055036	062737	000022 000544					ADD	#22,TSTCAS
16448	055044			LET COUNT := COUNT - #1					

16449	055044	005337	000470		DEC	COUNT
16450				:UNTIL COUNT EQ #0		
16451	055050			IF COUNT NE #0 THEN		
16452	055050	005737	000470		TST	COUNT
16453	055054	001402			BEQ	\$52340
16454	055056			INLINE <JMP \$5075>		
16455	055056	000137	054360		JMP	\$5075
16456	055062			ENDIF		
16457	055062					\$52340:
16458						
16459				:+		
16460				INLINE CASE - DIVPI, USING UNSIGNED PACKED DATA		
16461				:-		
16462						
16463	055062			CALL CLRDSB		
16464	055062	004737	057040		JSR	PC,CLRDSB
16465						
16466				:+		
16467				LOAD DATAXX AND CC.EXP FROM TEST CASE		
16468				:-		
16469						
16470	055066			LET R0 := #TAB89+2		
16471	055066	012700	055464		MOV	#TAB89+2,R0
16472	055072			LET DATA00 := (R0)+		
16473	055072	012037	000562			
16474	055076			LET DATA01 := (R0)+		
16475	055076	012037	000564			
16476	055102			LET DATA02 := (R0)+		
16477	055102	012037	000566			
16478	055106			LET DATA03 := (R0)+		
16479	055106	012037	000570			
16480	055112			LET DATA04 := (R0)+		
16481	055112	012037	000572			
16482	055116			LET DATA05 := (R0)+		
16483	055116	012037	000574			
16484	055122			LET CC.EXP := (R0)+		
16485	055122	012037	000522			
16486	055126			LET DATA05 := (R0)+		
16487	055126	012037	000574			
16488						
16489				:+		
16490				PACK BUFFERS FOR SRC1, SRC2, & DEST		
16491				:-		
16492						
16493	055132			LET R5 := #CHRSET		
16494	055132	012705	063672			
16495	055136			CALL PACK IN <#DSBUF1, DATA01, DATA00>		
16496	055136	010546				
16497	055140	010605				
16498	055142	162706	000006			
16499	055146	013745	000562			
16500	055152	013745	000564			
16501	055156	012745	064274			
16502	055162	004737	057420			
16503	055166	062706	000006			
16504	055172	012605				

16505	055174			CALL PACK IN <#DSBUF2,DATA03,DATA02>	;SRC2		
16506	055174	010546				MOV	R5,-(SP)
16507	055176	010605				MOV	SP,R5
16508	055200	162706	000006			SUB	#6,SP
16509	055204	013745	000566			MOV	DATA02,-(R5)
16510	055210	013745	000570			MOV	DATA03,-(R5)
16511	055214	012745	064336			MOV	#DSBUF2,-(R5)
16512	055220	004737	057420			JSR	PC,PACK
16513	055224	062706	000006			ADD	#6,SP
16514	055230	012605				MOV	(SP)+,R5
16515	055232			CALL PACK IN <#DS.EXP,DATA05,DATA04>	;DEST		
16516	055232	010546				MOV	R5,-(SP)
16517	055234	010605				MOV	SP,R5
16518	055236	162706	000006			SUB	#6,SP
16519	055242	013745	000572			MOV	DATA04,-(R5)
16520	055246	013745	000574			MOV	DATA05,-(R5)
16521	055252	012745	064442			MOV	#DS.EXP,-(R5)
16522	055256	004737	057420			JSR	PC,PACK
16523	055262	062706	000006			ADD	#6,SP
16524	055266	012605				MOV	(SP)+,R5
16525	055270			CALL CLRREG	;ENSURE GPR'S 0-5 ARE ZERO		
16526	055270	004737	056604			JSR	PC,CLRREG
16527							
16528	055274			INLINE <DIVPI>			
16529	055274	076175					DIVPI
16530	055276			INLINE <.WORD TMULPI>			
16531	055276	056264					.WORD TMULPI
16532	055300			INLINE <.WORD TMULPI+4>			
16533	055300	056270					.WORD TMULPI+4
16534	055302			INLINE <.WORD TMULPI+10>			
16535	055302	056274					.WORD TMULPI+10
16536	055304			LET CC := PSW			
16537	055304	013737	177776	000520		MOV	PSW,CC
16538	055312			LET CC := CC CLR.BY #177760			
16539	055312	042737	177760	000520		BIC	#177760,CC
16540	055320			IF CC NE #0 THEN			
16541	055320	005737	000520			TST	CC
16542	055324	001402				BEQ	\$52341
16543	055326			CALL ERROR			
16544	055326	004737	060044			JSR	PC,ERROR
16545	055332			ENDIF			
16546	055332						\$52341:
16547							
16548				:+			
16549				:-	ENSURE GPR'S STILL ZERO		
16550							
16551							
16552	055332			IF R0 NE #0 OR R1 NE #0 OR R2 NE #0 THEN			
16553	055332	005700				TST	R0
16554	055334	001004				BNE	\$52342
16555	055336	005701				TST	R1
16556	055340	001002				BNE	\$52342
16557	055342	005702				TST	R2
16558	055344	001402				BEQ	\$52343
16559	055346						\$52342:
16560	055346			CALL ERROR			

16561	055346	004737	060044		JSR	PC,ERROR
16562	055352			ENDIF		
16563	055352				\$52343:	
16564	055352			IF R3 NE #0 OR R4 NE #0 OR R5 NE #0 THEN		
16565	055352	005703			TST	R3
16566	055354	001004			BNE	\$52344
16567	055356	005704			TST	R4
16568	055360	001002			BNE	\$52344
16569	055362	005705			TST	R5
16570	055364	001402			BEQ	\$52345
16571	055366				\$52344:	
16572	055366			CALL ERROR		
16573	055366	004737	060044	ENDIF	JSR	PC,ERROR
16574	055372				\$52345:	
16575	055372					
16576						
16577				:+		
16578				:+ CHECK RESULTS		
16579				:-		
16580						
16581	055372			LET R0 := #DS.DST		
16582	055372	012700	064400		MOV	#DS.DST,R0
16583	055376			LET R1 := #DS.EXP ;EXPECTED DATA		
16584	055376	012701	064442		MOV	#DS.EXP,R1
16585	055402			LET R2 := TMULPI+10 CLR.BY #70000 ;GET STRING LENGTH		
16586	055402	013702	056274		MOV	TMULPI+10,R2
16587	055406	042702	070000		BIC	#70000,R2
16588	055412			INLINE <ASR R2> ;LENGTH/2 (PACKING FACTOR)		
16589	055412	006202			ASR	R2
16590	055414			LET R2 := R2 + #DS.EXP ;R2 POINTS TO MSD/SIGN		
16591	055414	062702	064442		ADD	#DS.EXP,R2
16592	055420			LET (R2) :B= (R2) SET.BY #17 ;MAKE IT UNSIGNED		
16593	055420	152712	000017		BISB	#17,(R2)
16594	055424			LET ERRFLG := #0		
16595	055424	005037	000456		CLR	ERRFLG
16596	055430			WHILE R0 NE #DS.DST+37 AND ERRFLG EQ #0 DO		
16597	055430				\$52346:	
16598	055430	020027	064437		CMP	R0,#DS.DST+37
16599	055434	001410			BEQ	\$52347
16600	055436	005737	000456		TST	ERRFLG
16601	055442	001005			BNE	\$52347
16602	055444			IFB (R0)+ NE (R1)+ THEN		
16603	055444	122021			CMPB	(R0)+,(R1)+
16604	055446	001402			BEQ	\$52350
16605	055450			CALL ERROR		
16606	055450	004737	060044	ENDIF	JSR	PC,ERROR
16607	055454				\$52350:	
16608	055454					
16609	055454			ENDDO		
16610	055454	000765			BR	\$52346
16611	055456				\$52347:	
16612	055456			INLINE <JMP CIS9END>		
16613	055456	000137	056300		JMP	CIS9END
16614						

			.SBTTL TEST CASES FOR CIS8 & CIS9			
			TAB89:			
16615						
16616						
16617	055462	000007	.WORD	7		:# OF TEST CASES
16618						
16619	055464	000017	.WORD	15.		:SRC1 LENGTH
16620	055466	055662	.WORD	MUL1		:SRC1 PTR
16621	055470	000021	.WORD	17.		:SRC2 LENGTH
16622	055472	055704	.WORD	MUL1A		:SRC2 PTR
16623	055474	000037	.WORD	31.		:DEST LENGTH
16624	055476	055730	.WORD	MUL1B		:DEST PTR (PRODUCT)
16625	055500	000000	.WORD	0		:CC.EXP (PRODUCT)
16626	055502	055772	.WORD	MUL1C		:DEST PTR (QUOTIENT)
16627	055504	000000	.WORD	0		:CC.EXP (QUOTIENT)
16628						
16629	055506	000004	.WORD	4		
16630	055510	056034	.WORD	MUL2		
16631	055512	000007	.WORD	7		
16632	055514	056042	.WORD	MUL2A		
16633	055516	000013	.WORD	11.		
16634	055520	056054	.WORD	MUL2B		
16635	055522	000010	.WORD	10		
16636	055524	056072	.WORD	MUL2C		
16637	055526	000010	.WORD	10		
16638						
16639	055530	000005	.WORD	5		
16640	055532	056110	.WORD	MUL3		
16641	055534	000002	.WORD	2		
16642	055536	056120	.WORD	MUL3A		
16643	055540	000006	.WORD	6		
16644	055542	056124	.WORD	MUL3B		
16645	055544	000000	.WORD	0		
16646	055546	056134	.WORD	MUL3C		
16647	055550	000004	.WORD	4		
16648						
16649	055552	000005	.WORD	5		
16650	055554	056144	.WORD	MUL4		
16651	055556	000001	.WORD	1		
16652	055560	056154	.WORD	MUL4A		
16653	055562	000001	.WORD	1		
16654	055564	056162	.WORD	MUL4B		
16655	055566	000004	.WORD	4		
16656	055570	056170	.WORD	MUL4C		
16657	055572	000004	.WORD	4		
16658						
16659	055574	000002	.WORD	2		
16660	055576	056176	.WORD	MUL5		
16661	055600	000005	.WORD	5		
16662	055602	056202	.WORD	MUL5A		
16663	055604	000000	.WORD	0		
16664	055606	056212	.WORD	MUL5B		
16665	055610	000006	.WORD	6		
16666	055612	056216	.WORD	MUL5C		
16667	055614	000006	.WORD	6		
16668						
16669	055616	000005	.WORD	5		
16670	055620	056144	.WORD	MUL4		

16671	055622	000001				.WORD	1
16672	055624	056154				.WORD	MUL4A
16673	055626	000004				.WORD	4
16674	055630	056162				.WORD	MUL4B
16675	055632	000004				.WORD	4
16676	055634	056170				.WORD	MUL4C
16677	055636	000004				.WORD	4
16678							
16679	055640	000004				.WORD	4
16680	055642	056222				.WORD	MUL7
16681	055644	000005				.WORD	5
16682	055646	056230				.WORD	MUL7A
16683	055650	000010				.WORD	8.
16684	055652	056240				.WORD	MUL7B
16685	055654	000000				.WORD	0
16686	055656	056252				.WORD	MUL7C
16687	055660	000000				.WORD	0
16688							
16689	055662	000000			MUL1:	.WORD	0
16690	055664	004	004	004		.BYTE	4,4,4,4,4,4,4,4,4,4,4,4,4,4
16691	055667	004	004	004			
16692	055672	004	004	004			
16693	055675	004	004	004			
16694	055700	004	004	004			
16695		055704				.EVEN	
16696							
16697	055704	000000			MUL1A:	.WORD	0
16698	055706	001	007	004		.BYTE	1,7,4,3,2,1,6,9.,5,3,4,7,9.,4,1,6,2
16699	055711	003	002	001			
16700	055714	006	011	005			
16701	055717	003	004	007			
16702	055722	011	004	001			
16703	055725	006	002				
16704		055730				.EVEN	
16705							
16706	055730	000000			MUL1B:	.WORD	0
16707	055732	007	007	004		.BYTE	7,7,4,7,6,3,0,9.,0,4,3,5,2,9.,5,3
16708	055735	007	006	003			
16709	055740	000	011	000			
16710	055743	004	003	005			
16711	055746	002	011	005			
16712	055751	003					
16713	055752	001	004	001		.BYTE	1,4,1,2,5,7,9.,8.,4,5,3,5,9.,2,8.
16714	055755	002	005	007			
16715	055760	011	010	004			
16716	055763	005	003	005			
16717	055766	011	002	010			
16718		055772				.EVEN	
16719							
16720	055772	000000			MUL1C:	.WORD	0
16721	055774	000	000	000		.BYTE	0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
16722	055777	000	000	000			
16723	056002	000	000	000			
16724	056005	000	000	000			
16725	056010	000	000	000			
16726	056013	000					

16727	056014	000	000	000		.BYTE	0,0,0,0,0,0,0,0,0,0,0,0,0,3,9.
16728	056017	000	000	000			
16729	056022	000	000	000			
16730	056025	000	000	000			
16731	056030	000	003	011			
16732	056034					.EVEN	
16733							
16734	056034	177777			MUL2:	.WORD	-1
16735	056036	002	007	004		.BYTE	2,7,4,1
16736	056041	001					
16737						.EVEN	
16738							
16739	056042	000000			MUL2A:	.WORD	0
16740	056044	010	004	006		.BYTE	8.,4,6,2,9.,7,3
16741	056047	002	011	007			
16742	056052	003					
16743	056054	177777			MUL2B:	.EVEN	
16744	056054	177777				.WORD	-1
16745	056056	002	003	001		.BYTE	2,3,1,9.,7,0,0,8.,9.,9.,3
16746	056061	011	007	000			
16747	056064	000	010	011			
16748	056067	011	003				
16749	056072					.EVEN	
16750							
16751	056072	177777			MUL2C:	.WORD	-1
16752	056074	000	000	000		.BYTE	0,0,0,0,0,0,0,3,0,8.,7
16753	056077	000	000	000			
16754	056102	000	003	000			
16755	056105	010	007				
16756	056110					.EVEN	
16757							
16758	056110	177777			MUL3:	.WORD	-1
16759	056112	004	001	007		.BYTE	4,1,7,3,9.
16760	056115	003	011				
16761	056120					.EVEN	
16762							
16763	056120	177777			MUL3A:	.WORD	-1
16764	056122	001	001			.BYTE	1,1
16765						.EVEN	
16766							
16767	056124	000000			MUL3B:	.WORD	0
16768	056126	004	005	011		.BYTE	4,5,9.,1,2,9.
16769	056131	001	002	011			
16770						.EVEN	
16771							
16772	056134	000000			MUL3C:	.WORD	0
16773	056136	000	000	000		.BYTE	0,0,0,0,0,0
16774	056141	000	000	000			
16775						.EVEN	
16776							
16777	056144	000000			MUL4:	.WORD	0
16778	056146	001	007	003		.BYTE	1,7,3,5,4
16779	056151	005	004				
16780	056154					.EVEN	
16781							
16782	056154	000000			MUL4A:	.WORD	0

```

16783 056156 000 000 000 .BYTE 0,0,0,0
16784 056161 000
16785 .EVEN
16786
16787 056162 000000 MUL4B: .WORD 0
16788 056164 000 000 000 .BYTE 0,0,0,0
16789 056167 000
16790 .EVEN
16791
16792 056170 000000 MUL4C: .WORD 0
16793 056172 000 000 000 .BYTE 0,0,0,0
16794 056175 000
16795 .EVEN
16796
16797 056176 000000 MUL5: .WORD 0
16798 056200 001 007 .BYTE 1,7
16799 .EVEN
16800
16801 056202 000000 MUL5A: .WORD 0
16802 056204 005 002 000 .BYTE 5,2,0,0,0
16803 056207 000 000
16804 056212 .EVEN
16805
16806 056212 000000 MUL5B: .WORD 0
16807 056214 000 .BYTE 0
16808 056216 .EVEN
16809
16810 056216 000000 MUL5C: .WORD 0
16811 056220 000 .BYTE 0
16812 056222 .EVEN
16813
16814 ;MUL6 USES THE MUL4 STRINGS
16815
16816 056222 000000 MUL7: .WORD 0
16817 056224 001 00C 000 .BYTE 1,0,0,1
16818 056227 001
16819 .EVEN
16820 056230 000000 MUL7A: .WORD 0
16821 056232 001 000 000 .BYTE 1,0,0,0,0
16822 056235 000 000
16823 056240 .EVEN
16824 056240 000000 MUL7B: .WORD 0
16825 056242 001 000 000 .BYTE 1,0,0,1,0,0,0,0
16826 056245 001 000 000
16827 056250 000 000
16828 .EVEN
16829 056252 000000 MUL7C: .WORD 0
16830 056254 000 000 000 .BYTE 0,0,0,0,0,0,0,9.
16831 056257 000 000 000
16832 056262 000 011
16833 .EVEN
16834
16835
16836 ;+
16837 ;-
16838 ;-
  
```

```

16839
16840 056264 070017
16841 056266 064274
16842 056270 070021
16843 056272 064336
16844 056274 070037
16845 056276 064400
16846
16847
16848 056300
16849
16850
16851
16852
16853
16854
16855
16856
16857
16858 056300
16859 056300 005737 000446
16860 056304 001012
16861 056306 032777 000200 122112
16862 056314 001006
16863 056316
16864 056316 005037 000402
16865 056322
16866 056322 005037 000404
16867 056326
16868 056326 000137 001524
16869 056332
16870 056332
16871
16872
16873
16874
16875
16876
16877
16878
16879
16880
16881 056332
16882 056332 000004
16883 056334 005037 000402
16884 056340 005037 000404
16885 056344 005237 000640
16886 056350 042737 100000 000640
16887 056356 005337 056504
16888 056362 003027
16889 056364 013737 056506 056504
16890 056372 012737 056513 000550
16891 056400 104401
16892 056402 013746 000640
16893 056406 104403
16894 056410 006

```

```

TMULPI: .WORD 070017
         .WORD #DSBUF1
         .WORD 070021
         .WORD #DSBUF2
         .WORD 070037
         .WORD #DS.DST

```

CIS9END:

```

: +
: BEFORE GOING TO $EOP DETERMINE IF TESTING WITH AND WITHOUT
: MEMORY MANAGEMENT ENABLED HAS OCCURRED IF APPLICABLE.
: IF MEMORY MANAGEMENT DISABLED THEN CLEAR TSTNM AND ITERATION
: COUNT AND GO TEST WITH MEMORY MANAGEMENT ENABLED.
: -

```

IF @#MMCYL EQ #0 AND #BIT07 NOTSETIN @SWR THEN

```

TST @#MMCYL
BNE $52351
BIT #BIT07,@SWR
BNE $52351
CLR $TSTNM
CLR $TIMES
JMP BEGIN
$52351:

```

LET \$TSTNM := #0

LET \$TIMES := #0

INLINE <JMP BEGIN>

ENDIF

```

.ABS
:*****
:SBTTL END OF PASS ROUTINE
:*****
:*INCREMENT THE PASS NUMBER ($PASS)
:*TYPE 'END PASS #XXXXX' (WHERE XXXXX IS A DECIMAL NUMBER)
:*IF THERES A MONITOR GO TO IT
:*IF THERE ISN'T JUMP TO BEGIN

```

\$EOP:

```

SCOPE
CLR $TSTNM ;;ZERO THE TEST NUMBER
CLR $TIMES
INC $PASS ;;INCREMENT THE PASS NUMBER
BIC #100000,$PASS ;;DON'T ALLOW A NEG. NUMBER
DEC $EOPCT ;;LOOP?
BGT $DOAGN ;;YES
MOV $ENDCT,$EOPCT ;;RESTORE COUNTER
MOV # $ENDMG,MSGADR
TYPE 'END PASS #'
MOV $PASS,-(SP) ;;SAVE $PASS FOR TYPEOUT
TYPOS ;;GO TYPE--OCTAL ASCII, NO LEADING ZEROES
.BYTE 6

```

```

16895 056411 000 .BYTE 0
16896 056412 013737 056510 000550 MOV $ENULL,MSGADR
16897 056420 104401 TYPE ;;TYPE A NULL CHARACTER
16898 056422 013700 000042 $GET42: MOV @#42,R0 ;;GET MONITOR ADDRESS
16899 056426 001405 BEQ $DOAGN ;;BRANCH IF NO MONITOR
16900 056430 000005 RESET ;;CLEAR THE WORLD
16901 056432 004710 $ENDAD: JSR PC,(R0) ;;GO TO MONITOR
16902 056434 000240 NOP ;;SAVE ROOM
16903 056436 000240 NOP ;;FOR
16904 056440 000240 NOP ;;ACT11
16905 056442
16906 056442 013737 000004 000476 $DOAGN: MOV @#4,@#TEMPO ;;SAVE CONTENTS OF LOCATION 4
16907 056450 012737 056466 000004 MOV #1$,@#4 ;;SET UP VECTOR IN CASE OF TRAP
16908 056456 012737 000001 164000 MOV #1,@#164000 ;;NOTIFY MULTI-TESTER OF PASS COMPLETE
16909 056464 000402 BR 2$ ;;NO TRAP --> DO NOT NEED TO RESTORE STACK
16910 056466 062706 000004 1$: ADD #4,SP ;;RESET STACK BECAUSE OF TRAP
16911 056472 013737 000476 000004 2$: MOV @#TEMPO,@#4 ;;RESTORE CONTENTS OF LOCATION 4
16912 056500 000137 00'524 JMP BEGIN ;;RETURN
16913 056504 000001 $EOPCT: .WORD 1
16914 056506 000001 $ENDCT: .WORD 1
16915 056510 377 377 000 $ENULL: .BYTE -1,-1,0 ;;NULL CHARACTER STRING
16916 056513 015 042412 042116 $ENDMG: .ASCIZ <15><12>/END PASS #/
16917 056520 050040 051501 020123
16918 056526 000043
16919
16920
16921
16922
16923
16924 056530 173
16925 056531 101
16926 056532 102
16927 056533 103
16928 056534 104
16929 056535 105
16930 056536 106
16931 056537 107
16932 056540 110
16933 056541 111
16934 056542 175
16935 056543 112
16936 056544 113
16937 056545 114
16938 056546 115
16939 056547 116
16940 056550 117
16941 056551 120
16942 056552 121
16943 056553 122
16944
16945
16946
16947
16948
16949
16950
    
```

 : TABLE OF SIGN/DIGIT COMBINATIONS FOR OVERPUNCH

```

OVPNCH: .BYTE 173 : +0
        .BYTE 101 : +1
        .BYTE 102 : +2
        .BYTE 103 : +3
        .BYTE 104 : +4
        .BYTE 105 : +5
        .BYTE 106 : +6
        .BYTE 107 : +7
        .BYTE 110 : +8
        .BYTE 111 : +9
        .BYTE 175 : -0
        .BYTE 112 : -1
        .BYTE 113 : -2
        .BYTE 114 : -3
        .BYTE 115 : -4
        .BYTE 116 : -5
        .BYTE 117 : -6
        .BYTE 120 : -7
        .BYTE 121 : -8
        .BYTE 122 : -9
    
```

 : SBTTL UNEXPECTED TIME-OUT TRAP ROUTINE

 : *ANY UNEXPECTED TIME-OUT TRAP WILL END UP HERE
 : *

```

16951
16952 056554 012737 062272 000550 TIMEOUT: MOV #TIMMSG,MSGADR ;TYPE OUT ERROR MESSAGE
16953 056562 104401 TYPE
16954 056564 000000 HALT
16955 056566 000002 RTI
16956
16957
16958 ::*****
16959 .SBTTL ILLEGAL/RESERVED INSTRUCTION TRAP ROUTINE
16960 ::*****
16961 :*ANY ILLEGAL OR RESERVED INSTRUCTION TRAPS WILL END UP HERE.*
16962 :*
16963
16964 056570 012737 062340 000550 ILLRES: MOV #ILLMSG,MSGADR ;TYPE OUT ERROR MESSAGE
16965 056576 104401 TYPE
16966 056600 000000 HALT
16967 056602 000002 RTI
16968
16969
16970 ::*****
16971 .SBTTL CLEAR REGISTERS R0 - R5 ROUTINE
16972 ::*****
16973 :*THIS ROUTINE CLEARS GENERAL PURPOSE REGISTERS R0 THROUGH R5.*
16974 :*
16975
16976 056604 ROUTINE CLRREG
16977 056604 CLRREG:
16978 056604 LET R0 := #0 CLR R0
16979 056604 005000 CLR R1
16980 056606 LET R1 := #0 CLR R1
16981 056606 005001 CLR R2
16982 056610 LET R2 := #0 CLR R2
16983 056610 005002 CLR R3
16984 056612 LET R3 := #0 CLR R3
16985 056612 005003 CLR R4
16986 056614 LET R4 := #0 CLR R4
16987 056614 005004 CLR R5
16988 056616 LET R5 := #0 CLR R5
16989 056616 005005
16990 056620 ENDRTN
16991 056620 $52352:
16992 056620 $52353:
16993 056620 000207 RTS PC
16994
16995
16996
16997
16998 ::*****
16999 .SBTTL CLEAR BUFFER ROUTINE
17000 ::*****
17001 :*THIS ROUTINE CLEARS A 100 BYTE BUFFER WHOSE STARTING ADDRESS
17002 :*HAS BEEN PUT INTO 'BFADRS'.*
17003 :*
17004
17005 056622 ROUTINE CLRBUF
17006 056622 CLRBUF:
    
```


17007 056622
17008 056622 010037 000600
17009 056626
17010 056626 010137 000602
17011 056632
17012 056632 013700 000542
17013 056636
17014 056636 012701 000040
17015 056642
17016 056642
17017 056642
17018 056642 005020
17019 056644
17020 056644 005301
17021 056646
17022 056646 005701
17023 056650 001374
17024 056652
17025 056652 013700 000600
17026 056656
17027 056656 013701 000602
17028 056662
17029 056662
17030 056662
17031 056662 000207
17032
17033
17034
17035
17036
17037
17038
17039
17040
17041
17042 056664
17043 056664
17044 056664
17045 056664 010037 000600
17046 056670
17047 056670 010137 000602
17048 056674
17049 056674 013700 000542
17050 056700
17051 056700 012701 000040
17052 056704
17053 056704
17054 056704
17055 056704 012720 177777
17056 056710
17057 056710 005301
17058 056712
17059 056712 005701
17060 056714 001373
17061 056716
17062 056716 013700 000600

LET SAVRO := R0
LET SAVR1 := R1
LET R0 := BFADRS
LET R1 := #40
REPEAT
 LET (R0)+ := #0
 LET R1 := R1 - #1
UNTIL R1 EQ #0
LET R0 := SAVRO
LET R1 := SAVR1
ENDRTN

MOV R0,SAVRO
MOV R1,SAVR1
MOV BFADRS,R0
MOV #40,R1
\$52356:
CLR (R0)+
DEC R1
TST R1
BNE \$52356
MOV SAVRO,R0
MOV SAVR1,R1
\$52354:
\$52355:
RTS PC

:SBTTL SET BUFFER ROUTINE
:*****
:*THIS ROUTINE SETS ALL BITS IN A 100 BYTE BUFFER WHOSE STARTING
:*ADDRESS HAS BEEN PUT INTO 'BFADRS'.
:*

ROUTINE SETBUF
LET SAVRO :- R0
LET SAVR1 := R1
LET R0 :- BFADRS
LET R1 : #40
REPEAT
 LET (R0)+' :- #177777
 LET R1 :- R1 - #1
UNTIL R1 EQ #0
LET R0 := SAVRO

SETBUF:
MOV R0,SAVRO
MOV R1,SAVR1
MOV BFADRS,R0
MOV #40,R1
\$52361:
MOV #177777,(R0)+
DEC R1
TST R1
BNE \$52361
MOV SAVRO,R0

```

17063 056722          LET R1 := SAVR1
17064 056722 013701 000602          MOV     SAVR1,R1
17065 056726          ENDRTN
17066 056726          $52357:
17067 056726          $52360:
17068 056726 000207          RTS     PC
17069
17070
17071
17072          ::*****
17073          .SBTTL  ONES AND ZEROES BUFFER ROUTINE
17074          ::*****
17075          ;*THIS ROUTINE SETS AN ALTERNATING ONES AND ZEROES PATTERN IN A 100 BYTE
17076          ;*BUFFER WHOSE STARTING ADDRESS HAS BEEN PUT INTO 'BFADRS'.
17077          ;*
17078
17079 056730          ROUTINE  ALTBUF
17080 056730          ALTBUF :
17081 056730          LET SAVR0 := R0
17082 056730 010037 000600          MOV     R0,SAVR0
17083 056734          LET SAVR1 := R1
17084 056734 010137 000602          MOV     R1,SAVR1
17085 056740          LET R0 := BFADRS
17086 056740 013700 000542          MOV     BFADRS,R0
17087 056744          LET R1 := #40
17088 056744 012701 000040          MOV     #40,R1
17089 056750          REPEAT
17090 056750          $52364:
17091 056750          LET (R0)+ := #125252
17092 056750 012720 125252          MOV     #125252,(R0)+
17093 056754          LET R1 := R1 - #1
17094 056754 005301          DEC     R1
17095 056756          UNTIL R1 EQ #0
17096 056756 005701          TST     R1
17097 056760 001373          BNE     $52364
17098 056762          LET R0 := SAVR0
17099 056762 013700 000600          MOV     SAVR0,R0
17100 056766          LET R1 := SAVR1
17101 056766 013701 000602          MOV     SAVR1,R1
17102 056772          ENDRTN
17103 056772          $52362:
17104 056772          $52363:
17105 056772 000207          RTS     PC
17106
17107          ::*****
17108          .SBTTL  ONES AND ZEROES BUFFER ROUTINE FOR 2000 BYTE BUFFER
17109          ::*****
17110          ;*THIS ROUTINE SETS AN ALTERNATING ONES AND ZEROES PATTERN IN A 2000 BYTE
17111          ;*BUFFER WHOSE STARTING ADDRESS HAS BEEN PUT INTO 'BFADRS'.
17112          ;*
17113
17114 056774          ROUTINE  ALTBU1
17115 056774          ALTBU1 :
17116 056774          LET SAVR0 := R0
17117 056774 010037 000600          MOV     R0,SAVR0
17118 057000          LET SAVR1 := R1
    
```

17119 057000 010137 000602
 17120 057004
 17121 057004 013700 000542
 17122 057010
 17123 057010 012701 001000
 17124 057014
 17125 057014
 17126 057014
 17127 057014 012720 125252
 17128 057020
 17129 057020 005301
 17130 057022
 17131 057022 005701
 17132 057024 001373
 17133 057026
 17134 057026 013700 000600
 17135 057032
 17136 057032 013701 000602
 17137 057036
 17138 057036
 17139 057036
 17140 057036 000207
 17141
 17142
 17143

LET R0 := BFADRS
 LET R1 := #1000
 REPEAT
 LET (R0)+ := #125252
 LET R1 := R1 - #1
 UNTIL R1 EQ #0
 LET R0 := SAVR0
 LET R1 := SAVR1
 ENDRTN

MOV R1,SAVR1
 MOV BFADRS,R0
 MOV #1000,R1
 \$52367:
 MOV #125252,(R0)+
 DEC R1
 TST R1
 BNE \$52367
 MOV SAVR0,R0
 MOV SAVR1,R1
 \$52365:
 \$52366:
 RTS PC

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 337
ONES AND ZEROES BUFFER ROUTINE FOR 2000 BYTE BUFFER

SEQ 0337

17144
17145
17146
17147

```

:*****
:SBTTL CLEAR DECIMAL STRING BUFFERS ROUTINE
:*****

```

17148
17149
17150
17151
17152
17153
17154
17155
17156
17157
17158
17159
17160
17161

057040
057040
057040
057040 010037 000600
057044
057044 010137 000602
057050
057050 012700 064273
057054
057054 012701 000207

;*THIS ROUTINE CLEARS ALL BITS IN THE FOUR 40-BYTE DECIMAL
;*STRING BUFFERS.
;*

ROUTINE CLRDSB
LET SAVRO := R0
LET SAVR1 := R1
LET R0 := #DSBUF1-1
LET R1 := #207

CLRDSB:
MOV R0,SAVRO
MOV R1,SAVR1
MOV #DSBUF1-1,R0
MOV #207,R1

17162 057060
 17163 057060
 17164 057060
 17165 057060 105020
 17166 057062
 17167 057062 005301
 17168 057064
 17169 057064 005701
 17170 057066 001374
 17171 057070
 17172 057070 013700 000600
 17173 057074
 17174 057074 013701 000602
 17175 057100
 17176 057100
 17177 057100
 17178 057100 000207
 17179
 17180
 17181
 17182
 17183
 17184
 17185
 17186
 17187
 17188 057102
 17189 057102
 17190 057102
 17191 057102 010037 000600
 17192 057106
 17193 057106 010137 000602
 17194 057112
 17195 057112 010237 000604

REPEAT
 LET (R0)+ :B= #0
 LET R1 := R1 - #1
 UNTIL R1 EQ #0
 LET R0 := SAVR0
 LET R1 := SAVR1
 ENDRTN

\$52372:
 CLRB (R0)+
 DEC R1
 TST R1
 BNE \$52372
 MOV SAVR0,R0
 MOV SAVR1,R1
 \$52370:
 \$52371:
 RTS PC

::*****
 :SBTTL SAVE REGISTERS R0 - R5
 :*****
 :*THIS ROUTINE SAVES R0 THROUGH R5 IN LOCATIONS SAVR0 - SAVR5.
 :*

ROUTINE SAVREG
 LET SAVR0 := R0
 LET SAVR1 := R1
 LET SAVR2 := R2

SAVREG:
 MOV R0,SAVR0
 MOV R1,SAVR1
 MOV R2,SAVR2

17196	057116			LET SAVR3 := R3		
17197	057116	010337	000606		MOV	R3,SAVR3
17198	057122			LET SAVR4 := R4		
17199	057122	010437	000610		MOV	R4,SAVR4
17200	057126			LET SAVR5 := R5		
17201	057126	010537	000612		MOV	R5,SAVR5
17202	057132			ENDRTN		
17203	057132				\$52373:	
17204	057132				\$52374:	
17205	057132	000207			RTS	PC
17206						
17207						

```
17208
17209
17210
17211
17212
17213
17214
17215 057134
17216 057134
17217 057134
17218 057134 013700 000600
17219 057140
17220 057140 013701 000602
17221 057144
17222 057144 013702 000604
17223 057150
17224 057150 013703 000606
17225 057154
17226 057154 013704 000610
17227 057160
17228 057160 013705 000612
17229 057164
17230 057164
17231 057164
17232 057164 000207
17233
17234
17235
17236
17237
17238
17239
17240
17241
17242
17243
17244 057166
17245 057166
17246
17247
17248
17249
17250
17251
17252
17253 057166
17254 057166 005137 000446
17255
17256
17257
17258
17259
17260
17261 057172
17262 057172 032777 000100 121226
17263 057200 001404
```

```
*****
:SBTTL RESTORE REGISTERS R0 - R5
:*****
:*THIS ROUTINE RESTORES R0 THROUGH R5 FROM LOCATIONS SAVR0 - SAVR5
:*
ROUTINE RESREG
RESREG:
LET R0 := SAVR0      MOV SAVR0,R0
LET R1 := SAVR1      MOV SAVR1,R1
LET R2 := SAVR2      MOV SAVR2,R2
LET R3 := SAVR3      MOV SAVR3,R3
LET R4 := SAVR4      MOV SAVR4,R4
LET R5 := SAVR5      MOV SAVR5,R5
ENDRTN
$52375:
$52376:
RTS PC

*****
:SBTTL MEMORY MANAGEMENT SET UP ROUTINE
: ROUTINE WILL DETERMINE IF MEMORY MANAGEMENT SHOULD BE ENABLED OR DISABLED.
: IF MEMORY MANAGEMENT IS TO BE ENABLED THE APRS WILL BE MAPPED SUCH THAT THE
: VIRTUAL ADDRESS WILL BE MAPPED TO ITSELF AND MEMORY MANAGEMENT IS ENABLED.
:*****
ROUTINE MMSETUP
MMSETUP:
: +
: COMPLIMENT THE MEMORY MANAGEMENT CYCLE FLAG
: MMCYL = 0 -KTOFF
: MMCYL = -1 -KTON
: -
LET @MMCYL : COMP @MMCYL      COM @MMCYL
: +
: DETERMINE WHETHER TO ONLY CYCLE WITH MEMORY MANAGEMENT ON OR TO CYCLE
: WITH MEMORY MANAGEMENT OFF.
: -
IF #BIT06 SETIN @SWR THEN
BIT #BIT06,@SWR
BEQ $52401
```



```

17264 057202
17265 057202 012737 177777 000446
17266 057210
17267 057210 000406
17268 057212
17269 057212
17270 057212 032777 000200 121206
17271 057220 001402
17272 057222
17273 057222 005037 000446
17274 057226
17275 057226
17276 057226
17277 057226
17278
17279
17280
17281
17282
17283 057226
17284 057226 005737 000640
17285 057232 001057
17286 057234
17287 057234 005037 172340
17288 057240
17289 057240 012737 000200 172342
17290 057246
17291 057246 012737 000400 172344
17292 057254
17293 057254 012737 000600 172346
17294 057262
17295 057262 012737 001000 172350
17296 057270
17297 057270 012737 001200 172352
17298 057276
17299 057276 012737 001400 172354
17300 057304
17301 057304 012737 007600 172356
17302
17303 057312
17304 057312 012737 077406 172300
17305 057320
17306 057320 012737 077406 172302
17307 057326
17308 057326 012737 077406 172304
17309 057334
17310 057334 012737 077406 172306
17311 057342
17312 057342 012737 077406 172310
17313 057350
17314 057350 012737 077406 172312
17315 057356
17316 057356 012737 077406 172314
17317 057364
17318 057364 012737 077406 172316
17319 057372

```

```

LET @AMMICYL := #-1
ELSE
IF #BIT07 SETIN @SWR THEN
LET @AMMICYL := #0
ENDIF
ENDIF
ENDIF
:~
:~ ONLY SET UP MEMORY MANAGEMENT REGISTERS IF 1ST PASS
:~
IF @M$PASS EQ #0 THEN
LET @M$KIPAR0 :- #0
LET @M$KIPAR1 :- #200
LET @M$KIPAR2 :- #400
LET @M$KIPAR3 :- #600
LET @M$KIPAR4 :- #1000
LET @M$KIPAR5 : #1200
LET @M$KIPAR6 :- #1400
LET @M$KIPAR7 :- #7600
LET @M$KIPDR0 :- #77406
LET @M$KIPDR1 := #77406
LET @M$KIPDR2 : #77406
LET @M$KIPDR3 := #77406
LET @M$KIPDR4 := #77406
LET @M$KIPDR5 :- #77406
LET @M$KIPDR6 :- #77406
LET @M$KIPDR7 := #77406
ENDIF

```

```

MOV #-1,@AMMICYL
BR $52402
$52401:
BIT #BIT07,@SWR
BEQ $52403
CLR @AMMICYL
$52403:
$52402:
TST @M$PASS
BNE $52404
CLR @M$KIPAR0
MOV #200,@M$KIPAR1
MOV #400,@M$KIPAR2
MOV #600,@M$KIPAR3
MOV #1000,@M$KIPAR4
MOV #1200,@M$KIPAR5
MOV #1400,@M$KIPAR6
MOV #7600,@M$KIPAR7
MOV #77406,@M$KIPDR0
MOV #77406,@M$KIPDR1
MOV #77406,@M$KIPDR2
MOV #77406,@M$KIPDR3
MOV #77406,@M$KIPDR4
MOV #77406,@M$KIPDR5
MOV #77406,@M$KIPDR6
MOV #77406,@M$KIPDR7

```

```

17320 057372                                $52404:
17321
17322
17323      :+ NOW TURN ON MEMORY MANAGEMENT IF IN MEMORY MANAGEMENT MEMORY MANAGEMENT CYCLE
17324      :-
17325
17326 057372                                IF @#MMCYL NE #0 THEN
17327 057372 005737 000446                    TST    @#MMCYL
17328 057376 001404                            BEQ    $52405
17329 057400                                LET @#SR1 := @#SR1 SET.BY #BIT0
17330 057400 052737 000001 177572          BIS    #BIT0,@#SR1
17331
17332      :+ ELSE TURN OFF MEMORY MANAGEMENT
17333      :-
17334 057406                                ELSE
17335 057406 000403
17336 057410
17337 057410                                LET @#SP1 := @#SR1 CLR.BY #BIT0
17338 057410 042737 000001 177572          BIC    #BIT0,@#SR1
17339 057416                                ENDIF
17340 057416                                ENDRTN
17341 057416
17342 057416
17343 057416
17344 057416 000207

```

```

$52404:
TST    @#MMCYL
BEQ    $52405
BIS    #BIT0,@#SR1

BR     $52406
$52405:
BIC    #BIT0,@#SR1

$52406:
$52377:
$52400:
RTS    PC

```

17345
 17346
 17347
 17348
 17349
 17350
 17351
 17352
 17353
 17354
 17355
 17356
 17357
 17358
 17359
 17360
 17361
 17362
 17363
 17364
 17365
 17366
 17367
 17368
 17369
 17370
 17371
 17372
 17373
 17374
 17375
 17376
 17377
 17378
 17379
 17380
 17381
 17382
 17383
 17384
 17385
 17386
 17387
 17388
 17389
 17390
 17391
 17392
 17393
 17394
 17395
 17396
 17397
 17398
 17399
 17400

057420
 057420
 057420 016500 000000
 057424
 057424 016501 000002
 057430
 057430 016502 000004
 057434 042702 070000

 .SBTTL PACK - PACK DECIMAL STRING ROUTINE

:+
 :
 : FUNCTIONAL DESCRIPTION

THE PACK ROUTINE ACCEPTS UNPACKED DATA FROM A TEST CASE TABLE AND PACKS IT (TWO BCD CHARS/BYTE) INTO ONE OF THE DECIMAL STRING BUFFERS.

INPUTS:

1. ADDRESS OF THE DESTINATION BUFFER. ** NOTE: ** THIS MUST BE #DSBUF1, #DSBUF2, OR #DS.EXP.
2. POINTER TO SOURCE DATA IN TEST CASE TABLE. ** NOTE: ** THIS MUST BE DATA01, DATA03, OR DATA05.
3. POINTER TO DESTINATION STRING LENGTH. ** NOTE: ** THIS MUST BE DATA00, DATA02, OR DATA04.

OUTPUTS:
 NONE

PATHOLOGICAL CONNECTIONS:
 NONE.

SUBORDINATE ROUTINES:
 NONE.

FUNCTIONAL SIDE-EFFECTS:
 NONE.

CALLING SEQUENCE:

CALL PACK IN <DSTADR, SRCADR, LENGTH>. FOR INPUT PARAMETERS REFER TO ABOVE INPUTS:.

NOTE:

R3 IS USED AS A BYTE ASSEMBLY AREA.
 R4 IS USED TO HOLD THE SIGN OF THE STRING (+ - #14, - = #15).

ROUTINE PACK <DSTADR, SRCADR, LENGTH>

LET R0 := DSTADR(R5)
 LET R1 := SRCADR(R5)
 LET R2 := LENGTH(R5) CLR.BY #70000

PACK:

MOV DSTADR(R5), R0
 MOV SRCADR(R5), R1
 MOV LENGTH(R5), R2
 BIC #70000, R2

:+
 : STORE SIGN OF STRING IN R4
 :-

```

17401 057440          IF (R1)+ EQ #0 THEN          ;SIGN IS +
17402 057440 005721          ;
17403 057442 001003          ;
17404 057444          LET R4 := #14
17405 057444 012704 000014          ;
17406 057450          ELSE          ;SIGN IS -
17407 057450 000402          ;
17408 057452          ;
17409 057452          LET R4 := #15
17410 057452 012704 000015          ;
17411 057456          ENDIF
17412 057456          ;
17413
17414
17415          ;+
17416          ; IF LENGTH IS ZERO, THEN SIGN STRING AND EXIT; ELSE IF
17417          ; STRING LENGTH IS EVEN, STORE THE MSD (STRING LENGTH IS
17418          ; NOW ODD).NOW STORE ALL BUT THE LAST DIGIT. BUILD LSD
17419          ; & SIGN AND EXIT.
17420          ;-
17421 057456          IF R2 EQ #0 THEN          ;STRING LENGTH = 0
17422 057456 005702          ;
17423 057460 001002          ;
17424 057462          LET (R0) :B= R4          ;SIGN THE BUFFER AND EXIT
17425 057462 110410          ;
17426 057464          ELSE
17427 057464 000425          ;
17428 057466          ;
17429 057466          IF #BIT00 NOTSET IN R2 THEN ;LENGTH IS EVEN & > 0
17430 057466 032702 000001          ;
17431 057472 001002          ;
17432 057474          LET (R0)+ :B= (R1)+          ;MOVE THE EVEN NIBBLE
17433 057474 112120          ;
17434 057476          LET R2 := R2 - #1          ;DEC NIBBLE COUNTER
17435 057476 005302          ;
17436 057500          ENDIF
17437 057500          ;
17438 057500          WHILE R2 NE #1 DO
17439 057500          ;
17440 057500 020227 000001          ;
17441 057504 001410          ;
17442 057506          LET R3 :B= (R1)+          ;GET A BCD
17443 057506 112103          ;
17444 057510          INLINE <ASH #4, R3>          ;SHIFT IT TO THE HIGH NIBBLE
17445 057510 072327 000004          ;
17446 057514          LET R3 :B= R3 SET.BY (R1)+          ;PACK IN LOW NIBBLE
17447 057514 152103          ;
17448 057516          LET (R0)+ :B= R3          ;STORE RESULT
17449 057516 110320          ;
17450 057520          LET R2 := R2 - #2          ;DEC NIBBLE COUNTER
17451 057520 162702 000002          ;
17452 057524          ENDDO
17453 057524 000765          ;
17454 057526          ;
17455 057526          LET R3 :B= (R1)          ;GET LAST BCD
17456 057526 111103          ;

```

```

TST (R1)+
BNE $52411
MOV #14,R4
BR $52412
$52411:
MOV #15,R4
$52412:
TST R2
BNE $52413
MOVB R4,(R0)
BR $52414
$52413:
BIT #BIT00,R2
BNE $52415
MOVB (R1)+,(R0)+
DEC R2
$52415:
$52416:
CMP R2,#1
BEQ $52417
MOVB (R1)+,R3
ASH #4,R3
BISB (R1)+,R3
MOVB R3,(R0)+
SUB #2,R2
BR $52416
$52417:
MOVB (R1),R3

```

17457 057530
17458 057530 072327 000004
17459 057534
17460 057534 150403
17461 057536
17462 057536 110310
17463 057540
17464 057540
17465 057540
17466 057540
17467 057540
17468 057540 000207
17469

INLINE <ASH #4, R3> ;SHIFT IT TO THE HIGH NIBBLE
LET R3 :B= R3 SET.BY R4 ;ADD SIGN
LET (R0) :B= R3 ;STORE LSD & SIGN
ENDIF
ENDPTN
\$52414:
\$52407:
\$52410:
RTS PC

17470
17471
17472
17473
17474
17475
17476
17477
17478
17479
17480
17481
17482
17483 057542
17484 057542 104406
17485 057544 032777 040000 120654
17486 057552 001034
17487 057554 032777 004000 120644
17488 057562 001011
17489 057564 005737 000640
17490 057570 001406
17491 057572 005237 000406
17492 057576 023737 000404 000406
17493 057604 002017
17494 057606 012737 000001 000406
17495 057614 013737 057660 000404
17496 057622 005237 000402
17497 057626 013737 000402 000636
17498 057634 011637 000410
17499 057640 005037 000456
17500 057644 013777 000402 120556
17501 057652 013716 000410
17502 057656 000002
17503 057660 000100
17504
17505
17506
17507
17508
17509
17510 057662 012737 060032 000024
17511 057670 012737 000340 000026
17512 057676 010046
17513 057700 010146
17514 057702 010246
17515 057704 010346
17516 057706 010446
17517 057710 010546
17518 057712 017746 120510
17519 057716 010637 060036
17520 057722 012737 057734 000024
17521 057730 000000
17522 057732 000776
17523
17524
17525

```
:::*****  
:SBTTL SCOPE HANDLER ROUTINE  
:*****  
:*THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT  
:*AND LOAD THE TEST NUMBER($STNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)  
:*AND LOAD THE ERROR FLAG (ERRFLG) INTO DISPLAY<15:08>  
:*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:  
:*SW14=1 LOOP ON TEST  
:*SW11=1 INHIBIT ITERATIONS  
:*  
:*CALL  
:* SCOPE ;:SCOPE=IOT  
$SCOPE:  
CKSWR ;:TEST FOR CHANGE IN SOFT-SWR  
BIT #BIT14,@SWR ;:LOOP ON PRESENT TEST?  
BNE $OVER ;:YES IF SW14=1  
BIT #BIT11,@SWR ;:INHIBIT ITERATIONS?  
BNE 1$ ;:BRANCH IF YES  
TST $PASS ;:IF FIRST PASS OF PROGRAM  
BEQ 1$ ;: INHIBIT ITERATIONS  
INC $ICNT ;:INCREMENT ITERATION COUNT  
CMP $TIMES,$ICNT ;:CHECK THE NUMBER OF ITERATIONS MADE  
BGE $OVER ;:BRANCH IF MORE ITERATION REQUIRED  
1$: MOV #1,$ICNT ;:REINITIALIZE THE ITERATION COUNTER  
MOV $MXCNT,$TIMES ;:SET NUMBER OF ITERATIONS TO DO  
$SVLAD: INC $STNM ;:COUNT TEST NUMBERS  
MOV $STNM,$TESTN ;:SET TEST NUMBER IN APT MAILBOX  
MOV (SP),$LPADR ;:SAVE SCOPE LOOP ADDRESS  
CLR ERRFLG ;:ZERO THE ERROR FLAG  
$OVER: MOV $STNM,@DISPLAY ;:DISPLAY TEST NUMBER  
MOV $LPADR,(SP) ;:FUDGE RETURN ADDRESS  
RTI ;:FIXES PS  
$MXCNT: .WORD 100 ;:MAXIMUM NUMBER OF ITERATIONS  
:*****  
:SBTTL POWER DOWN AND UP ROUTINES  
:*****  
:POWER DOWN ROUTINE  
$PWRDN: MOV #$ILLUP,@#PWRVEC ;:SET FOR FAST UP  
MOV #340,@#PWRVEC+2 ;:PRIO:7  
MOV R0,-(SP) ;:PUSH R0 ON STACK  
MOV R1,-(SP) ;:PUSH R1 ON STACK  
MOV R2,-(SP) ;:PUSH R2 ON STACK  
MOV R3,-(SP) ;:PUSH R3 ON STACK  
MOV R4,-(SP) ;:PUSH R4 ON STACK  
MOV R5,-(SP) ;:PUSH R5 ON STACK  
MOV @SWR,-(SP) ;:PUSH @SWR ON STACK  
MOV SP,$SAVR6 ;:SAVE SP  
MOV #PWRUP,@#PWRVEC ;:SET UP VECTOR  
HALT  
BR .-2 ;:HANG UP  
:*****  
:POWER UP ROUTINE
```

```
17526 057734 012737 060032 000024 $PWRUP: MOV    #$ILLUP,@#PWRVEC ;;SET FOR FAST DOWN
17527 057742 013706 060036          MOV    $SAVR6,SP      ;;GET SP
17528 057746 005037 060036          CLR    $SAVR6        ;;WAIT LOOP FOR THE TTY
17529 057752 005237 060036          1$:   INC    $SAVR6    ;;WAIT FOR THE INC
17530 057756 001375          BNE    1$           ;;OF WORD
17531 057760 012677 120442          MOV    (SP)+,@SWR    ;;POP STACK INTO @SWR
17532 057764 012605          MOV    (SP)+,R5     ;;POP STACK INTO R5
17533 057766 012604          MOV    (SP)+,R4     ;;POP STACK INTO R4
17534 057770 012603          MOV    (SP)+,R3     ;;POP STACK INTO R3
17535 057772 012602          MOV    (SP)+,R2     ;;POP STACK INTO R2
17536 057774 012601          MOV    (SP)+,R1     ;;POP STACK INTO R1
17537 057776 012600          MOV    (SP)+,R0     ;;POP STACK INTO R0
17538 060000 012737 057662 000024          MOV    #$PWRDN,@#PWRVEC ;;SET UP THE POWER DOWN VECTOR
17539 060006 012737 000340 000026          MOV    #340,@#PWRVEC+2 ;;PRIO:7
17540 060014 012737 062434 000550          MOV    #PWRMSG,MSGADR
17541 060022 104401          TYPE                                ;;REPORT THE POWER FAILURE
17542 060024 012716          MOV    (PC)+,(SP)  ;;RESTART AT PHALT
17543 060026 060040          $PWRAD: .WORD    PHALT ;;RESTART ADDRESS
17544 060030 000002          RTI
17545 060032 000000          $ILLUP: HALT      ;;THE POWER UP SEQUENCE WAS STARTED
17546 060034 000776          BR     .-2        ;;BEFORE THE POWER DOWN WAS COMPLETE
17547 060036 000000          $SAVR6: 0        ;;PUT THE SP HERE
17548
17549 060040
17550 060040 000137 001524          PHALT: JMP     @#BEGIN ;RESTART THE TEST AT THE BEGINNING
17551
17552          ;*****
17553          ;SBTTL  ERROR HANDLER ROUTINE
17554          ;*****
17555          ;*THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
17556          ;*SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
17557          ;*AND GO TO $ERRTYP ON ERROR
17558          ;*THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
17559          ;*SW15=1      HALT ON ERROR
17560          ;*SW13=1      INHIBIT ERROR TYPEOUTS
17561
17562          ERROR:
17563          060044 104406          CKSWR                                ;;TEST FOR CHANGE IN SOFT-SWR
17564 060046 005237 000456          7$:   INC    ERRFLG    ;;SET THE ERROR FLAG
17565 060052 001775          BEQ    7$           ;;DON'T LET THE FLAG GO TO ZERO
17566 060054 013777 000402 120346          MOV    $TSTNM,@DISPLAY ;;DISPLAY TEST NUMBER AND ERROR FLAG
17567 060062 005237 000414          1$:   INC    $ERTL     ;;COUNT THE NUMBER OF ERRORS
17568 060066 011637 000420          MOV    (SP',$ERRPC    ;;GET ADDRESS OF ERROR INSTRUCTION
17569 060072 162737 000004 000420          SUB    #4,$ERRPC
17570 060100 032777 020000 120320          BIT    #BIT13,@SWR   ;;SKIP TYPEOUT IF SET
17571 060106 001006          BNE    20$         ;;SKIP TYPEOUTS
17572 060110 004737 060200          JSR    PC,$ERRTYP    ;;GO TO USER ERROR ROUTINE
17573 060114 012737 000443 000550          MOV    #SCLRF,MSGADR
17574 060122 104401          TYPE
17575 060124
17576 060124 122737 000001 000652          20$:  LMPB   #APTENV,$ENV  ;;RUNNING IN APT MODE
17577 060132 001007          BNE    2$           ;;NO,SKIP APT ERROR REPORT
17578 060134 113737 000552 060146          MOVB   ERR,21$     ;;SET ITEM NUMBER AS ERROR NUMBER
17579 060142 004737 061746          JSR    PC,$ATY4    ;;REPORT FATAL ERROR TO APT
17580 060146 000          21$:  .BYTE  0
17581 060147 000          .BYTE  0
```

```

17582 060150 000777
17583 060152 005777 120250
17584 060156 100002
17585 060160 000000
17586 060162 104406
17587 060164
17588 060164 022737 056432 000042
17589 060172 001001
17590 060174 000000
17591 060176
17592 060176 000207
17593
17594
17595
17596
17597
17598
17599
17600
17601 060200
17602 060200 012737 062604 000550
17603 060206 104401
17604 060210 013737 000552 000474
17605 060216 005337 000474
17606 060222 006337 000474
17607 060226 062737 060360 000474
17608 060234 017737 120234 000550
17609 060242 104401
17610 060244 012737 062464 000550
17611 060252 104401
17612 060254 013746 000420
17613 060260 104402
17614 060262 013737 000546 000474
17615 060270 162737 000004 000474
17616 060276 006337 000474
17617 060302 062737 060366 000474
17618 060310 017737 120160 000550
17619 060316 104401
17620 060320 012737 000443 000550
17621 060326 104401
17622 060330 005737 000446
17623 060334 001004
17624 060336 012737 062772 000550
17625 060344 000403
17626 060346 012737 063032 000550
17627 060354 104401
17628 060356 000207
17629
17630 060360 062645
17631 060362 062676
17632 060364 062725
17633
17634 060366 062466
17635 060370 062503
17636 060372 062520
17637 060374 062535
    
```

```

22$: BR 22$ ;;APT ERROR LOOP
2$: TST @SWR ;;HALT ON ERROR
    BPL 3$ ;;SKIP IF CONTINUE
    HALT ;;HALT ON ERROR!
    CKSWR ;;TEST FOR CHANGE IN SOFT-SWR
3$: CMP #SENDAD,@#42 ;;ACT-11 AUTO-ACCEPT?
    BNE 6$ ;;BRANCH IF NO
    HALT ;;YES
6$: RTS PC ;;RETURN

;*****
;SBTTL ERROR MESSAGE TYPEOUT ROUTINE
;*****
;*THIS ROUTINE USES THE 'ERROR TYPE NUMBER' (ERR) TO DETERMINE WHICH
;*ERROR IS TO BE REPORTED. IT THEN OBTAINS, FROM THE 'ERROR TABLE' ($ERRTB),
;*AND REPORTS THE APPROPRIATE INFORMATION CONCERNING THE ERROR.

$ERRTYP:
    MOV #ERRHDR,MSGADR ;TYPE ERROR HEADER
    TYPE
    MOV ERR,ERRTMP ;TYPE APPROPRIATE ERROR MESSAGE
    DEC ERRTMP
    ASL ERRTMP
    ADD #ERRTBL,FRRTMP
    MOV @ERRTMP,MSGADR
    TYPE
    MOV #SPACE1,MSGADR ;TYPE ONE SPACE
    TYPE
    MOV $ERRPC,-(SP) ;TYPE THE ERROR PC
    TYPOC
    MOV CTL,ERRTMP ;TYPE FAILING CONTROL CHIP NUMBER
    SUB #4,ERRTMP
    ASL ERRTMP
    ADD #CTLTBL,ERRTMP
    MOV @ERRTMP,MSGADR
    TYPE
    MOV #SCLRF,MSGADR ;TYPE CARRIAGE-RETURN/LINE-FEED
    TYPE
    TST @MMCYL ;IF MEMORY MANAGEMENT NOT ENABLED
    BNE 1$ ; THEN TYPE OUT
    MOV #MMDIS,MSGADR ; MEMORY MANAGEMENT DISABLED MESS
    BR 2$
1$: MOV #MMENA,MSGADR ; ELSE TYPE OUT MEMORY MANAGEMENT ENABLED
2$: RTS PC

ERRTBL: .WORD ERR1 ;ADDRESSES OF ERROR MESSAGES
        .WORD ERR2
        .WORD ERR3

CTLTBL: .WORD CTL4 ;ADDRESSES OF CHIP NUMBER MESSAGES
        .WORD CTL5
        .WORD CTL6
        .WORD CTL7
    
```



```

17638 060376 062552          .WORD  CTL8
17639 060400 062567          .WORD  CTL9
17640
17641
17642
17643
17644
17645
17646
17647
17648
17649
17650
17651
17652 060402 105737 000435    $TYPE:  TSTB   $TPFLG   ;; IS THERE A TERMINAL?
17653 060406 100002          BPL     1$         ;; BR IF YES
17654 060410 000000          HALT    ;         ;; HALT HERE IF NO TERMINAL
17655 060412 000425          BR      3$         ;; LEAVE
17656 060414 010046          1$:    MOV     RO,-(SP)  ;; SAVE RO
17657 060416 013700 000550    MOV     MSGADR,RO   ;; GET ADDRESS OF ASCIZ STRING
17658 060422 122737 000001 000652    CMPB   #APTENV,$ENV  ;; RUNNING IN APT MODE
17659 060430 001006          BNE     62$        ;; NO,GO CHECK FOR APT CONSOLE
17660 060432 132737 000100 000653    BITB   #APTPOOL,$ENVM  ;; SPOOL MESSAGE TO APT
17661 060440 001402          BEQ     62$        ;; NO,GO CHECK FOR CONSOLE
17662 060442 004737 061736    JSR    PC,$ATY3     ;; SPOOL MESSAGE TO APT
17663 060446 132737 000040 000653    62$:   BITB   #APTCSUP,$ENVM  ;; APT CONSOLE SUPPRESSED
17664 060454 001003          BNE     60$        ;; YES,SKIP TYPE OUT
17665 060456 112046          2$:    MOVB   (RO)+,-(SP)  ;; PUSH CHARACTER TO BE TYPED ONTO STACK
17666 060460 001003          BNE     4$         ;; BR IF IT ISN'T THE TERMINATOR
17667 060462 005726          TST    (SP)+       ;; IF TERMINATOR POP IT OFF THE STACK
17668 060464 012600          60$:   MOV     (SP)+,RO     ;; RESTORE RO
17669 060466 000002          3$:    RTI          ;; RETURN
17670 060470 122716 000011    4$:    CMPB   #HT,(SP)    ;; BRANCH IF <HT>
17671 060474 001432          BEQ     8$         ;; BRANCH IF NOT <CRLF>
17672 060476 122716 000200    CMPB   #CRLF,(SP)
17673 060502 001010          BNE     5$         ;; POP <CR><LF> EQUIV
17674 060504 005726          TST    (SP)+
17675 060506 012737 000443 000550    MOV     #CRLF,MSGADR
17676 060514 104401          TYPE   ;         ;; TYPE A CR AND LF
17677 060516 105037 060652    CLRB   $CHARCNT    ;; CLEAR CHARACTER COUNT
17678 060522 000755          BR     2$         ;; GET NEXT CHARACTER
17679 060524 004737 060606    5$:    JSR    PC,$TYPEC    ;; GO TYPE THIS CHARACTER
17680 060530 123726 000434    6$:    CMPB   $FILLC,(SP)+  ;; IS IT TIME FOR FILLER CHARS.?
17681 060534 001350          BNE     2$         ;; IF NO GO GET NEXT CHAR.
17682 060536 013746 000432    MOV     $NULL,-(SP)  ;; GET # OF FILLER CHARS. NEEDED
17683
17684 060542 105366 000001    7$:    DECB   1(SP)      ;; AND THE NULL CHAR.
17685 060546 002770          BLT    6$         ;; DOES A NULL NEED TO BE TYPED?
17686 060550 004737 060606    JSR    PC,$TYPEC    ;; BR IF NO--GO POP THE NULL OFF OF STACK
17687 060554 105337 060652    DECB   $CHARCNT    ;; GO TYPE A NULL
17688 060560 000770          BR     7$         ;; DO NOT COUNT AS A COUNT
17689
17690
17691
17692 060562 112716 000040    ;HORIZONTAL TAB PROCESSOR
17693 060566 004737 060606    8$:    MOVB   #' ,(SP)    ;; REPLACE TAB WITH SPACE
17693 060566 004737 060606    9$:    JSR    PC,$TYPEC    ;; TYPE A SPACE
    
```

```

17694 060572 132737 C00007 060652 BITB #7,$CHARCNT ;;BRANCH IF NOT AT
17695 060600 001372 BNE 9$ ;;TAB STOP
17696 060602 005726 TST (SP)+ ;;POP SPACE OFF STACK
17697 060604 000724 BR 2$ ;;GET NEXT CHARACTER
17698 060606 105737 177564 $TYPEPC: TSTB @#TPS ;;WAIT UNTIL PRINTER IS READY
17699 060612 100375 BPL $TYPEPC
17700 060614 116637 000002 177566 MOVB 2(SP),@#TPB ;;LOAD CHAR TO BE TYPED INTO DATA REG.
17701 060622 122766 000015 000002 CMPB #CR,2(SP) ;;IS CHARACTER A CARRIAGE RETURN?
17702 060630 001003 BNE 1$ ;;BRANCH IF NO
17703 060632 105037 060652 CLRB $CHARCNT ;;YES--CLEAR CHARACTER COUNT
17704 060636 000406 BR $TYPEX ;;EXIT
17705 060640 122766 000012 000002 1$: CMPB #LF,2(SP) ;;IS CHARACTER A LINE FEED?
17706 060646 001402 BEQ $TYPEX ;;BRANCH IF YES
17707 060650 105227 INCB (PC)+ ;;COUNT THE CHARACTER
17708 060652 000000 $CHARCNT: .WORD 0 ;;CHARACTER COUNT STORAGE
17709 060654 000207 $TYPEX: RTS PC

*****
.SBTTL BINARY TO OCTAL (ASCII) AND TYPE
*****
*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
*OCTAL (ASCII) NUMBER AND TYPE IT.
*$TYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
*CALL:
* MOV NUM,-(SP) ;;NUMBER TO BE TYPED
* TYPOS ;;CALL FOR TYPEOUT
* .BYTE N ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
* .BYTE M ;;M=1 OR 0
* ;;1=TYPE LEADING ZEROS
* ;;0=SUPPRESS LEADING ZEROS
*$TYPON---ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
*$TYPOS OR $TYPOC
*CALL:
* MOV NUM,-(SP) ;;NUMBER TO BE TYPED
* TYPON ;;CALL FOR TYPEOUT
*$TYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER
*CALL:
* MOV NUM,-(SP) ;;NUMBER TO BE TYPED
* TYPOC ;;CALL FOR TYPEOUT
17736 060656 017646 000000 $TYPOS: MOV @ (SP),-(SP) ;;PICKUP THE MODE
17737 060662 116637 000001 061105 MOVB 1(SP),$OFILL ;;LOAD ZERO FILL SWITCH
17738 060670 112637 061107 MOVB (SP)+,$OMODE+1 ;;NUMBER OF DIGITS TO TYPE
17739 060674 062716 000002 ADD #2,(SP) ;;ADJUST RETURN ADDRESS
17740 060700 000406 BR $TYPON
17741 060702 112737 000001 061105 $TYPOC: MOVB #1,$OFILL ;;SET THE ZERO FILL SWITCH
17742 060710 112737 000006 061107 MOVB #6,$OMODE+1 ;;SET FOR SIX(6) DIGITS
17743 060716 112737 000005 061104 $TYPON: MOVB #5,$OCNT ;;SET THE ITERATION COUNT
17744 060724 010346 MOV R3,-(SP) ;;SAVE R3
17745 060726 010446 MOV R4,-(SP) ;;SAVE R4
17746 060730 010546 MOV R5,-(SP) ;;SAVE R5
17747 060732 113704 061107 MOVB $OMODE+1,R4 ;;GET THE NUMBER OF DIGITS TO TYPE
17748 060736 005404 NEG R4
17749 060740 062704 000006 ADD #6,R4 ;;SUBTRACT IT FOR MAX. ALLOWED
    
```

```

17750 060744 110437 061106      MOVB   R4,$OMODE      ;;SAVE IT FOR USE
17751 060750 113704 061105      MOVB   $OFILL,R4     ;;GET THE ZERO FILL SWITCH
17752 060754 016605 000012      MOV    12(SP),R5     ;;PICKUP THE INPUT NUMBER
17753 060760 005003              CLR    R3            ;;CLEAR THE OUTPUT WORD
17754 060762 006105      1$:   ROL    R5            ;;ROTATE MSB INTO 'C'
17755 060764 000404              BR     3$           ;;GO DO MSB
17756 060766 006105      2$:   ROL    R5            ;;FORM THIS DIGIT
17757 060770 006105              ROL    R5
17758 060772 006105              ROL    R5
17759 060774 010503              MOV    R5,R3
17760 060776 006103      3$:   ROL    R3            ;;GET LSB OF THIS DIGIT
17761 061000 105337 061106      DECB   $OMODE        ;;TYPE THIS DIGIT?
17762 061004 100020              BPL   7$           ;;BR IF NO
17763 061006 042703 177770      BIC    #177770,R3    ;;GET RID OF JUNK
17764 061012 001002              BNE   4$           ;;TEST FOR 0
17765 061014 005704              TST   R4            ;;SUPPRESS THIS 0?
17766 061016 001403              BEQ   5$           ;;BR IF YES
17767 061020 005204      4$:   INC    R4            ;;DON'T SUPPRESS ANYMORE 0'S
17768 061022 052703 000060      BIS    #'0,R3       ;;MAKE THIS DIGIT ASCII
17769 061026 052703 000040      5$:   BIS    #' ,R3     ;;MAKE ASCII IF NOT ALREADY
17770 061032 110337 061102      MOVB   R3,8$        ;;SAVE FOR TYPING
17771 061036 012737 061102 000550  MOV    #8$,MSGADR
17772 061044 104401              TYPE                ;;GO TYPE THIS DIGIT
17773 061046 105337 061104      7$:   DECB   $OCNT     ;;COUNT BY 1
17774 061052 003345              BGT   2$           ;;BR IF MORE TO DO
17775 061054 002402              BLT   6$           ;;BR IF DONE
17776 061056 005204              INC   R4            ;;INSURE LAST DIGIT ISN'T A BLANK
17777 061060 000742              BR    2$           ;;GO DO THE LAST DIGIT
17778 061062 012605      6$:   MOV    (SP)+,R5    ;;RESTORE R5
17779 061064 012604              MOV    (SP)+,R4    ;;RESTORE R4
17780 061066 012603              MOV    (SP)+,R3    ;;RESTORE R3
17781 061070 016666 000002 000004  MOV    2(SP),4(SP)  ;;SET THE STACK FOR RETURNING
17782 061076 012616              MOV    (SP)+,(SP)
17783 061100 000002              RTI                ;;RETURN
17784 061102 000          8$:   .BYTE  0            ;;STORAGE FOR ASCII DIGIT
17785 061103 000          .BYTE  0            ;;TERMINATOR FOR TYPE ROUTINE
17786 061104 000          $OCNT: .BYTE  0     ;;OCTAL DIGIT COUNTER
17787 061105 000          $OFILL: .BYTE  0    ;;ZERO FILL SWITCH
17788 061106 000000      $OMODE: .WORD  0    ;;NUMBER OF DIGITS TO TYPE
17789
17790
17791      ;;*****
17792      .SBTTL  -TTY INPUT ROUTINE
17793      ;;*****
17794      .ENABL  LSB
17795
17796      ;;*****
17797      ;*SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
17798      ;*ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
17799      ;*SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
17800      ;*WHEN OPERATING IN TTY FLAG MODE.
17801 061110 022737 000176 000426  $CKSWR: CMP    #SWREG,SWR  ;;IS THE SOFT-SWR SELECTED?
17802 061116 001106              BNE   15$         ;;BRANCH IF NO
17803 061120 105737 177560      TSTB  @#TKS       ;;CHAR THERE?
17804 061124 100103              BPL   15$         ;;IF NO, DON'T WAIT AROUND
17805 061126 113746 177562      MOVB  @#TKB,-(SP)  ;;SAVE THE CHAR

```

17806	061132	042716	177600		BIC	#^C177,(SP)	::STRIP-OFF THE ASCII
17807	061136	022726	000007		CMP	#7,(SP)+	::IS IT A CONTROL G?
17808	061142	001074			BNE	15\$::NO, RETURN TO USER
17809	061144	123727	000422	000001	CMPB	\$AUTOB,#1	::ARE WE RUNNING IN AUTO-MODE?
17810	061152	001470			BEQ	15\$::BRANCH IF YES
17811							
17812	061154	012737	061701	000550	MOV	#\$CNTLG,MSGADR	
17813	061162	104401			TYPE		::ECHO THE CONTROL-C (^G)
17814	061164	012737	061706	000550	\$GTSWR: MOV	#\$MSWR,MSGADR	
17815	061172	104401			TYPE		::TYPE CURRENT CONTENTS
17816	061174	013746	000176		MOV	SWREG,-(SP)	::SAVE SWREG FOR TYPEOUT
17817	061200	104402			TYPOC		::GO TYPE--OCTAL ASCII(ALL DIGITS)
17818	061202	012737	061717	000550	MOV	#\$MNEW,MSGADR	
17819	061210	104401			TYPE		::PROMPT FOR NEW SWR
17820	061212	005046		19\$:	CLR	-(SP)	::CLEAR COUNTER
17821	061214	005046			CLR	-(SP)	::THE NEW SWR
17822	061216	105737	177560	7\$:	TSTB	@#TKS	::CHAR THERE?
17823	061222	100375			BPL	7\$::IF NOT TRY AGAIN
17824							
17825	061224	113746	177562		MOVB	@#TKB,-(SP)	::PICK UP CHAR
17826	061230	042716	177600		BIC	#^C177,(SP)	::MAKE IT 7-BIT ASCII
17827							
17828							
17829							
17830	061234	021627	000025	9\$:	CMP	(SP),#25	::IS IT A CONTROL-U?
17831	061240	001007			BNE	10\$::BRANCH IF NOT
17832	061242	012737	061674	000550	MOV	#\$CNTLU,MSGADR	
17833	061250	104401			TYPE		::YES, ECHO CONTROL-U (^U)
17834	061252	062706	000006	20\$:	ADD	#6,SP	::IGNORE PREVIOUS INPUT
17835	061256	000755			BR	19\$::LET'S TRY IT AGAIN
17836							
17837							
17838	061260	021627	000015	10\$:	CMP	(SP),#15	::IS IT A <CR>?
17839	061264	001024			BNE	16\$::BRANCH IF NO
17840	061266	005766	000004		TST	4(SP)	::YES, IS IT THE FIRST CHAR?
17841	061272	001403			BEQ	11\$::BRANCH IF YES
17842	061274	016677	000002	117124	MOV	2(SP),@SWR	::SAVE NEW SWR
17843	061302	062706	000006	11\$:	ADD	#6,SP	::CLEAR UP STACK
17844	061306	012737	000443	000550	14\$:	MOV	#\$CRLF,MSGADR
17845	061314	104401			TYPE		::ECHO <CR> AND <LF>
17846	061316	123727	000423	000001	CMPB	\$INTAG,#1	::RE-ENABLE TTY KBD INTERRUPTS?
17847	061324	001003			BNE	15\$::BRANCH IF NOT
17848	061326	012737	000100	177560	MOV	#100,@#TKS	::RE-ENABLE TTY KBD INTERRUPTS
17849	061334	000002		15\$:	RTI		::RETURN
17850	061336	004737	060606	16\$:	JSR	PC,\$TYPEC	::ECHO CHAR
17851	061342	021627	000060		CMP	(SP),#60	::CHAR < 0?
17852	061346	002420			BLT	18\$::BRANCH IF YES
17853	061350	021627	000067		CMP	(SP),#67	::CHAR > 7?
17854	061354	003015			BGT	18\$::BRANCH IF YES
17855	061356	042726	000060		BIC	#60,(SP)+	::STRIP-OFF ASCII
17856	061362	005766	000002		TST	2(SP)	::IS THIS THE FIRST CHAR
17857	061366	001403			BEQ	17\$::BRANCH IF YES
17858	061370	006316			ASL	(SP)	::NO, SHIFT PRESENT
17859	061372	006316			ASL	(SP)	::CHAR OVER TO MAKE
17860	061374	006316			ASL	(SP)	::ROOM FOR NEW ONE.
17861	061376	005266	000002	17\$:	INC	2(SP)	::KEEP COUNT OF CHAR

```

17862 061402 056616 177776          BIS      -2(SP),(SP)      ;;SET IN NEW CHAR
17863 061406 000703                    BR       7$             ;;GET THE NEXT ONE
17864 061410 012737 000442 000550 18$:  MOV     #$QUES,MSGADR
17865 061416 104401                    TYPE
17866 061420 000714                    BR       20$           ;;SIMULATE CONTROL-U
17867          .DSABL  LSB
17868
17869
17870          ;;*****
17871          ;;*THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
17872          ;;*CALL:
17873          ;;*      RDCHR          ;;INPUT A SINGLE CHARACTER FROM THE TTY
17874          ;;*      RETURN HERE   ;;CHARACTER IS ON THE STACK
17875          ;;*                  ;;WITH PARITY BIT STRIPPED OFF
17876          ;;
17877
17878 061422 011646          $RDCHR: MOV     (SP),-(SP)      ;;PUSH DOWN THE PC
17879 061424 016666 000004 000002  MOV     4(SP),2(SP)      ;;SAVE THE PS
17880 061432 105737 177560 1$:      TSTB   @#TKS          ;;WAIT FOR
17881 061436 100375          BPL     1$             ;;A CHARACTER
17882 061440 113766 177562 000004  MOVB   @#TKB,4(SP)      ;;READ THE TTY
17883 061446 042766 177600 000004  BIC    #^C<177>,4(SP)  ;;GET RID OF JUNK IF ANY
17884 061454 026627 000004 000023  CMP    4(SP),#23       ;;IS IT A CONTROL-S?
17885 061462 001013          BNE    3$             ;;BRANCH IF NO
17886 061464 105737 177560 2$:      TSTB   @#TKS          ;;WAIT FOR A CHARACTER
17887 061470 100375          BPL     2$             ;;LOOP UNTIL ITS THERE
17888 061472 113746 177562          MOVB   @#TKB,-(SP)     ;;GET CHARACTER
17889 061476 042716 177600          BIC    #^C177,(SP)    ;;MAKE IT 7-BIT ASCII
17890 061502 022627 000021          CMP    (SP)+,#21      ;;IS IT A CONTROL-Q?
17891 061506 001366          BNE    2$             ;;IF NOT DISCARD IT
17892 061510 000750          BR     1$             ;;YES, RESUME
17893 061512 026627 000004 000140 3$:      CMP    4(SP),#140     ;;IS IT UPPER CASE?
17894 061520 002407          BLT    4$             ;;BRANCH IF YES
17895 061522 026627 000004 000175  CMP    4(SP),#175     ;;IS IT A SPECIAL CHAR?
17896 061530 003003          BGT    4$             ;;BRANCH IF YES
17897 061532 042766 000040 000004  BIC    #40,4(SP)      ;;MAKE IT UPPER CASE
17898 061540 000002 4$:      RTI                    ;;GO BACK TO USER
17899          ;;*****
17900          ;;*THIS ROUTINE WILL INPUT A STRING FROM THE TTY
17901          ;;*CALL:
17902          ;;*      RDLIN          ;;INPUT A STRING FROM THE TTY
17903          ;;*      RETURN HERE   ;;ADDRESS OF FIRST CHARACTER WILL BE ON THE STACK
17904          ;;*                  ;;TERMINATOR WILL BE A BYTE OF ALL 0'S
17905          ;;
17906 061542 010346          $RDLIN: MOV     R3,-(SP)      ;;SAVE R3
17907 061544 012703 061664 1$:      MOV     #$TTYIN,R3     ;;GET ADDRESS
17908 061550 022703 061674 2$:      CMP     #$TTYIN+8.,R3  ;;BUFFER FULL?
17909 061554 101405          BLOS   4$             ;;BR IF YES
17910 061556 100407          RDCHR          ;;GO READ ONE CHARACTER FROM THE TTY
17911 061560 112613          MOVB   (SP)+,(R3)     ;;GET CHARACTER
17912 061562 122713 000177 10$:     CMPB   #177,(R3)      ;;IS IT A RUBOUT
17913 061566 001005          BNE    3$             ;;SKIP IF NOT
17914 061570 013737 000442 000550 4$:      MOV     $QUES,MSGADR
17915 061576 104401          TYPE
17916 061600 000761          BR     1$             ;;CLEAR THE BUFFER AND LOOP
17917 061602 111337 061662 3$:      MOVB   (R3),9$        ;;ECHO THE CHARACTER

```


17974 062100 005737 000632
 17975 062104 001375
 17976 062106 017637 000004 000634
 17977 062114 062766 000002 000004
 17978 062122 005237 000632
 17979 062126 105037 062152
 17980 062132 105037 062151
 17981 062136 105037 062150
 17982 062142 012601
 17983 062144 012600
 17984 062146 000207
 17985 062150 000
 17986 062151 000
 17987 062152 000
 17988 062154
 17989 000200
 17990 000001
 17991 000100
 17992 000040
 17993
 17994
 17995
 17996
 17997
 17998
 17999
 18000
 18001
 18002
 18003 062154 010046
 18004 062156 016600 000002
 18005 062162 005740
 18006 062164 111000
 18007 062166 006300
 18008 062170 016000 062210
 18009 062174 000200
 18010
 18011
 18012
 18013
 18014 062176 011646
 18015 062200 016666 000004 000002
 18016 062206 000002
 18017
 18018
 18019
 18020
 18021
 18022
 18023
 18024
 18025 062210 062176
 18026 062212 060402
 18027 062214 060702
 18028 062216 060656
 18029 062220 060716

```

11$: TST $MSGTYPE      ;;FINISHED LAST MESSAGE?
      BNE 11$          ;;IF NOT: WAIT
      MOV @4(SP), $FATAL ;;GET ERROR #
      ADD #2, 4(SP)    ;;BUMP RETURN ADDR.
      INC $MSGTYPE     ;;TELL APT TO TAKE ERROR
12$: CLR $FFLG         ;;CLEAR FATAL FLAG
      CLR $LFLG        ;;CLEAR LOG FLAG
      CLR $MFLG        ;;CLEAR MESSAGE FLAG
      MOV (SP)+, R1    ;;POP STACK INTO R1
      MOV (SP)+, R0    ;;POP STACK INTO R0
      RTS PC          ;;RETURN
      $MFLG: .BYTE 0   ;;MESSG. FLAG
      $LFLG: .BYTE 0   ;;LOG FLAG
      $FFLG: .BYTE 0   ;;FATAL FLAG
      .EVEN
  
```

```

APTSIZE=200
APTENV=001
APTSPool=100
APTCSUP=040
  
```

```

*****
.SBTTL TRAP DECODER
*****
*THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE 'TRAP' INSTRUCTION
*AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
*OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
*GO TO THAT ROUTINE.
  
```

```

$TRAP: MOV R0, -(SP)      ;;SAVE R0
        MOV 2(SP), R0    ;;GET TRAP ADDRESS
        TST -(R0)        ;;BACKUP BY 2
        MOV B(R0), R0    ;;GET RIGHT BYTE OF TRAP
        ASL R0           ;;POSITION FOR INDEXING
        MOV $TRPAD(R0), R0 ;;INDEX TO TABLE
        RTS R0          ;;GO TO ROUTINE
  
```

;;THIS IS USE TO HANDLE THE 'GETPRI' MACRO

```

$TRAP2: MOV (SP), -(SP)  ;;MOVE THE PC DOWN
         MOV 4(SP), 2(SP) ;;MOVE THE PSW DOWN
         RTI             ;;RESTORE THE PSW
  
```

.SBTTL TRAP TABLE

```

*THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
*BY THE 'TRAP' INSTRUCTION.
  
```

```

; ROUTINE
; -----
$TRPAD: .WORD $TRAP2
        $TYPE  ;;CALL=TYPE TRAP+1(104401) TTY TYPEOUT ROUTINE
        $TYPOC ;;CALL=TYPOC TRAP+2(104402) TYPE OCTAL NUMBER (WITH LEADING ZEROS)
        $TYPOS ;;CALL=TYPOS TRAP+3(104403) TYPE OCTAL NUMBER (NO LEADING ZEROS)
        $TYPON ;;CALL=TYPON TRAP+4(104404) TYPE OCTAL NUMBER (AS PER LAST CALL)
  
```

18030	062222	061164			\$GTSWR	::CALL=GTSWR	TRAP+5(104405)	GET SOFT-SWR SETTING
18031	062224	061110			\$CKSWR	::CALL=CKSWR	TRAP+6(104406)	TEST FOR CHANGE IN SOFT-SWR
18032	062226	061422			\$RDCHR	::CALL=RDCHR	TRAP+07(104407)	TTY TYPEIN CHARACTER ROUTINE
18033	062230	061542			\$RDLIN	::CALL=RDLIN	TRAP+10(104410)	TTY TYPEIN STRING ROUTINE

18034
18035
18036
18037

```

:*****
:SBTTL TELETYPE MESSAGES
:*****
TITLE: .ASCIZ <200>/CJKDHA KEF11-B CIS DIAGNOSTIC/<200>

```

18038 062232 041600 045512 044104
18039 062240 020101 042513 030506
18040 062246 026461 020102 044503
18041 062254 020123 044504 043501
18042 062262 047516 052123 041511
18043 062270 000200
18044 062272 040600 020116 047125
18045 062300 054105 042520 052103
18046 062306 042105 052040 046511
18047 062314 026505 052517 020124
18048 062322 051124 050101 047440
18049 062330 041503 051125 042522
18050 062336 000104
18051 062340 040600 020116 047125
18052 062346 054105 042520 052103
18053 062354 042105 044440 046114
18054 062362 043505 046101 023040
18055 062370 051040 051505 051105
18056 062376 042526 020104 047111
18057 062404 052123 052522 052103
18058 062412 047511 020116 051124
18059 062420 050101 047440 041503
18060 062426 051125 042522 000104
18061 062434 050200 053517 051105
18062 062442 043040 044501 052514
18063 062450 042522 047440 041503
18064 062456 051125 042522 000104
18065 062464 000040
18066 062466 020040 020040 020040
18067 062474 031462 025055 032060
18068 062502 000
18069 062503 040 020040 020040
18070 062510 031040 026463 030052
18071 062516 000065
18072 062520 020040 020040 020040
18073 062526 031462 025055 033060
18074 062534 000
18075 062535 040 020040 020040
18076 062542 031040 026463 030052
18077 062550 000067
18078 062552 020040 020040 020040
18079 062560 031462 025055 034060
18080 062566 000
18081 062567 040 020040 020040
18082 062574 031040 026463 030052
18083 062602 000071
18084
18085

TIMMSG: .ASCIZ <200>/AN UNEXPECTED TIME-OUT TRAP OCCURRED/

ILLMSG: .ASCIZ <200>/AN UNEXPECTED ILLEGAL & RESERVED INSTRUCTION TRAP OCCURRED/

PWRMSG: .ASCIZ <200>/POWER FAILURE OCCURRED/

```

SPACE1: .ASCIZ / /
CTL4: .ASCIZ / / 23-*04/
CTL5: .ASCIZ / / 23-*05/
CTL6: .ASCIZ / / 23-*06/
CTL7: .ASCIZ / / 23-*07/
CTL8: .ASCIZ / / 23-*08/
CTL9: .ASCIZ / / 23-*09/

```


18086
18087
18088
18089 062604 100200 051105 047522
18090 062612 020122 041520 020040
18091 062620 020040 052103 020114
18092 062626 044103 050111 021440
18093 062634 020040 020040 020040
18094 062642 020040 000
18095 062645 103 051511 044440
18096 062652 051516 051124 041525
18097 062660 044524 047117 043040
18098 062666 044501 052514 042522
18099 062674 000200
18100 062676 044503 020123 047111
18101 062704 042524 051122 050125
18102 062712 020124 040506 046111
18103 062720 051125 100105 000
18104 062725 116 020117 051124
18105 062732 050101 047440 020116
18106 062740 046111 042514 040507
18107 062746 020114 044503 020123
18108 062754 046103 051501 020123
18109 062762 050117 047503 042504
18110 062770 000200

:::*****
:SBTTL TEST ERROR MESSAGES
:::*****

ERRHDR: .ASCIZ <200><200>/ERROR PC CTL CHIP # /

ERR1: .ASCIZ /CIS INSTRUCTION FAILURE/<200>

ERR2: .ASCIZ /CIS INTERRUPT FAILURE/<200>

ERR3: .ASCIZ /NO TRAP ON ILLEGAL CIS CLASS OPCODE/<200>

18111
18112
18113
18114
18115
18116 062772 042515 047515 054522
18117 063000 046440 047101 043501
18118 063006 046505 047105 020124
18119 063014 040527 020123 044504
18120 063022 040523 046102 042105
18121 063030 000200
18122 063032 042515 047515 054522
18123 063040 046440 047101 043501
18124 063046 046505 047105 020124
18125 063054 040527 020123 047105
18126 063062 041101 042514 100104
18127 063070 000

:::*****
:SBTTL MEMORY MANAGEMENT MESSAGES
:::*****

MMDIS: .ASCIZ/MEMORY MANAGEMENT WAS DISABLED/<200>

MMENA: .ASCIZ/MEMORY MANAGEMENT WAS ENABLED/<200>

18128
18129
18130
18131 063072
18132
18133

.EVEN

18134
18135
18136
18137
18138 063072 000100
18139 063172 000100
18140 063272 000400
18141 063672 000400

:::*****
:SBTTL BUFFERS & TABLES
:::*****

BUFF1: .BLKB 100 ;CHARACTER STRING BUFFER 1
BUFF2: .BLKB 100 ;CHARACTER STRING BUFFER 2
TRANS: .BLKB 400 ;TRANSLATION TABLE
CHRSET: .BLKB 400 ;CHARACTER SET TABLE

```

18142
18143
18144      064273      064273      .ODD
18145      064273      000          .BYTE 0          ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
18146      064274      000040      DSBUF1: .BLKB 40      ;DECIMAL STRING BUFFER 1
18147
18148      064335      064335      .ODD
18149      064335      000          .BYTE 0          ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
18150      064336      000040      DSBUF2: .BLKB 40      ;DECIMAL STRING BUFFER 2
18151
18152      064377      064377      .ODD
18153      064377      000          .BYTE 0          ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
18154      064400      000040      DS.DST: .BLKB 40      ;DESTINATION DECIMAL STRING BUFFER
18155
18156      064441      064441      .ODD
18157      064441      000          .BYTE 0          ;RESERVED FOR SIGN FOR LEADING SEPARATE STRINGS
18158      064442      000040      DS.EXP: .BLKB 40      ;DECIMAL STRING BUFFER OF EXPECTED RESULTS
18159
18160
18161      ;*****
18162      ;SBITL TEST CASES FOR CMPN AND CMPP
18163      ;*****
18164      ;
18165
18166      064502      000007      TCMP:   .WORD 7          ;# OF TEST CASES FOR CMPN & CMPP
18167
18168      064504      000000      .WORD 0          ;SRC1 LENGTH
18169      064506      064574      .WORD CMP1       ;SRC1 POINTER
18170      064510      000000      .WORD 0          ;SRC2 LENGTH
18171      064512      064600      .WORD CMP1A      ;SRC2 POINTER
18172
18173      064514      000007      .WORD 7          ;TEST CASE NUMBER 2
18174      064516      064604      .WORD CMP2       ;
18175      064520      000000      .WORD 0          ;
18176      064522      064616      .WORD CMP2A      ;
18177
18178      064524      000000      .WORD 0          ;TEST CASE NUMBER 3
18179      064526      064622      .WORD CMP3       ;
18180      064530      000020      .WORD 20         ;
18181      064532      064626      .WORD CMP3A      ;
18182
18183      064534      000011      .WORD 11         ;TEST CASE NUMBER 4
18184      064536      064650      .WORD CMP4       ;
18185      064540      000011      .WORD 11         ;
18186      064542      064664      .WORD CMP4A      ;
18187
18188      064544      000006      .WORD 6          ;TEST CASE NUMBER 5
18189      064546      064700      .WORD CMP5       ;
18190      064550      000006      .WORD 6          ;
18191      064552      064710      .WORD CMP5A      ;
18192
18193      064554      000001      .WORD 1          ;TEST CASE NUMBER 6
18194      064556      064720      .WORD CMP6       ;
18195      064560      000001      .WORD 1          ;
18196      064562      064724      .WORD CMP6A      ;
18197
    
```

Line No	Code	Value	Unit	Sub-Unit	Label	Format	Sign	Description
18198	064564	000006			.WORD	6		:TEST CASE NUMBER 7
18199	064566	064730			.WORD	CMP7		
18200	064570	000005			.WORD	5		
18201	064572	064740			.WORD	CMP7A		
18202								
18203								
18204	064574	000000			CMP1: .WORD	0		:SRC1 SIGN
18205	064576	000			.BYTE			:DECIMAL STRING FOR SRC1
18206		064600			.EVEN			
18207	064600	000000			CMP1A: .WORD	0		:SRC2 SIGN
18208	064602	000			.BYTE			:DECIMAL STRING FOR SRC2
18209		064604			.EVEN			
18210								
18211	064604	000000			CMP2: .WORD	0		:SRC1 SIGN
18212	064606	007	006	005	.BYTE	007,006,005,004,003,002,001		:DECIMAL STRING FOR SRC1
18213	064611	004	003	002				
18214	064614	001						
18215		064616			.EVEN			
18216	064616	000000			CMP2A: .WORD	0		:SRC2 SIGN
18217	064620	000			.BYTE			:DECIMAL STRING FOR SRC2
18218		064622			.EVEN			
18219								
18220	064622	177777			CMP3: .WORD	-1		:SRC1 SIGN
18221	064624	000			.BYTE			:DECIMAL STRING FOR SRC1
18222		064626			.EVEN			
18223	064626	177777			CMP3A: .WORD	-1		:SRC2 SIGN
18224	064630	001	000	011	.BYTE	001,000,011,000,002,000,010,000,003,000,007,000,004,000,004,000		
18225	064633	000	002	000				
18226	064636	010	000	003				
18227	064641	000	007	000				
18228	064644	004	000	004				
18229	064647	000						
18230								:DECIMAL STRING FOR SRC2
18231					.EVEN			
18232								
18233	064650	177777			CMP4: .WORD	-1		:SRC1 SIGN
18234	064652	001	002	003	.BYTE	001,002,003,000,005,006,007,010,011		:DECIMAL STRING FOR SRC1
18235	064655	000	005	006				
18236	064660	007	010	011				
18237		064664			.EVEN			
18238	064664	177777			CMP4A: .WORD	-1		:SRC2 SIGN
18239	064666	001	002	003	.BYTE	001,002,003,004,005,006,007,010,011		:DECIMAL STRING FOR SRC2
18240	064671	004	005	006				
18241	064674	007	010	011				
18242		064700			.EVEN			
18243								
18244	064700	000000			CMP5: .WORD	0		:SRC1 SIGN
18245	064702	002	001	004	.BYTE	002,001,004,011,040,007		:DECIMAL STRING FOR SRC1
18246	064705	011	040	007				
18247					.EVEN			
18248	064710	000000			CMP5A: .WORD	0		:SRC2 SIGN
18249	064712	002	001	004	.BYTE	002,001,004,011,040,007		:DECIMAL STRING FOR SRC2
18250	064715	011	040	007				
18251					.EVEN			
18252								
18253	064720	177777			CMP6: .WORD	-1		:SRC1 SIGN

18254	064722	004				.BYTE	004		;DECIMAL STRING FOR SRC1
18255		064724				.EVEN			
18256	064724	000000				CMP6A: .WORD	0		;SRC2 SIGN
18257	064726	004				.BYTE	004		;DECIMAL STRING FOR SRC2
18258		064730				.EVEN			
18259									
18260	064730	000000				CMP7: .WORD	0		;SRC1 SIGN
18261	064732	002	003	004		.BYTE	002,003,004,005,006,007		;DECIMAL STRING FOR SRC1
18262	064735	005	006	007					
18263									
18264	064740	000000				CMP7A: .EVEN			
18265	064742	003	004	005		.WORD	0		;SRC2 SIGN
18266	064745	006	007			.BYTE	003,004,005,006,007		;DECIMAL STRING FOR SRC2
18267		064750				.EVEN			
18268									
18269									
18270									
18271									
18272									
18273						.SBTTL	TEST CASES FOR ASHN AND ASHP		
18274						*****			
18275						: TABLE OF TEST CASES FOR ASHN AND ASHP			
18276						*****			
18277									
18278									
18279	064750	000017				TASH: .WORD	15.		;# OF TEST CASES FOR ASHN & ASHP
18280									
18281	064752	000005				.WORD	5		;SRC LENGTH
18282	064754	000005				.WORD	5		;DST LENGTH
18283	064756	000				.BYTE	0		;SHIFT COUNT
18284	064757	000				.BYTE	0		;ROUNDING DIGIT
18285									
18286	064760	000000				.WORD	0		;TEST CASE NUMBER 2
18287	064762	000017				.WORD	17		
18288	064764	000				.BYTE	0		
18289	064765	000				.BYTE	0		
18290									
18291	064766	000032				.WORD	32		;TEST CASE NUMBER 3
18292	064770	000032				.WORD	32		
18293	064772	000				.BYTE	0		
18294	064773	007				.BYTE	7		
18295									
18296	064774	000025				.WORD	25		;TEST CASE NUMBER 4
18297	064776	000025				.WORD	25		
18298	065000	377				.BYTE	-1		
18299	065001	005				.BYTE	5		
18300									
18301	065002	000014				.WORD	14		;TEST CASE NUMBER 5
18302	065004	000014				.WORD	14		
18303	065006	377				.BYTE	-1		
18304	065007	006				.BYTE	6		
18305									
18306	065010	000006				.WORD	6		;TEST CASE NUMBER 6
18307	065012	000006				.WORD	6		
18308	065014	001				.BYTE	1		
18309	065015	006				.BYTE	6		

18310					
18311	065016	000010	.WORD	10	;TEST CASE NUMBER 7
18312	065020	000011	.WORD	11	
18313	065022	001	.BYTE	1	
18314	065023	000	.BYTE	0	
18315					
18316	065024	000005	.WORD	5	;TEST CASE NUMBER 8
18317	065026	000006	.WORD	6	
18318	065030	002	.BYTE	2	
18319	065031	000	.BYTE	0	
18320					
18321	065032	000016	.WORD	16	;TEST CASE NUMBER 9
18322	065034	000004	.WORD	4	
18323	065036	363	.BYTE	-15	
18324	065037	000	.BYTE	0	
18325					
18326	065040	000020	.WORD	20	;TEST CASE NUMBER 10
18327	065042	000010	.WORD	10	
18328	065044	360	.BYTE	-20	
18329	065045	000	.BYTE	0	
18330					
18331	065046	000012	.WORD	12	;TEST CASE NUMBER 11
18332	065050	000007	.WORD	7	
18333	065052	006	.BYTE	6	
18334	065053	000	.BYTE	0	
18335					
18336	065054	000037	.WORD	37	;TEST CASE NUMBER 12
18337	065056	000030	.WORD	30	
18338	065060	030	.BYTE	30	
18339	065061	000	.BYTE	0	
18340					
18341	065062	000020	.WORD	20	;TEST CASE NUMBER 13
18342	065064	000010	.WORD	10	
18343	065066	373	.BYTE	-5	
18344	065067	000	.BYTE	0	
18345					
18346	065070	000005	.WORD	5	;TEST CASE NUMBER 14
18347	065072	000005	.WORD	5	
18348	065074	373	.BYTE	-5	
18349	065075	006	.BYTE	6	
18350					
18351	065076	000007	.WORD	7	;TEST CASE NUMBER 15
18352	065100	000007	.WORD	7	
18353	065102	366	.BYTE	-12	
18354	065103	006	.BYTE	6	
18355					
18356					

18357
 18358
 18359
 18360
 18361
 18362
 18363
 18364
 18365 065104 000006
 18366
 18367 065106 000020
 18368 065110 065262
 18369 065112 000021
 18370 065114 065304
 18371 065116 000024
 18372 065120 065330
 18373 065122 000000
 18374 065124 065356

.SBTTL ADDN/SUBN TEST DATA
 :*****
 :
 : TABLES FOR ADDN AND SUBN
 :*****

TADD:	.WORD	6		;# OF TEST CASES
	.WORD	16.		;SRC1 LENGTH
	.WORD	ADD1		;SRC1 PTR
	.WORD	17.		;SRC2 LENGTH
	.WORD	ADD1A		;SRC2 PTR
	.WORD	20.		;DEST LENGTH
	.WORD	ADD1B		;DEST PTR (SUM)
	.WORD	0		;CC.EXP (SUM)
	.WORD	ADD1C		;DEST PTR (DIFF)

18375	065126	000000	.WORD	0	;CC.EXP (DIFF)
18376					
18377	065130	000012	.WORD	10.	;SRC1 >SRC2, SUM +,DIFF -
18378	065132	065404	.WORD	ADD2	
18379	065134	000012	.WORD	10.	
18380	065136	065420	.WORD	ADD2A	
18381	065140	000012	.WORD	10.	

18382	065142	065434				.WORD	ADD2B	
18383	065144	000000				.WORD	0	
18384	065146	065450				.WORD	ADD2C	
18385	065150	000010				.WORD	10	
18386								
18387	065152	000013				.WORD	11.	:SRC1 +,SRC2 -, SUM & DIFF -
18388	065154	065464				.WORD	ADD3	
18389	065156	000013				.WORD	11.	
18390	065160	065502				.WORD	ADD3A	
18391	065162	000014				.WORD	12.	
18392	065164	065520				.WORD	ADD3B	
18393	065166	000010				.WORD	10	
18394	065170	065536				.WORD	ADD3C	
18395	065172	000010				.WORD	10	
18396								
18397	065174	000020				.WORD	16.	:SRC1 -,SRC2 -,SUM -, DIFF +
18398	065176	065554				.WORD	ADD4	
18399	065200	000003				.WORD	3	
18400	065202	065576				.WORD	ADD4A	
18401	065204	000020				.WORD	16.	
18402	065206	065604				.WORD	ADD4B	
18403	065210	000010				.WORD	10	
18404	065212	065626				.WORD	ADD4C	
18405	065214	000000				.WORD	0	
18406								
18407	065216	000016				.WORD	14.	:SRC1 = +,SRC2 0,DEST TRUNCATED WITH :SUM = +,DIFF = -
18408								
18409	065220	065650				.WORD	ADD5	
18410	065222	000001				.WORD	1	
18411	065224	065670				.WORD	ADD5A	
18412	065226	000011				.WORD	9.	
18413	065230	065674				.WORD	ADD5B	
18414	065232	000002				.WORD	2	
18415	065234	065710				.WORD	ADD5C	
18416	065236	000012				.WORD	12	
18417								
18418	065240	000010				.WORD	8.	:DEST LENGTH = 0
18419	065242	065724				.WORD	ADD6	
18420	065244	000011				.WORD	9.	
18421	065246	065736				.WORD	ADD6A	
18422	065250	000000				.WORD	0	
18423	065252	065752				.WORD	ADD6B	
18424	065254	000006				.WORD	6	
18425	065256	065752				.WORD	ADD6B	
18426	065260	000006				.WORD	6	
18427								
18428	065262	000000				.WORD	0	:SIGN1
18429	065264	005	011	007	ADD1:	.BYTE	5,9.,7,4,3,8.,1,6,6,7,8.,1,2,9.,0,6	
18430	065267	004	003	010				
18431	065272	001	006	006				
18432	065275	007	010	001				
18433	065300	002	011	000				
18434	065303	006						
18435								:SOURCE STRING 1
18436								
18437						.EVEN		

18438	065304	000000				ADD1A:	.WORD	0	:SIGN2
18439	065306	001	007	004			.BYTE	1,7,4,3,2,1,6,9,5,3,4,7,9,4,1,6,2	
18440	065311	003	002	001					
18441	065314	006	011	005					
18442	065317	003	004	007					
18443	065322	011	004	001					
18444	065325	006	002						
18445									:SOURCE STRING 2
18446									
18447		065330							
18448	065330	000000				ADD1B:	.EVEN		
18449	065332	000	000	000			.WORD	0	:SIGN3
18450	065335	002	003	004			.BYTE	0,0,0,2,3,4,0,6,5,5,1,2,0,2	
18451	065340	000	006	005					
18452	065343	005	001	002					
18453	065346	000	002						
18454	065350	006	000	007			.BYTE	6,0,7,0,6,8,	
18455	065353	000	006	010					:DEST STRING
18456									
18457							.EVEN		
18458									
18459	065356	000000				ADD1C:	.WORD	0	
18460	065360	000	000	000			.BYTE	0,0,0,1,1,4,5,7,7,8,7,8,6	
18461	065363	001	001	004					
18462	065366	005	007	007					
18463	065371	010	007	010					
18464	065374	006							
18465	065375	006	011	010			.BYTE	6,9,8,1,2,5,6	
18466	065400	001	002	005					
18467	065403	006							
18468							.EVEN		
18469									
18470	065404	000000				ADD2:	.WORD	0	
18471	065406	003	001	005			.BYTE	3,1,5,9,2,6,5,3,8,0	
18472	065411	011	002	006					
18473	065414	005	003	010					
18474	065417	000							
18475							.EVEN		
18476									
18477	065420	000000				ADD2A:	.WORD	0	
18478	065422	001	007	006			.BYTE	1,7,6,3,2,1,4,1,7,2	
18479	065425	003	002	001					
18480	065430	004	001	007					
18481	065433	002							
18482							.EVEN		
18483									
18484	065434	000000				ADD2B:	.WORD	0	
18485	065436	004	011	002			.BYTE	4,9,2,2,4,7,9,5,5,2	
18486	065441	002	004	007					
18487	065444	011	005	005					
18488	065447	002							

18489							.EVEN	
18490								
18491	065450	177777				ADD2C:	.WORD	-1
18492	065452	001	003	011			.BYTE	1,3,9.,6,0,5,1,2,0,8.
18493	065455	006	000	005				
18494	065460	001	002	000				
18495	065463	010						
18496							.EVEN	
18497								
18498	065464	000000				ADD3:	.WORD	0
18499	065466	004	002	003			.BYTE	4,2,3,1,1,7,9.,6,7.,.6
18500	065471	001	001	007				
18501	065474	011	006	007				
18502	065477	011	006					
18503		065502					.EVEN	
18504								
18505	065502	177777				ADD3A:	.WORD	-1
18506	065504	007	011	003			.BYTE	7,9.,3,6,5,2,2,2,7,8.,1
18507	065507	006	005	002				
18508	065512	002	002	007				
18509	065515	010	001					
18510		065520					.EVEN	
18511								
18512	065520	177777				ADD3B:	.WORD	-1
18513	065522	000	003	007			.BYTE	0,3,7,0,5,3,4,2,5,9.,8.,5
18514	065525	000	005	003				
18515	065530	004	002	005				
18516	065533	011	010	005				
18517							.EVEN	
18518								
18519	065536	177777				ADD3C:	.WORD	-1
18520	065540	001	002	001			.BYTE	1,2,1,6,7,7,0,1,9.,5,7,7
18521	065543	006	007	007				
18522	065546	000	001	011				
18523	065551	005	007	007				
18524							.EVEN	
18525								
18526	065554	177777				ADD4:	.WORD	-1
18527	065556	004	007	002			.BYTE	4,7,2,9.,3,2,7,7,1,6,1,7,9.,2,8.,4
18528	065561	011	003	002				
18529	065564	007	007	001				
18530	065567	006	001	007				
18531	065572	011	002	010				
18532	065575	004						
18533							.EVEN	
18534								
18535	065576	177777				ADD4A:	.WORD	-1
18536	065600	011	003	007			.BYTE	9.,3,7
18537		065604					.EVEN	
18538								
18539	065604	177777				ADD4B:	.WORD	-1
18540	065606	004	007	002			.BYTE	4,7,2,9.,3,2,7,7,1,6
18541	065611	011	003	002				
18542	065614	007	007	001				
18543	065617	006						
18544	065620	001	010	000			.BYTE	1,8.,0,2,2,1

18545	065623	002	002	001			
18546						.EVEN	
18547							
18548	065626	000000			ADD4C:	.WORD	0
18549	065630	004	007	002		.BYTE	4,7,2,9.,3,2,7,7,1,6
18550	065633	011	003	002			
18551	065636	007	007	001			
18552	065641	006					
18553	065642	001	007	010		.BYTE	1,7,8.,3,4,7
18554	065645	003	004	007			
18555						.EVEN	
18556							
18557	065650	000000			ADD5:	.WORD	0
18558	065652	004	000	001		.BYTE	4,0,1,7,7,7,9.,6,3,2,5,8.,8.,1
18559	065655	007	007	007			
18560	065660	011	006	003			
18561	065663	002	005	010			
18562	065666	010	001				
18563						.EVEN	
18564							
18565	065670	000000			ADD5A:	.WORD	0
18566	065672	000				.BYTE	0
18567		065674				.EVEN	
18568							
18569	065674	000000			ADD5B:	.ORD	0
18570	065676	007	011	006		.BYTE	7,9.,6,3,2,5,8.,8.,1
18571	065701	003	002	005			
18572	065704	010	010	001			
18573		065710				.EVEN	
18574							
18575	065710	177777			ADD5C:	.WORD	-1
18576	065712	007	011	006		.BYTE	7,9.,6,3,2,5,8.,8.,1
18577	065715	003	002	005			
18578	065720	010	010	001			
18579		065724				.EVEN	
18580							
18581	065724	000000			ADD6:	.WORD	0
18582	065726	001	007	011		.BYTE	1,7,9.,3,3,9.,8.,1
18583	065731	003	003	011			
18584	065734	010	001				
18585						.EVEN	
18586							
18587	065736	000000			ADD6A:	.WORD	0
18588	065740	010	010	003		.BYTE	8.,8.,3,9.,2,7,6,4,9.
18589	065743	011	002	007			
18590	065746	006	004	011			
18591		065752				.EVEN	
18592							
18593	065752	000000			ADD6B:	.WORD	0
18594	065754	000				.BYTE	0
18595		065756				.EVEN	
18596							
18597							
18598							
18599							
18600							

: +
: :
: -

INLINE TEST CASE (ADDNI & SUBNI)

```

18601 065756 010020 TADDI: .WORD 010020 ;SRC1 DESCRIPTOR
18602 065760 064274 .WORD #DSBUF1
18603 065762 010021 .WORD 010021 ;SRC2 DESCRIPTOR
18604 065764 064336 .WORD #DSBUF2
18605 065766 010024 .WORD 010024 ;DEST DESCRIPTOR
18606 065770 064400 .WORD #DS.DST
18607
18608
18609 :+
18610 :- INLINE TEST CASE (ADDPI & SUBPI)
18611
18612 065772 070020 TADDPI: .WORD 070020
18613 065774 064274 .WORD #DSBUF1
18614 065776 070021 .WORD 070021
18615 066000 064336 .WORD #DSBUF2
18616 066002 070024 .WORD 070024
18617 066004 064400 .WORD #DS.DST
18618
18619
18620
18621
18622 066006 INLINE <SPACE: .BLKW 2000> SPACE: .BLKW 2000
18623 066006 C02000
18624 072006 INLINE <SPACEY: .BLKW 2000> SPACEY: .BLKW 2000
18625 072006 002000
18626
18627
18628
18629 000001 .END

```


BIT07 = 000200	837#	847	5358	5368	7384	7393	7394	12867	12925	13000	13196	13293	13354
	13417	13480	15239	15301	15415	15422	16861	17270					
BIT08 = 000400	836#	846											
BIT09 = 001000	835#	845											
BIT1 = 000002	853#	5681											
BIT10 = 002000	834#												
BIT11 = 004000	833#	17487											
BIT12 = 010000	832#												
BIT13 = 020000	831#	17570											
BIT14 = 040000	830#	17485											
BIT15 = 100000	829#												
BIT2 = 000004	852#	10213											
BIT3 = 000010	851#	13803											
BIT4 = 000020	850#	15775											
BIT5 = 000040	849#	16208											
BIT6 = 000100	848#	5374											
BIT7 = 000200	847#												
BIT8 = 000400	846#	5570											
BIT9 = 001000	845#												
BPTVEC= 000014	861#												
BUFF1 063072	2069	2196	2204	2212	2220	2228	2236	2244	2250	2258	2266	2274	2282
	2290	2298	2306	2315	2664	2790	2798	2804	2812	2814	2820	3188	3317
	3325	3331	3339	3347	3390	3420	3429	3448	3452	3540	3546	3576	3633
	3676	3706	3715	3734	3738	3826	3832	3863	3916	3958	3986	3995	4014
	4018	4122	4126	4174	4228	4271	4299	4308	4327	4331	4432	4436	4481
	4535	4578	4606	4617	4618	4767	4777	4885	4926	4954	4971	4977	5114
	5122	5128	5156	5220	18138#								
BUFF2 063172	2073	2151	2158	2165	2167	2194	2202	2210	2218	2226	2234	2242	2252
	2260	2268	2276	2284	2292	2300	2308	2317	2668	2746	2753	2762	2773
	2788	2796	2806	2822	3192	3272	3279	3286	3300	3315	3323	3333	3341
	3349	4582	4610	4625	4626	4771	4887	4930	4958	5118	5222	18139#	
BUFLen= 000100	743#	2013	2031	2032	2167	2168	2227	2257	2620	2635	2636	2773	3143
	3146	3147	3300	3567	3632	3853	3915	4165	4220	4223	4224	4227	4472
	4527	4530	4531	4534	5515								
BYTECO= 000403	2970#	2973	5335#	5338	7426#	7429	8830#	8833	10252#	10255	11173#	11176	12134#
	12137	12161#	12164	12896#	12899	13020#	13023	13039#	13042	13604#	13607	13842#	13845
	14223#	14226	14645#	14648	14667#	14670	15815#	15818	16248#	16251			
CBIT = 000001	798#	1778	2394	2889	6973	7453							
CC 000520	1084#	1332*	1334*	1346	1356	1552*	1554*	1566	1576	1876*	1878*	1880	2482*
	2484*	2486	3003*	3005*	3007	3469*	3471*	3473	3755*	3757*	3759	4051*	4053*
	4055	4361*	4363*	4365	4696*	4698*	4700	5043*	5045*	5047	5139*	5141	5404*
	5406*	5408	5955*	5957*	5959	6063*	6065*	6067	6539*	6541*	6543	6647*	6649*
	6651	7009*	7011*	7013	7089*	7091*	7093	7485*	7487*	7489	7565*	7567*	7569
	8504*	8506*	8508	8613*	8615*	8617	9233*	9235*	9237	9337*	9339*	9341	10906*
	10908*	10910	11082*	11084*	11086	11833*	11835*	11837	12010*	12012*	12014	12635*	12637*
	12639	12745*	12747*	12749	13524*	13526*	13528	13678*	13680*	13682	13955*	13957*	13959
	14123*	14125*	14127	14341*	14343*	14345	14511*	14513*	14515	15004*	15006*	15008	15076*
	15078*	15080	15523*	15525*	15527	15651*	15653*	15655	15928*	15930*	15932	16095*	16097*
	16099	16366*	16368*	16370	16537*	16539*	16541						
CC.EXP 000522	1085#	1774*	1776*	1778	1780	1880	2390*	2392*	2394	2396	2486	2885*	2887*
	2889	2891	3007	3444*	3446*	3473	3730*	3732*	3759	4010*	4012*	4055	4323*
	4325*	4365	4620*	4628*	4680*	4700	5027*	5029*	5047	5408	5743*	5751*	5758*
	5779*	5804*	5891*	5959	6272*	6283*	6288*	6305*	6324*	6352*	6392*	6411*	6445*
	6472*	6543	6857*	6869*	6954*	6973*	7013	7280*	7294*	7343*	7387*	7417*	7445*
	7453*	7489	8034*	8210*	8227*	8240*	8330	8361*	8375*	8387*	8421	8442	8466
	8508	8802*	9064*	9082*	9095*	9237	10289*	10910	11210*	11214*	11837	12101*	12187*

CTL5	062503	17635	18069#	
CTL6	062520	17636	18072#	
CTL7	062535	17637	18075#	
CTL8	062552	17638	18078#	
CTL9	062567	17639	18081#	
CVTIL	032020	6937	7440	9935#
CVTLI	031114	5741	6270	9533#
CVTLN =	076057	938#	6531	
CVTLNI =	076157	947#	6641	
CVTLP =	076077	958#	5947	
CVTLPI =	076177	967#	6057	
CVTNAA	025021	7773	7918#	
CVTNBB	025024	7776	7921#	
CVTNCC	025027	7779	7924#	
CVTNDD	025032	7782	7927#	
CVTNEE	025035	7785	7930#	
CVTNFF	025040	7788	7933#	
CVTNGG	025043	7791	7936#	
CVTNHH	025046	7794	7939#	
CVTNII	025051	7797	7942#	
CVTNJJ	025054	7800	7945#	
CVTNL =	076053	934#	7477	
CVTNLA	024673	7695	7837#	
CVTNLB	024706	7698	7843#	
CVTNLC	024711	7701	7846#	
CVTNLD	024714	7704	7849#	
CVTNLE	024717	7707	7852#	
CVTNLF	024722	7710	7855#	
CVTNLG	024725	7713	7858#	
CVTNLH	024730	7716	7861#	
CVTNLI =	076153	943#	7559	
CVTNLJ	024736	7722	7867#	
CVTNLK	024741	7725	7870#	
CVTNLL	024744	7728	7873#	
CVTNLM	024747	7731	7876#	
CVTNLN	024752	7734	7879#	
CVTNLO	024755	7737	7882#	
CVTNLP	024760	7740	7885#	
CVTNLQ	024763	7743	7888#	
CVTNLR	024766	7746	7891#	
CVTNLS	024771	7749	7894#	
CVTNLT	024774	7752	7897#	
CVTNLU	024777	7755	7900#	
CVTNLV	025002	7758	7903#	
CVTNLW	025005	7761	7906#	
CVTNLX	025010	7764	7909#	
CVTNLY	025013	7767	7912#	
CVTNLZ	025016	7770	7915#	
CVTNL\$	024733	7719	7864#	
CVTNL1	024617	7668	7803#	
CVTNL2	024621	7671	7806#	
CVTNL3	024623	7674	7809#	
CVTNL4	024631	7677	7813#	
CVTNL5	024636	7680	7817#	
CVTNL6	024646	7683	7822#	
CVTNL7	024651	7686	7825#	

CVTNL8	024657	7689	7829#											
CVTNL9	024665	7692	7833#											
CVTNP =	076055	936#	9226											
CVTNP1=	076155	945#	9331											
CVTNP1	030750	9404	9440#											
CVTNP2	030754	9408	9444#											
CVTNP3	030760	9412	9448#											
CVTNP4	030764	9416	9452#											
CVTNP5	031012	9420	9462#											
CVTNP6	031024	9424	9468#											
CVTNP7	031040	9428	9474#											
CVTNP8	031062	9432	9482#											
CVTNP9	031072	9436	9487#											
CVTPL =	076073	954#	7001											
CVTPL1=	076173	963#	7083											
CVTPL1	023002	7167	7185#											
CVTPL2	023003	7170	7187#											
CVTPL3	023006	7173	7189#											
CVTPL4	023011	7176	7191#											
CVTPL5	023017	7179	7194#											
CVTPL6	023023	7182	7197#											
CVTPN =	076054	935#	8496											
CVTPN1=	076154	944#	8607											
CVTPN1	027104	8687	8725#											
CVTPN2	027105	8691	8727#											
CVTPN3	027106	8695	8729#											
CVTPN4	027111	8700	8731#											
CVTPN5	027124	8704	8736#											
CVTPN6	027130	8708	8739#											
CVTPN7	027134	8712	8742#											
CVTPN8	027137	8716	8744#											
CVTPN9	027145	8720	8747#											
DATA00	000562	1104#	1729*	1750	1940	1966	1986	1987	1989	1990	2346*	2366	2544	2549
		2550	2585	2597	2850*	2871	3071	3097	3118	3119	3121	3122	3376*	3388
		3415	3420	3421	3426	3661*	3674	3701	3706	3707	3712	3945*	3956	3981
		3986	3987	3992	4258*	4269	4294	4299	4300	4305	4565*	4576	4614	4655
		4675	4915*	4928	4998	5004	5285*	5296	5472	5492	5493	5495	5496	5724*
		5745	5870	5906	6253*	6274	6277	6490	6817*	6863	6939	6988	7242*	7288
		7320	7464	7989*	8043	8063	8485	8788*	8827	8849	9010	9202	10277*	10296
		10303	10387	10388	10432	10433	10457	10477	10495	10514	10536	11198*	11221	11228
		11312	11313	11357	11358	11382	11402	11420	11439	11461	12114*	12134	12147	12182
		12192	12215	12235	12245	12256	12264	12342	12343	12351	12352	12375	12378	12379
		12404	12432	12433	12466	12467	12614	12615	12830*	12843	12881	12911	12928	13003
		13052	13118	13123	13124	13149	13152	13153	13179	13194	13214	13217	13218	13239
		13251	13252	13276	13291	13315	13337	13352	13379	13380	13415	13478	13503	13867*
		13891	13923	14061*	14083	14248*	14276	14309	14448*	14474	14624*	14645	14688	14698
		14721	14741	14751	14762	14770	14840	14983	14984	15161*	15176	15253	15271	15287
		15297	15307	15411	15425	15459	15506	15840*	15864	15896	16035*	16057	16273*	16301
		16334	16473*	16499										
DATA01	000564	1105#	1731*	1752	1934	2348*	2368	2420	2549	2593	2852*	2873	2915	3063
		3378*	3392	3455	3663*	3741	3947*	3981	3992	3993	3995	3996	4018	4019
		4260*	4294	4305	4306	4308	4309	4331	4332	4567*	4580	4614	4665	4677
		4917*	4962	4965	5001	5002	5015	5016	5018	5019	5287*	5298	5321	5464
		5726*	5737	5910	6255*	6266	6494	6819*	6861	6990	7244*	7284	7401	7466
		7991*	8058	8123	8487	8790*	8810	8821	9024	9131	10279*	10299	11200*	11224
		12116*	12127	12204	12220	12223	12233	12354	12381	12409	12435	12454	12832*	12891

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 379
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0378

ILLTST	016326	5557	5594#							
INC1	033136	10416#	10997							
INC2	033530	10563#	10992							
INC21	035744	11341#	11925							
INC22	036336	11488#	11920							
INC23	036734	11636#	11915							
INC3	034122	10709#	10987							
INTFLG	000460	1068#								
INTPAS	000462	1069#								
INTST	015222	5198	5234#							
IOTVEC=	000020	862#	1185*	1186*						
KIPAR0=	172340	761#	17287*							
KIPAR1=	172342	762#	17289*							
KIPAR2=	172344	763#	17291*							
KIPAR3=	172346	764#	17293*							
KIPAR4=	172350	765#	17295*							
KIPAR5=	172352	766#	17297*							
KIPAR6=	172354	767#	17299*							
KIPAR7=	172356	768#	17301*							
KIPDR0=	172300	752#	17304*							
KIPDR1=	172302	753#	17306*							
KIPDR2=	172304	754#	17308*							
KIPDR3=	172306	755#	17310*							
KIPDR4=	172310	756#	17312*							
KIPDR5=	172312	757#	17314*							
KIPDR6=	172314	758#	17316*							
KIPDR7=	172316	759#	17318*							
LENGTH=	000004	17389#	17394							
LF =	000012	736#	17705							
LKS =	177546	742#								
LOCC =	076040	912#	3462							
LOCCI =	076140	922#	3551							
LOOP2	015624	5378#	5382							
LTCVEC=	000100	868#	1193*							
L2D0 =	076020	889#	1319							
L2D1 =	076021	890#								
L2D2 =	076022	891#								
L2D3 =	076023	892#								
L2D4 =	076024	893#	1328							
L2D5 =	076025	894#								
L2D6 =	076026	895#								
L2D7 =	076027	896#	1436							
L3D0 =	076060	898#								
L3D1 =	076061	899#								
L3D2 =	076062	900#								
L3D3 =	076063	901#	1539							
L3D4 =	076064	902#								
L3D5 =	076065	903#	1548							
L3D6 =	076066	904#								
L3D7 =	076067	905#	1637							
MATC =	076045	917#	5036							
MATCI =	076145	927#	5133							
MFPT =	000007	886#								
MMCYL	000446	1063#	1196*	16859	17254*	17265*	17273*	17327	17622	
MMDIS	062772	17624	18116#							
MMENA	063032	17626	18122#							

SYMS - 000004

15440#	15445#	15454#	15455#	15461#	15473#	15475#	15480#	15486#	15488#	15506#	15511#	15525#
15554#	15594#	15612#	15614#	15653#	15769#	15818#	15896#	15901#	15906#	15930#	16009#	16011#
16097#	16144#	16149#	16151#	16202#	16251#	16334#	16339#	16344#	16368#	16447#	16449#	16539#
16586#	16591#	16593#	17020#	17057#	17094#	17129#	17167#	17330#	17338#	17394#	17435#	17447#
17451#	17460#											
1264#	1334#	1422#	1554#	1627#	1776#	1789#	1878#	1958#	1981#	1986#	1989#	2001#
2028#	2031#	2051#	2053#	2167#	2392#	2405#	2484#	2549#	2551#	2566#	2583#	2585#
2595#	2597#	2599#	2623#	2635#	2646#	2648#	2887#	2900#	2973#	3005#	3065#	3089#
3113#	3118#	3121#	3133#	3146#	3148#	3150#	3170#	3172#	3286#	3420#	3426#	3429#
3446#	3452#	3471#	3522#	3524#	3546#	3706#	3712#	3715#	3732#	3738#	3757#	3808#
3810#	3832#	3853#	3986#	3992#	3995#	4012#	4018#	4034#	4053#	4104#	4106#	4126#
4142#	4299#	4305#	4308#	4325#	4331#	4363#	4414#	4416#	4436#	4617#	4625#	4639#
4642#	4645#	4648#	4660#	4670#	4698#	4749#	4751#	4777#	4792#	4965#	4971#	5001#
5009#	5015#	5018#	5029#	5045#	5096#	5098#	5338#	5406#	5466#	5487#	5492#	5495#
5507#	5518#	5521#	5578#	5675#	5745#	5751#	5758#	5774#	5779#	5787#	5789#	5798#
5800#	5804#	5821#	5823#	5825#	5827#	5840#	5854#	5859#	5873#	5882#	5888#	5891#
5957#	6036#	6038#	6065#	6274#	6277#	6283#	6288#	6305#	6324#	6347#	6352#	6361#
6363#	6370#	6371#	6389#	6390#	6392#	6398#	6411#	6413#	6415#	6428#	6437#	6443#
6445#	6462#	6472#	6541#	6620#	6622#	6649#	6863#	6869#	6878#	6880#	6888#	6892#
6910#	6912#	6914#	6917#	6919#	6934#	6939#	6945#	6954#	6963#	6973#	7011#	7063#
7065#	7091#	7288#	7294#	7303#	7305#	7313#	7320#	7333#	7335#	7343#	7358#	7363#
7365#	7377#	7382#	7387#	7393#	7401#	7417#	7429#	7434#	7445#	7453#	7487#	7539#
7541#	7567#	8043#	8058#	8060#	8115#	8117#	8123#	8136#	8138#	8140#	8155#	8157#
8159#	8161#	8163#	8168#	8174#	8176#	8178#	8210#	8227#	8240#	8253#	8255#	8291#
8310#	8327#	8333#	8338#	8354#	8357#	8361#	8375#	8387#	8391#	8394#	8400#	8418#
8424#	8439#	8461#	8489#	8506#	8587#	8589#	8615#	8823#	8827#	8833#	8849#	8861#
8866#	8871#	8888#	8890#	8905#	8912#	8936#	8993#	8998#	9001#	9010#	9013#	9064#
9082#	9095#	9129#	9131#	9133#	9149#	9151#	9168#	9170#	9172#	9175#	9180#	9186#
9188#	9235#	9310#	9312#	9339#	9559#	9623#	9640#	9644#	9646#	9656#	9660#	9669#
9675#	9679#	9681#	9691#	9695#	9704#	9710#	9714#	9716#	9726#	9730#	9739#	9745#
9749#	9751#	9761#	9765#	9774#	9780#	9784#	9786#	9796#	9800#	9809#	9815#	9819#
9828#	9830#	9840#	9846#	9848#	9858#	9864#	9866#	9876#	9882#	9884#	9889#	9904#
9909#	9911#	9960#	9964#	9966#	9976#	9982#	9986#	9988#	9998#	10004#	10008#	10010#
10020#	10026#	10030#	10032#	10042#	10048#	10052#	10054#	10064#	10070#	10074#	10084#	10090#
10094#	10104#	10110#	10114#	10124#	10130#	10134#	10144#	10148#	10162#	10207#	10255#	10387#
10394#	10401#	10432#	10462#	10467#	10477#	10485#	10495#	10503#	10514#	10517#	10536#	10539#
10579#	10609#	10614#	10624#	10632#	10642#	10650#	10660#	10663#	10682#	10685#	10725#	10758#
10763#	10775#	10783#	10788#	10800#	10808#	10813#	10825#	10828#	10847#	10850#	10908#	11001#
11003#	11027#	11043#	11059#	11084#	11176#	11312#	11319#	11326#	11357#	11387#	11392#	11402#
11410#	11420#	11428#	11439#	11442#	11461#	11464#	11506#	11536#	11541#	11551#	11559#	11569#
11577#	11587#	11590#	11609#	11612#	11652#	11685#	11690#	11702#	11710#	11715#	11727#	11735#
11740#	11752#	11755#	11774#	11777#	11835#	11929#	11931#	11955#	11971#	11987#	12012#	12127#
12137#	12154#	12164#	12187#	12197#	12207#	12226#	12240#	12250#	12276#	12289#	12299#	12308#
12335#	12342#	12351#	12357#	12362#	12378#	12393#	12421#	12432#	12466#	12481#	12488#	12497#
12503#	12508#	12521#	12536#	12559#	12568#	12602#	12614#	12619#	12637#	12688#	12698#	12709#
12711#	12747#	12853#	12874#	12884#	12899#	12917#	12928#	12931#	12949#	12958#	12975#	12979#
12981#	12991#	13003#	13009#	13016#	13023#	13036#	13042#	13062#	13076#	13085#	13087#	13105#
13123#	13136#	13152#	13174#	13185#	13190#	13201#	13212#	13217#	13227#	13230#	13247#	13251#
13271#	13302#	13304#	13363#	13365#	13379#	13397#	13429#	13438#	13443#	13492#	13503#	13508#
13526#	13547#	13586#	13607#	13623#	13625#	13680#	13797#	13845#	13923#	13928#	13933#	13957#
14035#	14037#	14125#	14172#	14177#	14179#	14226#	14309#	14314#	14319#	14343#	14422#	14424#
14513#	14560#	14565#	14567#	14638#	14648#	14660#	14670#	14693#	14703#	14713#	14732#	14746#
14756#	14782#	14795#	14805#	14814#	14837#	14865#	14877#	14881#	14893#	14898#	14909#	14937#
14949#	14953#	14965#	14970#	14983#	14988#	15006#	15039#	15041#	15078#	15201#	15211#	15246#
15256#	15267#	15275#	15280#	15293#	15307#	15314#	15326#	15331#	15333#	15339#	15341#	15350#
15352#	15358#	15360#	15373#	15380#	15384#	15389#	15393#	15395#	15405#	15425#	15431#	15438#

5861	5871	5876	5880	5885	5915	5921	5960	5967	5974	5981	5988	5995
6002	6022	6024	6027	6033	6042	6068	6075	6082	6089	6096	6103	6110
6119	6128	6135	6138	6149	6158	6281	6286	6293	6296	6300	6312	6315
6319	6333	6335	6343	6349	6357	6359	6368	6377	6381	6387	6404	6407
6409	6422	6426	6430	6441	6456	6460	6468	6474	6499	6505	6544	6551
6558	6565	6572	6579	6586	6606	6608	6611	6617	6626	6652	6659	6666
6673	6680	6687	6694	6703	6714	6723	6831	6837	6867	6875	6886	6898
6921	6943	6951	6961	6970	7014	7021	7028	7035	7042	7049	7056	7069
7094	7101	7108	7115	7122	7129	7136	7143	7150	7256	7262	7292	7300
7316	7325	7330	7339	7352	7356	7360	7380	7385	7399	7409	7413	7427
7436	7438	7443	7450	7490	7497	7504	7511	7518	7525	7532	7545	7570
7581	7593	7601	7608	8007	8013	8066	8071	8075	8079	8096	8100	8153
8166	8182	8192	8196	8206	8208	8224	8238	8281	8285	8296	8308	8320
8325	8331	8335	8349	8352	8368	8371	8373	8380	8383	8385	8398	8413
8416	8422	8437	8443	8447	8459	8467	8471	8509	8516	8523	8530	8537
8544	8551	8567	8569	8572	8578	8593	8618	8629	8641	8649	8656	8663
8670	8831	8840	8842	8856	8859	8864	8868	8883	8886	8900	8903	8910
8926	8929	8934	8950	8953	8957	8970	8973	8977	9017	9021	9027	9032
9042	9055	9057	9060	9062	9079	9093	9107	9111	9143	9165	9178	9192
9238	9245	9252	9259	9266	9273	9280	9296	9298	9301	9307	9316	9342
9353	9365	9373	9380	9387	9553	9563	9565	9568	9573	9614	9621	9638
9648	9652	9654	9673	9683	9687	9689	9708	9718	9722	9724	9743	9753
9757	9759	9778	9788	9792	9794	9813	9821	9826	9832	9844	9850	9862
9868	9880	9886	9902	9906	9958	9968	9980	9990	10002	10012	10024	10034
10046	10056	10068	10076	10088	10096	10108	10116	10128	10136	10153	10160	10211
10214	10253	10262	10264	10297	10313	10326	10342	10355	10373	10419	10443	10460
10464	10474	10483	10492	10501	10511	10521	10525	10533	10545	10549	10566	10590
10607	10611	10621	10630	10639	10648	10657	10667	10671	10679	10691	10695	10712
10736	10753	10756	10760	10772	10779	10786	10797	10804	10811	10822	10832	10836
10844	10856	10860	10911	10918	10925	10932	10939	10946	10953	10972	10974	10977
10983	11007	11087	11103	11115	11135	11137	11140	11146	11174	11183	11185	11222
11238	11251	11267	11280	11298	11344	11368	11385	11389	11399	11408	11417	11426
11436	11446	11450	11458	11470	11474	11491	11517	11534	11538	11548	11557	11566
11575	11584	11594	11598	11606	11618	11622	11639	11663	11680	11683	11687	11699
11706	11713	11724	11731	11738	11749	11759	11763	11771	11783	11787	11838	11845
11852	11859	11866	11873	11880	11900	11902	11905	11911	11935	12015	12031	12043
12063	12065	12068	12074	12135	12144	12146	12148	12162	12171	12173	12183	12185
12189	12193	12195	12199	12203	12205	12216	12218	12221	12224	12230	12236	12238
12242	12246	12248	12257	12272	12274	12278	12282	12285	12287	12291	12295	12297
12305	12333	12349	12355	12359	12373	12376	12382	12386	12402	12405	12410	12414
12430	12436	12440	12450	12455	12459	12479	12495	12501	12505	12519	12525	12529
12543	12548	12552	12566	12572	12576	12586	12591	12595	12640	12647	12654	12661
12668	12675	12682	12692	12702	12715	12750	12761	12773	12849	12855	12868	12880
12882	12892	12897	12906	12908	12914	12919	12926	12939	12942	12946	12956	12964
12966	12971	12973	12983	12987	12989	13001	13007	13014	13021	13030	13032	13033
13040	13049	13051	13056	13058	13064	13074	13110	13119	13132	13145	13150	13156
13160	13172	13182	13187	13195	13197	13199	13210	13215	13225	13237	13240	13243
13245	13255	13259	13269	13279	13284	13292	13294	13296	13313	13316	13319	13323
13333	13340	13345	13353	13355	13357	13374	13383	13387	13403	13408	13416	13418
13420	13427	13436	13452	13456	13466	13471	13479	13481	13483	13490	13529	13540
13551	13558	13575	13577	13580	13588	13592	13605	13614	13616	13621	13631	13683
13694	13706	13717	13724	13734	13736	13739	13747	13801	13804	13843	13852	13854
13960	13967	13974	13981	13988	13995	14002	14021	14023	14026	14032	14041	14128
14144	14156	14185	14187	14190	14196	14224	14233	14235	14346	14353	14360	14367
14374	14381	14388	14408	14410	14413	14419	14428	14516	14532	14544	14573	14575
14578	14584	14646	14655	14657	14668	14677	14679	14689	14691	14695	14699	14701

2552	2562#	2566#	2567	2579#	2583#	2584#	2585#	2586	2593#	2594	2595#	2596#
2597#	2598	2599#	2600#	2601	2611#	2623#	2624#	2625	2631#	2635#	2636#	2637
2646#	2647	2648#	2649	2664#	2665	2666#	2668#	2669	2670#	2672#	2673	2674#
2675	2676#	2677	2678#	2679	2680#	2681	2682#	2683	2684#	2685	2698#	2707#
2714#	2721#	2728#	2735#	2742#	2749#	2756#	2760#	2761	2762#	2763	2769#	2834#
2835	2836#	2837	2838#	2839	2848#	2849	2850#	2851	2852#	2853	2854#	2855
2856#	2857	2858#	2859	2860#	2861	2862#	2863	2871#	2872	2873#	2874	2875#
2876	2877#	2878	2879#	2880	2881#	2882	2885#	2886	2887#	2888	2895#	2896
2900#	2901#	2902	2905#	2906	2907#	2908	2909#	2910	2911#	2912	2913#	2914
2915#	2916	2920#	2928#	2933#	2939#	2940	2944#	2952#	2957#	2963#	2964	2970#
2971	2973#	2974	2977	2978#	2979	2983#	2984	2985#	2986	2987#	2988	2989#
2990	3003#	3004	3005#	3006	3010#	3017#	3024#	3031#	3038#	3045#	3052#	3061#
3062	3063#	3064	3065#	3066	3067#	3068	3069#	3070	3074#	3075	3085#	3089#
3090	3097#	3098	3108#	3113#	3114	3118#	3119#	3120	3121#	3122#	3123	3129#
3133#	3134	3146#	3147#	3148#	3149	3150#	3151#	3152	3158#	3170#	3171	3172#
3173	3188#	3189	3190#	3192#	3193	3194#	3196#	3197	3198#	3199	3200#	3201
3202#	3203	3204#	3205	3206#	3207	3208#	3209	3224#	3233#	3240#	3247#	3254#
3261#	3268#	3275#	3282#	3286#	3287#	3288	3289#	3290	3296#	3361#	3362	3363#
3364	3365#	3366	3374#	3375	3376#	3377	3378#	3379	3380#	3381	3388#	3389
3390#	3391	3392#	3393	3398#	3399	3400#	3401	3402#	3403	3407#	3408	3409#
3410	3411#	3412	3418#	3419	3420#	3421#	3422	3426#	3427#	3428	3429#	3430#
3431	3434#	3435	3436#	3437	3438#	3439	3440#	3441	3444#	3445	3446#	3447
3448#	3449	3450#	3452#	3453#	3454	3455#	3456	3469#	3470	3471#	3472	3476#
3483#	3490#	3497#	3504#	3511#	3518#	3522#	3523	3524#	3525	3540#	3541	3542#
3544#	3546#	3547#	3548	3549#	3550	3557#	3558	3563#	3567#	3568	3572#	3579#
3586#	3593#	3600#	3607#	3645#	3646	3647#	3648	3649#	3650	3659#	3660	3661#
3662	3663#	3664	3665#	3666	3674#	3675	3676#	3677	3678#	3679	3684#	3685
3686#	3687	3688#	3689	3693#	3694	3695#	3696	3697#	3698	3704#	3705	3706#
3707#	3708	3712#	3713#	3714	3715#	3716#	3717	3720#	3721	3722#	3723	3724#
3725	3726#	3727	3730#	3731	3732#	3733	3734#	3735	3736#	3738#	3739#	3740
3741#	3742	3755#	3756	3757#	3758	3762#	3769#	3776#	3783#	3790#	3797#	3804#
3808#	3809	3810#	3811	3826#	3827	3828#	3830#	3832#	3833#	3834	3835#	3836
3843#	3844	3849#	3853#	3854#	3855	3859#	3866#	3873#	3880#	3887#	3894#	3929#
3930	3931#	3932	3933#	3934	3943#	3944	3945#	3946	3947#	3948	3956#	3957
3958#	3959	3960#	3961	3962#	3963	3968#	3969	3970#	3971	3975#	3976	3977#
3978	3984#	3985	3986#	3987#	3988	3992#	3993#	3994	3995#	3996#	3997	4000#
4001	4002#	4003	4004#	4005	4006#	4007	4010#	4011	4012#	4013	4014#	4015
4016#	4018#	4019#	4020	4021#	4022	4032#	4033	4034#	4035#	4036	4037#	4038
4051#	4052	4053#	4054	4058#	4065#	4072#	4079#	4086#	4093#	4100#	4104#	4105
4106#	4107	4122#	4123	4124#	4126#	4127#	4128	4129#	4130	4131#	4132	4142#
4143#	4144	4145#	4146	4147#	4155#	4156	4161#	4165#	4166	4170#	4177#	4184#
4191#	4198#	4205#	4242#	4243	4244#	4245	4246#	4247	4256#	4257	4258#	4259
4260#	4261	4269#	4270	4271#	4272	4273#	4274	4275#	4276	4281#	4282	4283#
4284	4288#	4289	4290#	4291	4297#	4298	4299#	4300#	4301	4305#	4306#	4307
4308#	4309#	4310	4313#	4314	4315#	4316	4317#	4318	4319#	4320	4323#	4324
4325#	4326	4327#	4328	4329#	4331#	4332#	4333	4334#	4335	4345#	4346	4347#
4348	4361#	4362	4363#	4364	4368#	4375#	4382#	4389#	4396#	4403#	4410#	4414#
4415	4416#	4417	4432#	4433	4434#	4436#	4437#	4438	4439#	4440	4441#	4442
4447#	4448	4452#	4454#	4455	4462#	4463	4468#	4472#	4473	4477#	4484#	4491#
4498#	4505#	4512#	4549#	4550	4551#	4552	4553#	4554	4563#	4564	4565#	4566
4567#	4568	4569#	4570	4576#	4577	4578#	4579	4580#	4581	4582#	4583	4584#
4585	4586#	4587	4592#	4593	4597#	4598	4606#	4607	4608#	4610#	4611	4612#
4617#	4618#	4619	4620#	4621	4625#	4626#	4627	4628#	4629	4632#	4633	4639#
4640#	4641	4642#	4643#	4644	4645#	4646#	4647	4648#	4649#	4650	4651#	4652
4653#	4654	4658#	4659	4660#	4661#	4662	4668#	4669	4670#	4671#	4672	4680#
4681	4696#	4697	4698#	4699	4703#	4710#	4717#	4724#	4731#	4738#	4745#	4749#

4750	4751#	4752	4767#	4768	4769#	4771#	4772	4773#	4775#	4777#	4778#	4779
4780#	4781	4790#	4791	4792#	4793	4797#	4804#	4811#	4818#	4825#	4832#	4839#
4899#	4900	4901#	4902	4903#	4904	4913#	4914	4915#	4916	4917#	4918	4924#
4925	4926#	4927	4928#	4929	4930#	4931	4936#	4937	4938#	4939	4943#	4944
4945#	4946	4954#	4955	4956#	4958#	4959	4960#	4965#	4966	4971#	4972#	4973
4977#	4978	4986#	4987	4988#	4989	4990#	4991	4992#	4993	4994#	4995	4996#
4997	5001#	5002#	5003	5007#	5008	5009#	5010#	5011	5015#	5016#	5017	5018#
5019#	5020	5027#	5028	5029#	5030	5043#	5044	5045#	5046	5050#	5057#	5064#
5071#	5078#	5085#	5092#	5096#	5097	5098#	5099	5114#	5115	5116#	5118#	5119
5120#	5122#	5123	5131#	5139#	5140	5145#	5152#	5159#	5166#	5173#	5180#	5187#
5194#	5251#	5252	5253#	5254	5255#	5256	5257#	5258	5259#	5260	5261#	5262
5268#	5269	5270#	5271	5272#	5273	5279#	5283#	5284	5285#	5286	5287#	5288
5289#	5290	5291#	5292	5293#	5294	5296#	5297	5298#	5299	5300#	5301	5302#
5303	5304#	5305	5306#	5307	5309#	5310	5311#	5312	5313#	5314	5315#	5316
5317#	5318	5319#	5320	5321#	5322	5324#	5326#	5327	5328#	5335#	5336	5338#
5339	5342	5343#	5344	5348#	5349	5350#	5351	5352#	5353	5354#	5355	5364#
5365	5372#	5373	5374#	5375	5376#	5377	5395#	5404#	5405	5406#	5407	5411#
5418#	5425#	5432#	5439#	5446#	5453#	5462#	5463	5464#	5465	5466#	5467	5468#
5469	5470#	5471	5472#	5473	5483#	5487#	5488	5492#	5493#	5494	5495#	5496#
5497	5503#	5507#	5508	5518#	5519#	5520	5521#	5522#	5523	5529#	5543#	5544
5545#	5546	5547#	5548	5549#	5550	5551#	5553#	5554	5566#	5567	5568#	5569
5573#	5574	5578#	5579	5598#	5599	5600#	5601	5602#	5603	5604#	5605	5606#
5607	5608#	5609	5612#	5614#	5615	5633#	5634	5635#	5636	5637#	5638	5675#
5676#	5677	5690#	5691	5692#	5693	5694#	5695	5696#	5697	5708#	5709	5710#
5711	5712#	5713	5722#	5723	5724#	5725	5726#	5727	5728#	5729	5735#	5737#
5738	5739#	5740	5741#	5743#	5744	5745#	5746#	5747	5751#	5752	5753#	5754
5758#	5759	5770#	5771	5774#	5775#	5776	5777#	5778	5779#	5780	5787#	5788#
5789#	5790	5791#	5792	5798#	5799#	5800#	5801	5802#	5803	5804#	5805	5806#
5807	5813#	5814	5817#	5818	5821#	5822#	5823#	5824	5825#	5826#	5827#	5828
5838#	5839	5840#	5841	5844#	5845	5850#	5851	5854#	5855#	5856	5857#	5858
5859#	5860	5866#	5867	5873#	5874#	5875	5882#	5883#	5884	5888#	5889#	5890
5891#	5892	5906#	5907	5908#	5909	5910#	5911	5912#	5913	5917#	5918	5919#
5920	5924#	5925	5926#	5927	5930#	5931	5932#	5933	5934#	5935	5936#	5937
5938#	5939	5940#	5941	5955#	5956	5957#	5958	5962#	5969#	5976#	5983#	5990#
5997#	6004#	6014#	6015	6016#	6017	6018#	6019	6029#	6036#	6037	6038#	6039
6053#	6055#	6063#	6064	6065#	6066	6070#	6077#	6084#	6091#	6098#	6105#	6112#
6116#	6117	6121#	6130#	6140#	6144#	6145	6151#	6237#	6238	6239#	6240	6241#
6242	6251#	6252	6253#	6254	6255#	6256	6257#	6258	6264#	6266#	6267	6268#
6269	6270#	6272#	6273	6274#	6275#	6276	6277#	6278#	6279	6283#	6284	6288#
6289	6298#	6299	6303#	6304	6305#	6306	6317#	6318	6322#	6323	6324#	6325
6337#	6338	6339#	6340	6345#	6346	6347#	6348	6352#	6353	6361#	6362#	6363#
6364	6365#	6366	6370#	6371#	6372	6373#	6374	6379#	6380	6389#	6390#	6391
6392#	6393	6398#	6399#	6400	6401#	6402	6411#	6412	6413#	6414	6415#	6416
6428#	6429	6433#	6434	6437#	6438#	6439	6443#	6444	6445#	6446	6449#	6450
6462#	6463#	6464	6470#	6471	6472#	6473	6477#	6478	6490#	6491	6492#	6493
6494#	6495	6496#	6497	6501#	6502	6503#	6504	6508#	6509	6510#	6511	6514#
6515	6516#	6517	6518#	6519	6520#	6521	6522#	6523	6524#	6525	6539#	6540
6541#	6542	6546#	6553#	6560#	6567#	6574#	6581#	6588#	6598#	6599	6600#	6601
6602#	6603	6613#	6620#	6621	6622#	6623	6637#	6639#	6647#	6648	6649#	6650
6654#	6661#	6668#	6675#	6682#	6689#	6696#	6700#	6701	6705#	6709#	6710	6716#
6801#	6802	6803#	6804	6805#	6806	6815#	6816	6817#	6818	6819#	6820	6826#
6827	6828#	6829	6833#	6834	6835#	6836	6840#	6841	6842#	6843	6846#	6847
6848#	6849	6855#	6857#	6858	6859#	6860	6861#	6862	6863#	6864#	6865	6869#
6870	6871#	6872	6873#	6874	6878#	6879#	6880#	6881	6883#	6888#	6889	6890#
6891	6892#	6893	6900#	6901	6910#	6911	6912#	6913	6914#	6915#	6916	6917#
6918	6919#	6920	6924#	6925	6934#	6935	6937#	6939#	6940#	6941	6945#	6946#

6947	6954#	6955	6956#	6957	6958#	6959	6963#	6964	6973#	6974	6988#	6989
6990#	6991	6992#	6993	6994#	6995	7009#	7010	7011#	7012	7016#	7023#	7030#
7037#	7044#	7051#	7058#	7063#	7064	7065#	7066	7079#	7081#	7089#	7090	7091#
7092	7096#	7103#	7110#	7117#	7124#	7131#	7138#	7145#	7152#	7226#	7227	7228#
7229	7230#	7231	7240#	7241	7242#	7243	7244#	7245	7251#	7252	7253#	7254
7258#	7259	7260#	7261	7265#	7266	7267#	7268	7271#	7272	7273#	7274	7280#
7281	7282#	7283	7284#	7285	7286#	7287	7288#	7289#	7290	7294#	7295	7296#
7297	7298#	7299	7303#	7304#	7305#	7306	7307#	7308	7311#	7312	7313#	7314
7320#	7321#	7322	7333#	7334	7335#	7336#	7337	7341#	7342	7343#	7344	7358#
7359	7363#	7364#	7365#	7366	7369#	7370	7371#	7372	7375#	7376	7377#	7378
7382#	7383	7387#	7388	7389#	7390	7393#	7394#	7395	7401#	7402#	7403	7415#
7416	7417#	7418	7426#	7427	7429#	7430	7433	7434#	7435	7440#	7445#	7446
7453#	7454	7464#	7465	7466#	7467	7468#	7469	7470#	7471	7485#	7486	7487#
7488	7492#	7499#	7506#	7513#	7520#	7527#	7534#	7539#	7540	7541#	7542	7555#
7557#	7565#	7566	7567#	7568	7572#	7584#	7596#	7603#	7610#	7973#	7974	7975#
7976	7977#	7978	7987#	7988	7989#	7990	7991#	7992	7993#	7994	8000#	8001
8002#	8003	8004#	8005	8009#	8010	8011#	8012	8016#	8017	8018#	8019	8022#
8023	8024#	8025	8032#	8034#	8035	8043#	8044#	8045	8046#	8047	8054#	8055
8058#	8059	8060#	8061#	8062	8069#	8070	8077#	8078	8082#	8083	8098#	8099
8103#	8104	8115#	8116#	8117#	8118	8119#	8120	8123#	8124	8132#	8133	8136#
8137#	8138#	8139	8140#	8141	8155#	8156	8157#	8158	8159#	8160#	8161#	8162
8163#	8164	8168#	8169	8170#	8171	8174#	8175#	8176#	8177	8178#	8179	8194#
8195	8210#	8211	8227#	8228	8240#	8241	8251#	8252	8253#	8254#	8255#	8256
8257#	8258	8277#	8278	8283#	8284	8291#	8292	8305#	8306	8310#	8311#	8312
8313#	8314	8322#	8323	8327#	8328#	8329	8333#	8334	8338#	8339	8354#	8355#
8356	8357#	8358	8361#	8362	8375#	8376	8387#	8388	8391#	8392#	8393	8394#
8395#	8396	8400#	8401	8404#	8405	8418#	8419#	8420	8424#	8425	8428#	8429
8439#	8440#	8441	8445#	8446	8450#	8451	8461#	8462#	8463	8464#	8465	8469#
8470	8474#	8475	8485#	8486	8487#	8488	8489#	8490#	8491	8492#	8493	8494#
8495	8504#	8505	8506#	8507	8511#	8518#	8525#	8532#	8539#	8546#	8553#	8559#
8560	8561#	8562	8563#	8564	8574#	8587#	8588	8589#	8590	8603#	8605#	8613#
8614	8615#	8616	8620#	8632#	8644#	8651#	8658#	8665#	8672#	8772#	8773	8774#
8775	8776#	8777	8786#	8787	8788#	8789	8790#	8791	8792#	8793	8800#	8802#
8803	8810#	8811	8819#	8821#	8822	8823#	8824	8825#	8826	8827#	8828#	8829
8830#	8831	8833#	8834	8837	8838#	8839	8849#	8850#	8851	8861#	8862	8866#
8867	8871#	8872	8888#	8889	8890#	8891	8905#	8906	8907#	8908	8912#	3913
8916#	8917	8931#	8932	8936#	8937	8940#	8941	8955#	8956	8960#	8961	8975#
8976	8980#	8981	8993#	8994#	8995	8998#	8999#	9000	9001#	9002#	9003	9010#
9011#	9012	9013#	9014#	9015	9019#	9020	9030#	9031	9035#	9036	9044#	9045
9064#	9065	9082#	9083	9095#	9096	9109#	9110	9114#	9115	9125#	9126	9129#
9130	9131#	9132#	9133#	9134	9145#	9146	9149#	9150	9151#	9152	9168#	9169
9170#	9171	9172#	9173#	9174	9175#	9176	9180#	9181	9182#	9183	9186#	9187
9188#	9189	9202#	9203	9204#	9205	9206#	9207	9208#	9209	9210#	9211	9212#
9213	9214#	9215	9216#	9217	9218#	9219	9220#	9221	9222#	9223	9224#	9225
9233#	9234	9235#	9236	9240#	9247#	9254#	9261#	9268#	9275#	9282#	9288#	9289
9290#	9291	9292#	9293	9303#	9310#	9311	9312#	9313	9327#	9329#	9337#	9338
9339#	9340	9344#	9356#	9368#	9375#	9382#	9389#	9540#	9542#	9543	9544#	9545
9555#	9556	9559#	9560#	9561	9576#	9577	9578#	9579	9580#	9581	9582#	9583
9584#	9585	9586#	9587	9588#	9589	9590#	9591	9592#	9593	9594#	9595	9596#
9597	9598#	9599	9616#	9617	9618#	9619	9623#	9624	9629#	9630	9640#	9641
9644#	9645	9646#	9647	9656#	9657	9658#	9659	9660#	9661	9669#	9670	9675#
9676	9679#	9680	9681#	9682	9691#	9692	9693#	9694	9695#	9696	9704#	9705
9710#	9711	9714#	9715	9716#	9717	9726#	9727	9728#	9729	9730#	9731	9739#
9740	9745#	9746	9749#	9750	9751#	9752	9761#	9762	9763#	9764	9765#	9766
9774#	9775	9780#	9781	9784#	9785	9786#	9787	9796#	9797	9798#	9799	9800#
9801	9809#	9810	9815#	9816	9819#	9820	9828#	9829	9830#	9831	9840#	9841

9846#	9847	9848#	9849	9858#	9859	9864#	9865	9866#	9867	9876#	9877	9882#
9883	9884#	9885	9889#	9890	9891#	9892	9898#	9899	9904#	9905	9909#	9910
9911#	9912#	9913	9914#	9942#	9944#	9945	9946#	9947	9948#	9949	9960#	9961
9964#	9965	9966#	9967	9976#	9977	9982#	9983	9986#	9987	9988#	9989	9998#
9999	10004#	10005	10008#	10009	10010#	10011	10020#	10021	10026#	10027	10030#	10031
10032#	10033	10042#	10043	10048#	10049	10052#	10053	10054#	10055	10064#	10065	10070#
10071	10074#	10075	10084#	10085	10090#	10091	10094#	10095	10104#	10105	10110#	10111
10114#	10115	10124#	10125	10130#	10131	10134#	10135	10144#	10145	10148#	10149	10155#
10156	10157#	10158	10162#	10163	10168#	10207#	10208#	10209	10222#	10223	10224#	10225
10226#	10227	10228#	10229	10241#	10242	10243#	10244	10245#	10246	10252#	10253	10255#
10256	10259	10268#	10275#	10276	10277#	10278	10279#	10280	10281#	10282	10283#	10284
10285#	10286	10287#	10288	10289#	10290	10299#	10300	10301#	10302	10303#	10304	10305#
10306	10316#	10317	10328#	10329	10330#	10331	10332#	10333	10334#	10335	10345#	10346
10357#	10358	10359#	10360	10361#	10362	10363#	10364	10376#	10377	10385#	10386	10387#
10388#	1038#	10390#	10391	10392#	10393	10394#	10395#	10396	10397#	10398	10399#	10400
10401#	10402#	10403	10404#	10405	10406#	10407	10408#	10409	10410#	10411	10430#	10431
10432#	10433#	10434	10435#	10436	10457#	10458	10462#	10463	10467#	10468	10471#	10472
10477#	10478#	10479	10480#	10481	10485#	10486	10489#	10490	10495#	10496#	10497	10498#
10499	10503#	10504	10507#	10508	10514#	10515#	10516	10517#	10518#	10519	10523#	10524
10528#	10529	10536#	10537#	10538	10539#	10540#	10541	10542#	10543	10547#	10548	10552#
10553	10577#	10578	10579#	10580#	10581	10582#	10583	10604#	10605	10609#	10610	10614#
10615	10618#	10619	10624#	10625#	10626	10627#	10628	10632#	10633	10636#	10637	10642#
10643#	10644	10645#	10646	10650#	10651	10654#	10655	10660#	10661#	10662	10663#	10664#
10665	10669#	10670	10674#	10675	10682#	10683#	10684	10685#	10686#	10687	10688#	10689
10693#	10694	10698#	10699	10723#	10724	10725#	10726#	10727	10728#	10729	10750#	10751
10758#	10759	10763#	10764	10767#	10768	10775#	10776#	10777	10781#	10782	10783#	10784
10788#	10789	10792#	10793	10800#	10801#	10802	10806#	10807	10808#	10809	10813#	10814
10817#	10818	10825#	10826#	10827	10828#	10829#	10830	10834#	10835	10839#	10840	10847#
10848#	10849	10850#	10851#	10852	10853#	10854	10858#	10859	10863#	10864	10875#	10876
10877#	10878	10883#	10884	10885#	10886	10887#	10888	10889#	10890	10891#	10892	10893#
10894	10895#	10896	10897#	10898	10906#	10907	10908#	10909	10913#	10920#	10927#	10934#
10941#	10948#	10955#	10964#	10965	10966#	10967	10968#	10969	10979#	11001#	11002	11003#
11004	11018#	11025#	11026	11027#	11028#	11029	11030#	11031	11041#	11042	11043#	11044#
11045	11046#	11047	11057#	11058	11059#	11060#	11061	11062#	11063	11071#	11082#	11083
11084#	11085	11089#	11106#	11118#	11127#	11128	11129#	11130	11131#	11132	11142#	11162#
11163	11164#	11165	11166#	11167	11173#	11174	11176#	11177	11180	11189#	11196#	11197
11198#	11199	11200#	11201	11202#	11203	11204#	11205	11206#	11207	11208#	11209	11210#
11211	11212#	11213	11214#	11215	11224#	11225	11226#	11227	11228#	11229	11230#	11231
11241#	11242	11253#	11254	11255#	11256	11257#	11258	11259#	11260	11270#	11271	11282#
11283	11284#	11285	11286#	11287	11288#	11289	11301#	11302	11310#	11311	11312#	11313#
11314	11315#	11316	11317#	11318	11319#	11320#	11321	11322#	11323	11324#	11325	11326#
11327#	11328	11329#	11330	11331#	11332	11333#	11334	11335#	11336	11355#	11356	11357#
11358#	11359	11360#	11361	11382#	11383	11387#	11388	11392#	11393	11396#	11397	11402#
11403#	11404	11405#	11406	11410#	11411	11414#	11415	11420#	11421#	11422	11423#	11424
11428#	11429	11432#	11433	11439#	11440#	11441	11442#	11443#	11444	11448#	11449	11453#
11454	11461#	11462#	11463	11464#	11465#	11466	11467#	11468	11472#	11473	11477#	11478
11497#	11498	11504#	11505	11506#	11507#	11508	11509#	11510	11531#	11532	11536#	11537
11541#	11542	11545#	11546	11551#	11552#	11553	11554#	11555	11559#	11560	11563#	11564
11569#	11570#	11571	11572#	11573	11577#	11578	11581#	11582	11587#	11588#	11589	11590#
11591#	11592	11596#	11597	11601#	11602	11609#	11610#	11611	11612#	11613#	11614	11615#
11616	11620#	11621	11625#	11626	11650#	11651	11652#	11653#	11654	11655#	11656	11677#
11678	11685#	11686	11690#	11691	11694#	11695	11702#	11703#	11704	11708#	11709	11710#
11711	11715#	11716	11719#	11720	11727#	11728#	11729	11733#	11734	11735#	11736	11740#
11741	11744#	11745	11752#	11753#	11754	11755#	11756#	11757	11761#	11762	11766#	11767
11774#	11775#	11776	11777#	11778#	11779	11780#	11781	11785#	11786	11790#	11791	11802#
11803	11804#	11805	11810#	11811	11812#	11813	11814#	11815	11816#	11817	11818#	11819

11820#	11821	11822#	11823	11824#	11825	11833#	11834	11835#	11836	11840#	11847#	11854#
11861#	11868#	11875#	11882#	11892#	11893	11894#	11895	11896#	11897	11907#	11929#	11930
11931#	11932	11946#	11953#	11954	11955#	11956#	11957	11958#	11959	11969#	11970	11971#
11972#	11973	11974#	11975	11985#	11986	11987#	11988#	11989	11990#	11991	11999#	12010#
12011	12012#	12013	12017#	12034#	12046#	12055#	12056	12057#	12058	12059#	12060	12070#
12090#	12091	12092#	12093	12094#	12095	12099#	12101#	12102	12103#	12104	12105#	12106
12112#	12113	12114#	12115	12116#	12117	12118#	12119	12120#	12121	12127#	12128#	12129
12130#	12131	12132#	12133	12134#	12135	12137#	12138	12141	12142#	12143	12150#	12151
12.54#	12155#	12156	12157#	12158	12159#	12160	12161#	12162	12164#	12165	12168	12169#
12170	12174#	12175	12187#	12188	12197#	12198	12207#	12208	12226#	12227	12233#	12234
12240#	12241	12250#	12251	12264#	12265	12266#	12267	12268#	12269	12276#	12277	12289#
12290	12299#	12300	12308#	12309	12335#	12336	12340#	12341	12342#	12343#	12344	12345#
12346	12351#	12352#	12353	12357#	12358	12362#	12363	12366#	12367	12368#	12369	12378#
12379#	12380	12384#	12385	12389#	12390	12393#	12394	12395#	12396	12407#	12408	12412#
12413	12417#	12418	12421#	12422	12423#	12424	12432#	12433#	12434	12438#	12439	12443#
12444	12452#	12453	12457#	12458	12462#	12463	12466#	12467#	12468	12469#	12470	12473#
12474	12481#	12482	12486#	12487	12488#	12489#	12490	12491#	12492	12497#	12498#	12499
12503#	12504	12508#	12509	12512#	12513	12514#	12515	12521#	12522#	12523	12527#	12528
12532#	12533	12536#	12537	12538#	12539	12545#	12546	12550#	12551	12555#	12556	12559#
12560	12561#	12562	12568#	12569#	12570	12574#	12575	12579#	12580	12588#	12589	12593#
12594	12598#	12599	12602#	12603#	12604	12605#	12606	12614#	12615#	12616	12617#	12618
12619#	12620#	12621	12622#	12623	12624#	12625	12626#	12627	12635#	12636	12637#	12638
12642#	12649#	12656#	12663#	12670#	12677#	12684#	12688#	12689	12698#	12699	12709#	12710
12711#	12712	12727#	12728	12729#	12730	12731#	12732	12733#	12734	12735#	12736	12737#
12738	12745#	12746	12747#	12748	12752#	12764#	12776#	12808#	12809	12810#	12811	12812#
12813	12814#	12815	12819#	12820	12821#	12822	12828#	12829	12830#	12831	12832#	12833
12834#	12835	12836#	12843#	12844	12845#	12846	12851#	12852	12853#	12854	12865#	12866
12870#	12871	12874#	12875	12884#	12885	12894#	12895	12896#	12897	12899#	12900	12903
12904#	12905	12917#	12918	12928#	12929#	12930	12931#	12932#	12933	12934#	12935	12936#
12937	12944#	12945	12949#	12950	12958#	12959	12968#	12969	12975#	12976	12979#	12980
12981#	12982	12991#	12992	13003#	13004#	13005	13009#	13010	13016#	13017	13018#	13019
13020#	13021	13023#	13024	13027	13028#	13029	13036#	13037#	13038	13039#	13040	13042#
13043	13046	13047#	13048	13052#	13053	13060#	13061	13062#	13063	13076#	13077	13083#
13084	13085#	13086#	13087#	13088	13089#	13090	13105#	13106	13121#	13122	13123#	13124#
13125	13126#	13127	13134#	13135	13136#	13137#	13138	13139#	13140	13147#	13148	13152#
13153#	13154	13158#	13159	13163#	13164	13169#	13170	13174#	13175#	13176	13185#	13186
13190#	13191	13201#	13202	13212#	13213	13217#	13218#	13219	13220#	13221	13227#	13228#
13229	13230#	13231	13247#	13248	13251#	13252#	13253	13257#	13258	13262#	13263	13271#
13272#	13273	13282#	13283	13287#	13288	13298#	13299	13302#	13303	13304#	13305	13306#
13307	13321#	13322	13326#	13327	13343#	13344	13348#	13349	13359#	13360	13363#	13364
13365#	13366	13367#	13368	13379#	13380#	13381	13385#	13386	13390#	13391	13397#	13398#
13399	13406#	13407	13411#	13412	13422#	13423	13429#	13430	13438#	13439#	13440	13441#
13442	13443#	13444#	13445	13446#	13447	13454#	13455	13459#	13460	13469#	13470	13474#
13475	13485#	13486	13492#	13493	13503#	13504#	13505	13506#	13507	13508#	13509#	13510
13511#	13512	13513#	13514	13515#	13516	13524#	13525	13526#	13527	13531#	13543#	13547#
13548#	13549	13553#	13560#	13565#	13566	13567#	13568	13569#	13570	13571#	13572	13582#
13583	13586#	13587	13594#	13604#	13605	13607#	13608	13611	13612#	13613	13617#	13618
13623#	13624	13625#	13626	13643#	13646#	13647	13648#	13649	13650#	13651	13652#	13653
13654#	13655	13656#	13657	13658#	13659	13660#	13661	13662#	13663	13664#	13665	13666#
13667	13668#	13669	13678#	13679	13680#	13681	13685#	13697#	13709#	13714#	13715	13719#
13726#	13730#	13731	13741#	13743#	13744	13797#	13798#	13799	13812#	13813	13814#	13815
13816#	13817	13818#	13819	13831#	13832	13833#	13834	13835#	13836	13842#	13843	13845#
13846	13849	13858#	13865#	13866	13867#	13868	13869#	13870	13871#	13872	13873#	13874
13875#	13876	13877#	13878	13879#	13880	13886#	13887	13888#	13898#	13908#	13923#	13924#
13925	13926#	13927	13928#	13929#	13930	13931#	13932	13933#	13934#	13935	13936#	13937
13944#	13945	13946#	13947	13955#	13956	13957#	13958	13962#	13969#	13976#	13983#	13990#

13997#	14004#	14013#	14014	14015#	14016	14017#	14018	14028#	14035#	14036	14037#	14038
14052#	14059#	14060	14061#	14062	14063#	14064	14065#	14066	14067#	14068	14069#	14070
14071#	14072	14078#	14079	14080#	14090#	14100#	14111#	14123#	14124	14125#	14126	14130#
14147#	14159#	14168#	14169	14170#	14171	14172#	14173#	14174	14177#	14178	14179#	14180
14181#	14182	14192#	14212#	14213	14214#	14215	14216#	14217	14223#	14224	14226#	14227
14230	14239#	14246#	14247	14248#	14249	14250#	14251	14252#	14253	14254#	14255	14256#
14257	14258#	14259	14260#	14261	14262#	14263	14264#	14265	14271#	14272	14273#	14283#
14293#	14309#	14310#	14311	14312#	14313	14314#	14315#	14316	14317#	14318	14319#	14320#
14321	14322#	14323	14330#	14331	14332#	14333	14341#	14342	14343#	14344	14348#	14355#
14362#	14369#	14376#	14383#	14390#	14400#	14401	14402#	14403	14404#	14405	14415#	14422#
14423	14424#	14425	14439#	14446#	14447	14448#	14449	14450#	14451	14452#	14453	14454#
14455	14456#	14457	14458#	14459	14460#	14461	14462#	14463	14469#	14470	14471#	14481#
14491#	14501#	14511#	14512	14513#	14514	14518#	14535#	14547#	14556#	14557	14558#	14559
14560#	14561#	14562	14565#	14566	14567#	14568	14569#	14570	14580#	14600#	14601	14602#
14603	14604#	14605	14609#	14611#	14612	14613#	14614	14615#	14616	14622#	14623	14624#
14625	14626#	14627	14628#	14629	14630#	14631	14638#	14639#	14640	14641#	14642	14643#
14644	14645#	14646	14648#	14649	14652	14653#	14654	14658#	14659	14660#	14661#	14662
14663#	14664	14665#	14666	14667#	14668	14670#	14671	14674	14675#	14676	14680#	14681
14693#	14694	14703#	14704	14713#	14714	14732#	14733	14739#	14740	14746#	14747	14756#
14757	14770#	14771	14772#	14773	14774#	14775	14782#	14783	14795#	14796	14805#	14806
14814#	14815	14832#	14835#	14836	14837#	14838#	14839	14840#	14841	14848#	14849	14853#
14854	14863#	14864	14865#	14866	14873#	14874	14877#	14878	14879#	14880	14881#	14882
14886#	14887	14893#	14894	14898#	14899	14902#	14903	14907#	14908	14909#	14910#	14911
14912#	14913	14920#	14921	14925#	14926	14935#	14936	14937#	14938	14945#	14946	14949#
14950	14951#	14952	14953#	14954	14958#	14959	14965#	14966	14970#	14971	14974#	14975
14983#	14984#	14985	14986#	14987	14988#	14989#	14990	14991#	14992	14993#	14994	14995#
14996	15004#	15005	15006#	15007	15011#	15023#	15035#	15039#	15040	15041#	15042	15058#
15059	15060#	15061	15062#	15063	15064#	15065	15066#	15067	15068#	15069	15076#	15077
15078#	15079	15083#	15095#	15107#	15139#	15140	15141#	15142	15143#	15144	15145#	15146
15150#	15152#	15153	15159#	15160	15161#	15162	15163#	15164	15165#	15166	15174#	15175
15176#	15177	15184#	15185	15189#	15190	15199#	15200	15201#	15202	15209#	15210	15211#
15212	15219#	15220	15224#	15225	15237#	15238	15242#	15243	15246#	15247	15256#	15257
15263#	15264	15267#	15268	15275#	15276	15280#	15281	15293#	15294	15307#	15308#	15309
15310#	15311	15314#	15315	15316#	15317	15318#	15319	15326#	15327	15331#	15332	15333#
15334	15335#	15336	15339#	15340	15341#	15342	15350#	15351	15352#	15353	15354#	15355
15358#	15359	15360#	15361	15373#	15374	15380#	15381	15384#	15385	15389#	15390	15393#
15394	15395#	15396	15405#	15406	15425#	15426#	15427	15431#	15432	15438#	15439	15440#
15441	15445#	15446#	15447	15450#	15451	15454#	15455#	15456	15457#	15458	15459#	15460
15461#	15462#	15463	15473#	15474	15475#	15476	15480#	15481	15484#	15485	15486#	15487
15488#	15489	15506#	15507#	15508	15509#	15510	15511#	15512#	15513	15514#	15515	15516#
15517	15518#	15519	15523#	15524	15525#	15526	15530#	15540#	15550#	15554#	15555#	15556
15560#	15567#	15573#	15574	15575#	15576	15577#	15578	15579#	15580	15590#	15591	15594#
15595	15602#	15606#	15607	15612#	15613	15614#	15615	15633#	15635#	15636	15637#	15638
15639#	15640	15641#	15651#	15652	15653#	15654	15658#	15670#	15682#	15687#	15688	15692#
15699#	15703#	15704	15714#	15715	15716#	15769#	15770#	15771	15784#	15785	15786#	15787
15788#	15789	15790#	15791	15804#	15805	15806#	15807	15808#	15809	15815#	15816	15818#
15819	15822	15831#	15838#	15839	15840#	15841	15842#	15843	15844#	15845	15846#	15847
15848#	15849	15850#	15851	15852#	15853	15859#	15860	15861#	15871#	15881#	15896#	15897#
15898	15899#	15900	15901#	15902#	15903	15904#	15905	15906#	15907#	15908	15909#	15910
15917#	15918	15919#	15920	15928#	15929	15930#	15931	15935#	15942#	15949#	15956#	15963#
15970#	15977#	15987#	15988	15989#	15990	15991#	15992	16002#	16009#	16010	16011#	16012
16026#	16033#	16034	16035#	16036	16037#	16038	16039#	16040	16041#	16042	16043#	16044
16045#	16046	16052#	16053	16054#	16064#	16074#	16084#	16095#	16096	16097#	16098	16102#
16119#	16131#	16140#	16141	16142#	16143	16144#	16145#	16146	16149#	16150	16151#	16152
16153#	16154	16164#	16202#	16203#	16204	16217#	16218	16219#	16220	16221#	16222	16223#
16224	16237#	16238	16239#	16240	16241#	16242	16248#	16249	16251#	16252	16255	16264#

16271#	16272	16273#	16274	16275#	16276	16277#	16278	16279#	16280	16281#	16282	16283#
16284	16285#	16286	16287#	16288	16289#	16290	16296#	16297	16298#	16308#	16318#	16334#
16335#	16336	16337#	16338	16339#	16340#	16341	16342#	16343	16344#	16345#	16346	16347#
16348	16355#	16356	16357#	16358	16366#	16367	16368#	16369	16373#	16380#	16387#	16394#
16401#	16408#	16415#	16425#	16426	16427#	16428	16429#	16430	16440#	16447#	16448	16449#
16450	16464#	16471#	16472	16473#	16474	16475#	16476	16477#	16478	16479#	16480	16481#
16482	16483#	16484	16485#	16486	16487#	16488	16494#	16495	16496#	16506#	16516#	16526#
16537#	16538	16539#	16540	16544#	16561#	16573#	16582#	16583	16584#	16585	16586#	16587#
16588	16591#	16592	16593#	16594	16595#	16596	16606#	16864#	16865	16866#	16867	16979#
16980	16981#	16982	16983#	16984	16985#	16986	16987#	16988	16989#	16990	17008#	17009
17010#	17011	17012#	17013	17014#	17015	17018#	17019	17020#	17021	17025#	17026	17027#
17028	17045#	17046	17047#	17048	17049#	17050	17051#	17052	17055#	17056	17057#	17058
17062#	17063	17064#	17065	17082#	17083	17084#	17085	17086#	17087	17088#	17089	17092#
17093	17094#	17095	17099#	17100	17101#	17102	17117#	17118	17119#	17120	17121#	17122
17123#	17124	17127#	17128	17129#	17130	17134#	17135	17136#	17137	17155#	17156	17157#
17158	17159#	17160	17161#	17162	17165#	17166	17167#	17168	17172#	17173	17174#	17175
17191#	17192	17193#	17194	17195#	17196	17197#	17198	17199#	17200	17201#	17202	17218#
17219	17220#	17221	17222#	17223	17224#	17225	17226#	17227	17228#	17229	17254#	17255
17265#	17266	17273#	17274	17287#	17288	17289#	17290	17291#	17292	17293#	17294	17295#
17296	17297#	17298	17299#	17300	17301#	17302	17304#	17305	17306#	17307	17308#	17309
17310#	17311	17312#	17313	17314#	17315	17316#	17317	17318#	17319	17330#	17331	17338#
17339	17390#	17391	17392#	17393	17394#	17395#	17396	17405#	17406	17410#	17411	17425#
17426	17433#	17434	17435#	17436	17443#	17444	17447#	17448	17449#	17450	17451#	17452
17456#	17457	17460#	17461	17462#	17463							
\$ERMAX	000416											
\$ERRPC	000420											
\$ERRTY	060200											
\$ERTTL	000414											
\$ETABL	000652											
\$ETEND	000664											
\$FATAL	000634											
\$FFLG	062152											
\$FILLC	000434											
\$FILLS	000433											
\$F\$AND	000310											
1045#												
1047#	17568*	17569*	17612									
17572	17601#											
1044#	17567*											
1144#												
1029	1162#											
1137#	17976*											
17944*	17947*	17970	17979*	17987#								
1055#	17680											
1054#												
720#	1269	1272	1314	1345	1348	1358	1367	1374	1381	1388	1395	1402
1405	1415	1450	1457	1464	1471	1478	1485	1534	1565	1568	1578	1587
1594	1601	1608	1615	1622	1653	1660	1667	1674	1681	1688	1695	1762
1780	1808	1816	1832	1840	1854	1882	1889	1896	1903	1910	1917	1924
1942	1948	1953	1971	1976	1996	2015	2018	2038	2058	2111	2118	2125
2132	2139	2146	2153	2160	2174	2378	2396	2424	2432	2448	2456	2488
2495	2502	2509	2516	2523	2530	2546	2556	2561	2578	2605	2610	2622
2630	2653	2706	2713	2720	2727	2734	2741	2748	2755	2768	2891	2919
2927	2943	2951	3009	3016	3023	3030	3037	3044	3051	3073	3079	3084
3102	3107	3128	3145	3157	3177	3232	3239	3246	3253	3260	3267	3274
3281	3295	3417	3475	3482	3489	3496	3503	3510	3517	3529	3571	3578
3585	3592	3599	3606	3703	3761	3768	3775	3782	3789	3796	3803	3815
3858	3865	3872	3879	3886	3893	3983	4026	4057	4064	4071	4078	4085
4092	4099	4111	4136	4169	4176	4183	4190	4197	4204	4296	4339	4367
4374	4381	4388	4395	4402	4409	4421	4446	4476	4483	4490	4497	4504
4511	4616	4657	4667	4677	4702	4709	4716	4723	4730	4737	4744	4756
4796	4803	4810	4817	4824	4831	4838	4964	5000	5006	5049	5056	5063
5070	5077	5084	5091	5103	5151	5158	5165	5172	5179	5186	5193	5240
5243	5360	5394	5410	5417	5424	5431	5438	5445	5452	5477	5482	5502
5517	5528	5572	5627	5680	5683	5750	5757	5766	5769	5786	5797	5812
5834	5837	5849	5872	5881	5916	5961	5968	5975	5982	5989	5996	6003
6023	6028	6043	6069	6076	6083	6090	6097	6104	6111	6120	6129	6139

6150	6282	6287	6294	6297	6313	6316	6334	6344	6358	6369	6378	6388
6408	6421	6427	6442	6455	6461	6469	6500	6545	6552	6559	6566	6573
6580	6587	6607	6612	6627	6653	6660	6667	6674	6681	6688	6695	6704
6715	6832	6868	6887	6899	6944	6950	6962	6969	7015	7022	7029	7036
7043	7050	7057	7070	7095	7102	7109	7116	7123	7130	7137	7144	7151
7257	7293	7326	7329	7340	7351	7357	7386	7400	7408	7414	7444	7449
7491	7498	7505	7512	7519	7526	7533	7546	7571	7578	7580	7590	7592
7602	7609	8008	8065	8076	8097	8154	8167	8193	8207	8223	8239	8282
8297	8309	8321	8326	8332	8350	8353	8369	8372	8381	8384	8399	8414
8417	8423	8438	8444	8460	8468	8510	8517	8524	8531	8538	8545	8552
8568	8573	8594	8619	8626	8628	8638	8640	8650	8657	8664	8671	8857
8860	8865	8884	8887	8901	8904	8911	8927	8930	8935	8951	8954	8971
8974	9018	9026	9043	9056	9061	9078	9094	9108	9144	9166	9179	9239
9246	9253	9260	9267	9274	9281	9297	9302	9317	9343	9350	9352	9362
9364	9374	9381	9388	9564	9569	9615	9622	9639	9653	9674	9688	9709
9723	9744	9758	9779	9793	9814	9827	9845	9863	9881	9903	9959	9981
10003	10025	10047	10069	10089	10109	10129	10154	10161	10212	10215	10298	10327
10356	10420	10461	10484	10502	10522	10546	10567	10608	10631	10649	10668	10692
10713	10754	10757	10780	10787	10805	10812	10833	10857	10912	10919	10926	10933
10940	10947	10954	10973	10978	11008	11088	11100	11102	11112	11114	11136	11141
11223	11252	11281	11345	11386	11409	11427	11447	11471	11492	11535	11558	11576
11595	11619	11640	11681	11684	11707	11714	11732	11739	11760	11784	11839	11846
11853	11860	11867	11874	11881	11901	11906	11936	12016	12028	12030	12040	12042
12064	12069	12149	12184	12194	12204	12217	12222	12225	12237	12247	12258	12273
12283	12286	12296	12334	12350	12356	12374	12377	12383	12403	12406	12411	12431
12437	12451	12456	12480	12496	12502	12520	12526	12544	12549	12567	12573	12587
12592	12641	12648	12655	12662	12669	12676	12683	12693	12703	12716	12751	12758
12760	12770	12772	12850	12869	12881	12893	12913	12927	12940	12943	12957	12965
12972	12988	13002	13008	13015	13057	13075	13111	13120	13133	13146	13151	13157
13173	13179	13181	13196	13198	13211	13216	13226	13238	13241	13244	13256	13270
13276	13278	13293	13295	13314	13317	13320	13334	13337	13339	13354	13356	13375
13384	13402	13417	13419	13428	13437	13453	13465	13480	13482	13491	13530	13537
13539	13552	13559	13576	13581	13593	13622	13632	13684	13691	13693	13703	13705
13718	13725	13735	13740	13802	13805	13961	13968	13975	13982	13989	13996	14003
14022	14027	14042	14129	14141	14143	14153	14155	14186	14191	14347	14354	14361
14368	14375	14382	14389	14409	14414	14429	14517	14529	14531	14541	14543	14574
14579	14690	14700	14710	14723	14728	14731	14743	14753	14764	14779	14789	14792
14802	14844	14847	14862	14872	14892	14916	14919	14934	14944	14964	15010	15017
15019	15029	15031	15046	15082	15089	15091	15101	15103	15180	15183	15198	15208
15218	15241	15253	15271	15289	15299	15301	15322	15325	15349	15367	15372	15377
15402	15413	15415	15424	15430	15437	15467	15472	15529	15536	15546	15559	15566
15584	15589	15601	15611	15621	15657	15664	15666	15676	15678	15691	15698	15708
15713	15774	15777	15934	15941	15948	15955	15962	15969	15976	15996	16001	16016
16101	16113	16115	16125	16127	16158	16163	16207	16210	16372	16379	16386	16393
16400	16407	16414	16434	16439	16454	16543	16555	16557	16567	16569	16600	16605
16861	17264	17272	17286	17329	17404	17424	17432	17442				
720*	1264	1269	1272	1280	1282	1284	1298	1300	1314	1315	1324	1332
1334	1336	1345	1348	1358	1367	1374	1381	1388	1395	1402	1405	1415
1422	1432	1434	1450	1457	1464	1471	1478	1485	1518	1520	1534	1535
1544	1552	1554	1556	1565	1568	1578	1587	1594	1601	1608	1615	1622
1627	1653	1660	1667	1674	1681	1688	1695	1713	1715	1717	1727	1729
1731	1733	1735	1737	1739	1741	1750	1752	1754	1756	1758	1762	1763
1768	1774	1776	1783	1784	1789	1794	1796	1798	1800	1802	1804	1808
1816	1828	1832	1840	1854	1855	1860	1876	1878	1882	1889	1896	1903
1910	1917	1924	1934	1936	1938	1942	1943	1950	1953	1958	1966	1973
1976	1981	1986	1989	1996	2001	2011	2015	2018	2019	2024	2028	2031

SF SBAD- 000401

2038	2051	2053	2058	2069	2073	2077	2079	2081	2083	2085	2087	2089
2111	2118	2125	2132	2139	2146	2153	2160	2165	2167	2174	2330	2332
2334	2344	2346	2348	2350	2352	2354	2356	2358	2366	2368	2370	2372
2374	2378	2379	2384	2390	2392	2399	2400	2405	2410	2412	2414	2416
2418	2420	2424	2432	2444	2448	2456	2468	2482	2484	2488	2495	2502
2509	2516	2523	2530	2540	2542	2546	2547	2549	2551	2558	2561	2566
2578	2583	2585	2593	2595	2597	2599	2607	2610	2622	2623	2630	2635
2646	2648	2653	2664	2668	2672	2674	2676	2678	2680	2682	2684	2706
2713	2720	2727	2734	2741	2748	2755	2760	2762	2768	2834	2836	2838
2848	2850	2852	2854	2856	2858	2860	2862	2871	2873	2875	2877	2879
2881	2885	2887	2894	2895	2900	2905	2907	2909	2911	2913	2915	2919
2927	2939	2943	2951	2963	2970	2973	2974	2978	2983	2985	2987	2989
3003	3005	3009	3016	3023	3030	3037	3044	3051	3061	3063	3065	3067
3069	3073	3074	3081	3084	3089	3097	3104	3107	3113	3118	3121	3128
3133	3145	3146	3148	3150	3157	3170	3172	3177	3188	3192	3196	3198
3200	3202	3204	3206	3208	3232	3239	3246	3253	3260	3267	3274	3281
3286	3289	3295	3361	3363	3365	3374	3376	3378	3380	3388	3390	3392
3398	3400	3402	3407	3409	3411	3417	3418	3420	3426	3429	3434	3436
3438	3440	3444	3446	3448	3452	3455	3469	3471	3475	3482	3489	3496
3503	3510	3517	3522	3524	3529	3540	3546	3549	3557	3567	3571	3578
3585	3592	3599	3606	3645	3647	3649	3659	3661	3663	3665	3674	3676
3678	3684	3686	3688	3693	3695	3697	3703	3704	3706	3712	3715	3720
3722	3724	3726	3730	3732	3734	3738	3741	3755	3757	3761	3768	3775
3782	3789	3796	3803	3808	3810	3815	3826	3832	3835	3843	3853	3858
3865	3872	3879	3886	3893	3929	3931	3933	3943	3945	3947	3956	3958
3960	3962	3968	3970	3975	3977	3983	3984	3986	3992	3995	4000	4002
4004	4006	4010	4012	4014	4018	4021	4026	4032	4034	4037	4051	4053
4057	4064	4071	4078	4085	4092	4099	4104	4106	4111	4122	4126	4129
4131	4136	4142	4145	4155	4165	4169	4176	4183	4190	4197	4204	4242
4244	4246	4256	4258	4260	4269	4271	4273	4275	4281	4283	4288	4290
4296	4297	4299	4305	4308	4313	4315	4317	4319	4323	4325	4327	4331
4334	4339	4345	4347	4361	4363	4367	4374	4381	4388	4395	4402	4409
4414	4416	4421	4432	4436	4439	4441	4446	4447	4454	4462	4472	4476
4483	4490	4497	4504	4511	4549	4551	4553	4563	4565	4567	4569	4576
4578	4580	4582	4584	4586	4592	4597	4606	4610	4616	4617	4620	4625
4628	4632	4639	4642	4645	4648	4651	4653	4657	4658	4660	4667	4668
4670	4679	4680	4696	4698	4702	4709	4716	4723	4730	4737	4744	4749
4751	4756	4767	4771	4777	4780	4790	4792	4796	4803	4810	4817	4824
4831	4838	4899	4901	4903	4913	4915	4917	4924	4926	4928	4930	4936
4938	4943	4945	4954	4958	4964	4965	4971	4977	4986	4988	4990	4992
4994	4996	5000	5001	5006	5007	5009	5015	5018	5027	5029	5043	5045
5049	5056	5063	5070	5077	5084	5091	5096	5098	5103	5114	5118	5122
5139	5151	5158	5165	5172	5179	5186	5193	5240	5243	5251	5253	5255
5257	5259	5261	5268	5270	5272	5283	5285	5287	5289	5291	5293	5296
5298	5300	5302	5304	5306	5309	5311	5313	5315	5317	5319	5321	5326
5335	5338	5339	5343	5348	5350	5352	5354	5360	5364	5372	5374	5376
5394	5404	5406	5410	5417	5424	5431	5438	5445	5452	5462	5464	5466
5468	5470	5472	5479	5482	5487	5492	5495	5502	5507	5517	5518	5521
5528	5543	5545	5547	5549	5553	5566	5568	5572	5573	5578	5598	5600
5602	5604	5606	5608	5614	5627	5633	5635	5637	5675	5680	5683	5690
5692	5694	5696	5708	5710	5712	5722	5724	5726	5728	5737	5739	5743
5745	5750	5751	5753	5757	5758	5766	5769	5770	5774	5777	5779	5786
5787	5789	5791	5797	5798	5800	5802	5804	5806	5812	5813	5817	5821
5823	5825	5827	5834	5837	5838	5840	5844	5849	5850	5854	5857	5859
5866	5872	5873	5881	5882	5888	5891	5906	5908	5910	5912	5916	5917
5919	5924	5926	5930	5932	5934	5936	5938	5940	5955	5957	5961	5968

5975	5982	5989	5996	6003	6014	6016	6018	6025	6028	6036	6038	6043
6063	6065	6069	6076	6083	6090	6097	6104	6111	6116	6120	6129	6139
6144	6150	6237	6239	6241	6251	6253	6255	6257	6266	6268	6272	6274
6277	6282	6283	6287	6288	6294	6297	6298	6303	6305	6313	6316	6317
6322	6324	6336	6337	6339	6344	6345	6347	6352	6360	6361	6363	6365
6369	6370	6371	6373	6378	6379	6388	6389	6390	6392	6398	6401	6410
6411	6413	6415	6424	6427	6428	6433	6437	6442	6443	6445	6449	6458
6461	6462	6469	6470	6472	6477	6490	6492	6494	6496	6500	6501	6503
6508	6510	6514	6516	6518	6520	6522	6524	6539	6541	6545	6552	6559
6566	6573	6580	6587	6598	6600	6602	6609	6612	6620	6622	6627	6647
6649	6653	6660	6667	6674	6681	6688	6695	6700	6704	6709	6715	6801
6803	6805	6815	6817	6819	6826	6828	6832	6833	6835	6840	6842	6846
6848	6857	6859	6861	6863	6868	6869	6871	6873	6878	6880	6887	6888
6890	6892	6899	6900	6910	6912	6914	6917	6919	6924	6934	6939	6944
6945	6953	6954	6956	6958	6962	6963	6972	6973	6988	6990	6992	6994
7009	7011	7015	7022	7029	7036	7043	7050	7057	7063	7065	7070	7089
7091	7095	7102	7109	7116	7123	7130	7137	7144	7151	7226	7228	7230
7240	7242	7244	7251	7253	7257	7258	7260	7265	7267	7271	7273	7280
7282	7284	7286	7288	7293	7294	7296	7298	7303	7305	7307	7311	7313
7320	7326	7332	7333	7335	7340	7341	7343	7354	7357	7358	7363	7365
7369	7371	7375	7377	7382	7386	7387	7389	7393	7400	7401	7411	7414
7415	7417	7426	7429	7430	7434	7444	7445	7452	7453	7464	7466	7468
7470	7485	7487	7491	7498	7505	7512	7519	7526	7533	7539	7541	7546
7565	7567	7571	7583	7595	7602	7609	7973	7975	7977	7987	7989	7991
7993	8000	8002	8004	8008	8009	8011	8016	8018	8022	8024	8034	8043
8046	8054	8058	8060	8068	8069	8076	8077	8082	8097	8098	8103	8115
8117	8119	8123	8132	8136	8138	8140	8154	8155	8157	8159	8161	8163
8167	8168	8170	8174	8176	8178	8193	8194	8209	8210	8226	8227	8239
8240	8251	8253	8255	8257	8277	8282	8283	8291	8297	8305	8309	8310
8313	8321	8322	8326	8327	8332	8333	8338	8350	8353	8354	8357	8361
8369	8374	8375	8381	8386	8387	8391	8394	8399	8400	8404	8414	8417
8418	8423	8424	8428	8438	8439	8444	8445	8450	8460	8461	8464	8468
8469	8474	8485	8487	8489	8492	8494	8504	8506	8510	8517	8524	8531
8538	8545	8552	8559	8561	8563	8570	8573	8587	8589	8594	8613	8615
8619	8631	8643	8650	8657	8664	8671	8772	8774	8776	8786	8788	8790
8792	8802	8810	8821	8823	8825	8827	8830	8833	8834	8838	8849	8857
8860	8861	8865	8866	8871	8884	8887	8888	8890	8901	8904	8905	8907
8911	8912	8916	8927	8930	8931	8935	8936	8940	8951	8954	8955	8960
8971	8974	8975	8980	8993	8998	9001	9010	9013	9018	9019	9029	9030
9035	9043	9044	9058	9063	9064	9081	9082	9094	9095	9108	9109	9114
9125	9129	9131	9133	9144	9145	9149	9151	9166	9168	9170	9172	9175
9179	9180	9182	9186	9188	9202	9204	9206	9208	9210	9212	9214	9216
9218	9220	9222	9224	9233	9235	9239	9246	9253	9260	9267	9274	9281
9288	9290	9292	9299	9302	9310	9312	9317	9337	9339	9343	9355	9367
9374	9381	9388	9542	9544	9555	9559	9566	9569	9576	9578	9580	9582
9584	9586	9588	9590	9592	9594	9596	9598	9615	9616	9618	9622	9623
9629	9639	9640	9644	9646	9655	9656	9658	9660	9669	9674	9675	9679
9681	9690	9691	9693	9695	9704	9709	9710	9714	9716	9725	9726	9728
9730	9739	9744	9745	9749	9751	9760	9761	9763	9765	9774	9779	9780
9784	9786	9795	9796	9798	9800	9809	9814	9815	9819	9827	9828	9830
9840	9845	9846	9848	9858	9863	9864	9866	9876	9881	9882	9884	9889
9891	9898	9903	9904	9909	9911	9944	9946	9948	9959	9960	9964	9966
9976	9981	9982	9986	9988	9998	10003	10004	10008	10010	10020	10025	10026
10030	10032	10042	10047	10048	10052	10054	10064	10069	10070	10074	10084	10089
10090	10094	10104	10109	10110	10114	10124	10129	10130	10134	10144	10148	10154
10155	10157	10161	10162	10207	10212	10215	10222	10224	10226	10228	10241	10243

10245	10252	10255	10256	10275	10277	10279	10281	10283	10285	10287	10289	10298
10299	10301	10303	10305	10316	10327	10328	10330	10332	10334	10345	10356	10357
10359	10361	10363	10376	10385	10387	10390	10392	10394	10397	10399	10401	10404
10406	10408	10410	10420	10430	10432	10435	10442	10457	10461	10462	10467	10471
10477	10480	10484	10485	10489	10495	10498	10502	10503	10507	10514	10517	10522
10523	10528	10536	10539	10542	10546	10547	10552	10567	10577	10579	10582	10589
10604	10608	10609	10614	10618	10624	10627	10631	10632	10636	10642	10645	10649
10650	10654	10660	10663	10668	10669	10674	10682	10685	10688	10692	10693	10698
10713	10723	10725	10728	10735	10750	10754	10757	10758	10763	10767	10775	10780
10781	10783	10787	10788	10792	10800	10805	10806	10808	10812	10813	10817	10825
10828	10833	10834	10839	10847	10850	10853	10857	10858	10863	10875	10877	10883
10885	10887	10889	10891	10893	10895	10897	10906	10908	10912	10919	10926	10933
10940	10947	10954	10964	10966	10968	10975	10978	11001	11003	11008	11025	11027
11030	11041	11043	11046	11057	11059	11062	11082	11084	11088	11105	11117	11127
11129	11131	11138	11141	11162	11164	11166	11173	11176	11177	11196	11198	11200
11202	11204	11206	11208	11210	11212	11214	11223	11224	11226	11228	11230	11241
11252	11253	11255	11257	11259	11270	11281	11282	11284	11286	11288	11301	11310
11312	11315	11317	11319	11322	11324	11326	11329	11331	11333	11335	11345	11355
11357	11360	11367	11382	11386	11387	11392	11396	11402	11405	11409	11410	11414
11420	11423	11427	11428	11432	11439	11442	11447	11448	11453	11461	11464	11467
11471	11472	11477	11492	11497	11504	11506	11509	11516	11531	11535	11536	11541
11545	11551	11554	11558	11559	11563	11569	11572	11576	11577	11581	11587	11590
11595	11596	11601	11609	11612	11615	11619	11620	11625	11640	11650	11652	11655
11662	11677	11681	11684	11685	11690	11694	11702	11707	11708	11710	11714	11715
11719	11727	11732	11733	11735	11739	11740	11744	11752	11755	11760	11761	11766
11774	11777	11780	11784	11785	11790	11802	11804	11810	11812	11814	11816	11818
11820	11822	11824	11833	11835	11839	11846	11853	11860	11867	11874	11881	11892
11894	11896	11903	11906	11929	11931	11936	11953	11955	11958	11969	11971	11974
11985	11987	11990	12010	12012	12016	12033	12045	12055	12057	12059	12066	12069
12090	12092	12094	12101	12103	12105	12112	12114	12116	12118	12120	12127	12130
12132	12134	12137	12138	12142	12149	12150	12154	12157	12159	12161	12164	12165
12169	12174	12186	12187	12196	12197	12206	12207	12219	12222	12225	12226	12233
12239	12240	12249	12250	12258	12264	12266	12268	12275	12276	12283	12288	12289
12298	12299	12308	12334	12335	12340	12342	12345	12350	12351	12356	12357	12362
12366	12368	12374	12377	12378	12383	12384	12389	12393	12395	12403	12406	12407
12411	12412	12417	12421	12423	12431	12432	12437	12438	12443	12451	12452	12456
12457	12462	12466	12469	12473	12480	12481	12486	12488	12491	12496	12497	12502
12503	12508	12512	12514	12520	12521	12526	12527	12532	12536	12538	12544	12545
12549	12550	12555	12559	12561	12567	12568	12573	12574	12579	12587	12588	12592
12593	12598	12602	12605	12614	12617	12619	12622	12624	12626	12635	12637	12641
12648	12655	12662	12669	12676	12683	12688	12693	12698	12703	12709	12711	12716
12727	12729	12731	12733	12735	12737	12745	12747	12751	12763	12775	12808	12810
12812	12814	12819	12821	12828	12830	12832	12834	12843	12845	12850	12851	12853
12865	12869	12870	12874	12883	12884	12893	12894	12896	12899	12900	12904	12916
12917	12927	12928	12931	12934	12936	12940	12943	12944	12949	12957	12958	12967
12968	12974	12975	12979	12981	12990	12991	13002	13003	13008	13009	13015	13016
13018	13020	13023	13024	13028	13036	13039	13042	13043	13047	13052	13059	13060
13062	13075	13076	13083	13085	13087	13089	13105	13111	13120	13121	13123	13126
13133	13134	13136	13139	13146	13147	13151	13152	13157	13158	13163	13169	13173
13174	13184	13185	13190	13200	13201	13211	13212	13216	13217	13220	13226	13227
13230	13238	13241	13246	13247	13251	13256	13257	13262	13270	13271	13281	13282
13287	13297	13298	13302	13304	13306	13314	13317	13320	13321	13326	13334	13342
13343	13348	13358	13359	13363	13365	13367	13375	13379	13384	13385	13390	13397
13405	13406	13411	13421	13422	13428	13429	13437	13438	13441	13443	13446	13453
13454	13459	13468	13469	13474	13484	13485	13491	13492	13503	13506	13508	13511
13513	13515	13524	13526	13530	13542	13547	13552	13559	13565	13567	13569	13571

13578	13581	13582	13586	13593	13604	13607	13608	13612	13617	13622	13623	13625
13632	13646	13648	13650	13652	13654	13656	13658	13660	13662	13664	13666	13668
13678	13680	13684	13696	13708	13714	13718	13725	13730	13737	13740	13743	13797
13802	13805	13812	13814	13816	13818	13831	13833	13835	13842	13845	13846	13865
13867	13869	13871	13873	13875	13877	13879	13886	13923	13926	13928	13931	13933
13936	13944	13946	13955	13957	13961	13968	13975	13982	13989	13996	14003	14013
14015	14017	14024	14027	14035	14037	14042	14059	14061	14063	14065	14067	14069
14071	14078	14123	14125	14129	14146	14158	14168	14170	14172	14177	14179	14181
14188	14191	14212	14214	14216	14223	14226	14227	14246	14248	14250	14252	14254
14256	14258	14260	14262	14264	14271	14309	14312	14314	14317	14319	14322	14330
14332	14341	14343	14347	14354	14361	14368	14375	14382	14389	14400	14402	14404
14411	14414	14422	14424	14429	14446	14448	14450	14452	14454	14456	14458	14460
14462	14469	14511	14513	14517	14534	14546	14556	14558	14560	14565	14567	14569
14576	14579	14600	14602	14604	14611	14613	14615	14622	14624	14626	14628	14630
14638	14641	14643	14645	14648	14649	14653	14658	14660	14663	14665	14667	14670
14671	14675	14680	14692	14693	14702	14703	14712	14713	14725	14728	14731	14732
14739	14745	14746	14755	14756	14764	14770	14772	14774	14781	14782	14789	14794
14795	14804	14805	14814	14835	14837	14840	14844	14847	14848	14853	14862	14863
14865	14872	14873	14877	14879	14881	14886	14892	14893	14898	14902	14907	14909
14912	14916	14919	14920	14925	14934	14935	14937	14944	14945	14949	14951	14953
14958	14964	14965	14970	14974	14983	14986	14988	14991	14993	14995	15004	15006
15010	15022	15034	15039	15041	15046	15058	15060	15062	15064	15066	15068	15076
15078	15082	15094	15106	15139	15141	15143	15145	15152	15159	15161	15163	15165
15174	15176	15180	15183	15184	15189	15198	15199	15201	15208	15209	15211	15218
15219	15224	15237	15241	15242	15246	15255	15256	15263	15267	15274	15275	15280
15292	15293	15303	15307	15310	15314	15316	15318	15322	15325	15326	15331	15333
15335	15339	15341	15349	15350	15352	15354	15358	15360	15369	15372	15373	15379
15380	15384	15389	15393	15395	15404	15405	15417	15424	15425	15430	15431	15437
15438	15440	15445	15450	15454	15455	15457	15459	15461	15469	15472	15473	15475
15480	15484	15486	15488	15506	15509	15511	15514	15516	15518	15523	15525	15529
15539	15549	15554	15559	15566	15573	15575	15577	15579	15586	15589	15590	15594
15601	15606	15611	15612	15614	15621	15635	15637	15639	15651	15653	15657	15669
15681	15687	15691	15698	15703	15710	15713	15714	15769	15774	15777	15784	15786
15788	15790	15804	15806	15808	15815	15818	15819	15838	15840	15842	15844	15846
15848	15850	15852	15859	15896	15899	15901	15904	15906	15909	15917	15919	15928
15930	15934	15941	15948	15955	15962	15969	15976	15987	15989	15991	15998	16001
16009	16011	16016	16033	16035	16037	16039	16041	16043	16045	16052	16095	16097
16101	16118	16130	16140	16142	16144	16149	16151	16153	16160	16163	16202	16207
16210	16217	16219	16221	16223	16237	16239	16241	16248	16251	16252	16271	16273
16275	16277	16279	16281	16283	16285	16287	16289	16296	16334	16337	16339	16342
16344	16347	16355	16357	16366	16368	16372	16379	16386	16393	16400	16407	16414
16425	16427	16429	16436	16439	16447	16449	16454	16471	16473	16475	16477	16479
16481	16483	16485	16487	16494	16537	16539	16543	16560	16572	16582	16584	16586
16591	16593	16595	16602	16605	16863	16864	16866	16979	16981	16983	16985	16987
16989	17008	17010	17012	17014	17018	17020	17025	17027	17045	17047	17049	17051
17055	17057	17062	17064	17082	17084	17086	17088	17092	17094	17099	17101	17117
17119	17121	17123	17127	17129	17134	17136	17155	17157	17159	17161	17165	17167
17172	17174	17191	17193	17195	17197	17199	17201	17218	17220	17222	17224	17226
17228	17254	17264	17265	17272	17273	17286	17287	17289	17291	17293	17295	17297
17299	17301	17304	17306	17308	17310	17312	17314	17316	17318	17329	17330	17338
17390	17392	17394	17404	17405	17410	17424	17425	17432	17433	17435	17442	17443
17447	17449	17451	17456	17460	17462							

\$F\$BLA= 000170
 \$F\$CAS= 000150
 \$F\$DEC= 000220
 \$F\$DO = 000340

720#												
720#												
720#	2970	2980	5335	5345	12134	12144	12161	12171	14645	14655	14667	14677
720#	1950	1973	2558	2607	3081	3104	4026	4136	4339	4446	5360	5479

5849	6025	6344	6378	6609	6899	8154	8193	8282	8570	9166	9299	9639
9674	9709	9744	9779	9814	9827	9845	9863	9881	9903	9959	9981	10003
10025	10047	10069	10089	10109	10129	10975	11138	11903	12066	12275	12850	12967
13059	13578	13737	14024	14188	14411	14576	14781	14872	14944	15208	15369	15469
15586	15710	15998	16160	16436	16602	17442						
720#	10454	10601	10747	11379	11528	11674						
720#	1234	1264	1265	1266	1267	1270	1280	1281	1282	1283	1284	1285
1298	1299	1300	1301	1310	1312	1315	1316	1324	1325	1332	1333	1334
1335	1336	1337	1343	1346	1349	1356	1359	1365	1368	1372	1375	1379
1382	1386	1389	1393	1396	1400	1403	1406	1413	1416	1422	1423	1432
1433	1434	1435	1446	1448	1451	1455	1458	1462	1465	1469	1472	1476
1479	1483	1486	1518	1519	1520	1521	1530	1532	1535	1536	1544	1545
1552	1553	1554	1555	1556	1557	1563	1566	1569	1576	1579	1585	1588
1592	1595	1599	1602	1606	1609	1613	1616	1620	1623	1627	1628	1649
1651	1654	1658	1661	1665	1668	1672	1675	1679	1682	1686	1689	1693
1696	1713	1714	1715	1716	1717	1718	1727	1728	1729	1730	1731	1732
1733	1734	1735	1736	1737	1738	1739	1740	1741	1742	1750	1751	1752
1753	1754	1755	1756	1757	1758	1759	1760	1763	1764	1768	1769	1774
1775	1776	1777	1778	1784	1785	1789	1790	1791	1794	1795	1796	1797
1798	1799	1800	1801	1802	1803	1804	1805	1806	1809	1814	1817	1822
1828	1829	1830	1833	1838	1841	1846	1852	1855	1856	1860	1861	1876
1877	1878	1879	1880	1883	1887	1890	1894	1897	1901	1904	1908	1911
1915	1918	1922	1925	1934	1935	1936	1937	1938	1939	1940	1943	1944
1945	1946	1951	1954	1958	1959	1966	1967	1968	1969	1974	1977	1981
1982	1986	1987	1988	1989	1990	1991	1994	1997	2001	2002	2011	2012
2013	2016	2019	2020	2024	2025	2028	2029	2030	2031	2032	2033	2036
2039	2051	2052	2053	2054	2056	2069	2070	2071	2073	2074	2075	2077
2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090
2099	2101	2103	2109	2112	2116	2119	2123	2126	2130	2133	2137	2140
2144	2147	2151	2154	2158	2161	2165	2166	2167	2168	2169	2172	2175
2330	2331	2332	2333	2334	2335	2344	2345	2346	2347	2348	2349	2350
2351	2352	2353	2354	2355	2356	2357	2358	2359	2366	2367	2368	2369
2370	2371	2372	2373	2374	2375	2376	2379	2380	2384	2385	2390	2391
2392	2393	2394	2400	2401	2405	2406	2407	2410	2411	2412	2413	2414
2415	2416	2417	2418	2419	2420	2421	2422	2425	2430	2433	2438	2444
2445	2446	2449	2454	2457	2462	2468	2469	2482	2483	2484	2485	2486
2489	2493	2496	2500	2503	2507	2510	2514	2517	2521	2524	2528	2531
2540	2541	2542	2543	2544	2547	2548	2549	2550	2551	2552	2553	2554
2559	2562	2566	2567	2576	2579	2583	2584	2585	2586	2593	2594	2595
2596	2597	2598	2599	2600	2601	2602	2603	2608	2611	2620	2623	2624
2625	2628	2631	2635	2636	2637	2646	2647	2648	2649	2651	2664	2665
2666	2668	2669	2670	2672	2673	2674	2675	2676	2677	2678	2679	2680
2681	2682	2683	2684	2685	2694	2696	2698	2704	2707	2711	2714	2718
2721	2725	2728	2732	2735	2739	2742	2746	2749	2753	2756	2760	2761
2762	2763	2766	2769	2834	2835	2836	2837	2838	2839	2848	2849	2850
2851	2852	2853	2854	2855	2856	2857	2858	2859	2860	2861	2862	2863
2871	2872	2873	2874	2875	2876	2877	2878	2879	2880	2881	2882	2885
2886	2887	2888	2889	2895	2896	2900	2901	2902	2905	2906	2907	2908
2909	2910	2911	2912	2913	2914	2915	2916	2917	2920	2925	2928	2933
2939	2940	2941	2944	2949	2952	2957	2963	2964	2970	2971	2973	2977
2978	2979	2983	2984	2985	2986	2987	2988	2989	2990	3003	3004	3005
3006	3007	3010	3014	3017	3021	3024	3028	3031	3035	3038	3042	3045
3049	3052	3061	3062	3063	3064	3065	3066	3067	3068	3069	3070	3071
3074	3075	3076	3077	3082	3085	3089	3090	3097	3098	3099	3100	3105
3108	3113	3114	3118	3119	3120	3121	3122	3123	3126	3129	3133	3134
3143	3146	3147	3148	3149	3150	3151	3152	3155	3158	3170	3171	3172

SF\$FAL= 000405
SF\$GOO= 000400

3173	3175	3188	3189	3190	3192	3193	3194	3196	3197	3198	3199	3200
3201	3202	3203	3204	3205	3206	3207	3208	3209	3220	3222	3224	3230
3233	3237	3240	3244	3247	3251	3254	3258	3261	3265	3268	3272	3275
3279	3282	3286	3287	3288	3289	3290	3293	3296	3361	3362	3363	3364
3365	3366	3374	3375	3376	3377	3378	3379	3380	3381	3388	3389	3390
3391	3392	3393	3396	3398	3399	3400	3401	3402	3403	3407	3408	3409
3410	3411	3412	3415	3418	3419	3420	3421	3422	3426	3427	3428	3429
3430	3431	3434	3435	3436	3437	3438	3439	3440	3441	3444	3445	3446
3447	3448	3449	3450	3452	3453	3454	3455	3456	3469	3470	3471	3472
3473	3476	3480	3483	3487	3490	3494	3497	3501	3504	3508	3511	3515
3518	3522	3523	3524	3525	3527	3540	3541	3542	3544	3546	3547	3548
3549	3550	3557	3558	3561	3563	3567	3568	3569	3572	3576	3579	3583
3586	3590	3593	3597	3600	3604	3607	3645	3646	3647	3648	3649	3650
3659	3660	3661	3662	3663	3664	3665	3666	3674	3675	3676	3677	3678
3679	3682	3684	3685	3686	3687	3688	3689	3693	3694	3695	3696	3697
3698	3701	3704	3705	3706	3707	3708	3712	3713	3714	3715	3716	3717
3720	3721	3722	3723	3724	3725	3726	3727	3730	3731	3732	3733	3734
3735	3736	3738	3739	3740	3741	3742	3755	3756	3757	3758	3759	3762
3766	3769	3773	3776	3780	3783	3787	3790	3794	3797	3801	3804	3808
3809	3810	3811	3813	3826	3827	3828	3830	3832	3833	3834	3835	3836
3843	3844	3847	3849	3853	3854	3855	3856	3859	3863	3866	3870	3873
3877	3880	3884	3887	3891	3894	3929	3930	3931	3932	3933	3934	3943
3944	3945	3946	3947	3948	3956	3957	3958	3959	3960	3961	3962	3963
3966	3968	3969	3970	3971	3975	3976	3977	3978	3981	3984	3985	3986
3987	3988	3992	3993	3994	3995	3996	3997	4000	4001	4002	4003	4004
4005	4006	4007	4010	4011	4012	4013	4014	4015	4016	4018	4019	4020
4021	4022	4023	4024	4032	4033	4034	4035	4036	4037	4038	4051	4052
4053	4054	4055	4058	4062	4065	4069	4072	4076	4079	4083	4086	4090
4093	4097	4100	4104	4105	4106	4107	4109	4122	4123	4124	4126	4127
4128	4129	4130	4131	4132	4133	4134	4142	4143	4144	4145	4146	4147
4155	4156	4159	4161	4165	4166	4167	4170	4174	4177	4181	4184	4188
4191	4195	4198	4202	4205	4242	4243	4244	4245	4246	4247	4256	4257
4258	4259	4260	4261	4269	4270	4271	4272	4273	4274	4275	4276	4279
4281	4282	4283	4284	4288	4289	4290	4291	4294	4297	4298	4299	4300
4301	4305	4306	4307	4308	4309	4310	4313	4314	4315	4316	4317	4318
4319	4320	4323	4324	4325	4326	4327	4328	4329	4331	4332	4333	4334
4335	4336	4337	4345	4346	4347	4348	4361	4362	4363	4364	4365	4368
4372	4375	4379	4382	4386	4389	4393	4396	4400	4403	4407	4410	4414
4415	4416	4417	4419	4432	4433	4434	4436	4437	4438	4439	4440	4441
4442	4443	4444	4447	4448	4452	4454	4455	4462	4463	4466	4468	4472
4473	4474	4477	4481	4484	4488	4491	4495	4498	4502	4505	4509	4512
4549	4550	4551	4552	4553	4554	4563	4564	4565	4566	4567	4568	4569
4570	4576	4577	4578	4579	4580	4581	4582	4583	4584	4585	4586	4587
4590	4592	4593	4597	4598	4606	4607	4608	4610	4611	4612	4614	4617
4618	4619	4620	4621	4625	4626	4627	4628	4629	4632	4633	4639	4640
4641	4642	4643	4644	4645	4646	4647	4648	4649	4650	4651	4652	4653
4654	4655	4658	4659	4660	4661	4662	4665	4668	4669	4670	4671	4672
4675	4680	4681	4696	4697	4698	4699	4700	4703	4707	4710	4714	4717
4721	4724	4728	4731	4735	4738	4742	4745	4749	4750	4751	4752	4754
4767	4768	4769	4771	4772	4773	4775	4777	4778	4779	4780	4781	4790
4791	4792	4793	4794	4797	4801	4804	4808	4811	4815	4818	4822	4825
4829	4832	4836	4839	4899	4900	4901	4902	4903	4904	4913	4914	4915
4916	4917	4918	4924	4925	4926	4927	4928	4929	4930	4931	4934	4936
4937	4938	4939	4943	4944	4945	4946	4954	4955	4956	4958	4959	4960
4962	4965	4966	4971	4972	4973	4977	4978	4986	4987	4988	4989	4990
4991	4992	4993	4994	4995	4996	4997	4998	5001	5002	5003	5004	5007

5008	5009	5010	5011	5015	5016	5017	5018	5019	5020	5027	5028	5029
5030	5043	5044	5045	5046	5047	5050	5054	5057	5061	5064	5068	5071
5075	5078	5082	5085	5089	5092	5096	5097	5098	5099	5101	5114	5115
5116	5118	5119	5120	5122	5123	5131	5139	5140	5143	5145	5149	5152
5156	5159	5163	5166	5170	5173	5177	5180	5184	5187	5191	5194	5238
5241	5251	5252	5253	5254	5255	5256	5257	5258	5259	5260	5261	5262
5268	5269	5270	5271	5272	5273	5279	5283	5284	5285	5286	5287	5288
5289	5290	5291	5292	5293	5294	5296	5297	5298	5299	5300	5301	5302
5303	5304	5305	5306	5307	5309	5310	5311	5312	5313	5314	5315	5316
5317	5318	5319	5320	5321	5322	5324	5326	5327	5328	5335	5336	5338
5342	5343	5344	5348	5349	5350	5351	5352	5353	5354	5355	5357	5358
5364	5365	5372	5373	5374	5375	5376	5377	5392	5395	5404	5405	5406
5407	5408	5411	5415	5418	5422	5425	5429	5432	5436	5439	5443	5446
5450	5453	5462	5463	5464	5465	5466	5467	5468	5469	5470	5471	5472
5473	5474	5475	5480	5483	5487	5488	5492	5493	5494	5495	5496	5497
5500	5503	5507	5508	5515	5518	5519	5520	5521	5522	5523	5526	5529
5543	5544	5545	5546	5547	5548	5549	5550	5551	5553	5554	5566	5567
5568	5569	5570	5573	5574	5578	5579	5598	5599	5600	5601	5602	5603
5604	5605	5606	5607	5608	5609	5612	5614	5615	5625	5633	5634	5635
5636	5637	5638	5675	5676	5677	5678	5681	5690	5691	5692	5693	5694
5695	5696	5697	5708	5709	5710	5711	5712	5713	5722	5723	5724	5725
5726	5727	5728	5729	5735	5737	5738	5739	5740	5741	5743	5744	5745
5746	5747	5748	5751	5752	5753	5754	5755	5758	5759	5764	5767	5770
5771	5774	5775	5776	5777	5778	5779	5780	5784	5787	5788	5789	5790
5791	5792	5795	5798	5799	5800	5801	5802	5803	5804	5805	5806	5807
5810	5813	5814	5817	5818	5821	5822	5823	5824	5825	5826	5827	5828
5832	5835	5838	5839	5840	5841	5844	5845	5846	5847	5850	5851	5854
5855	5856	5857	5858	5859	5860	5866	5867	5870	5873	5874	5875	5879
5882	5883	5884	5888	5889	5890	5891	5892	5906	5907	5908	5909	5910
5911	5912	5913	5914	5917	5918	5919	5920	5924	5925	5926	5927	5930
5931	5932	5933	5934	5935	5936	5937	5938	5939	5940	5941	5955	5956
5957	5958	5959	5962	5966	5969	5973	5976	5980	5983	5987	5990	5994
5997	6001	6004	6014	6015	6016	6017	6018	6019	6020	6021	6026	6029
6031	6037	6038	6039	6041	6053	6055	6063	6064	6065	6066	6067	6070
6074	6077	6081	6084	6088	6091	6095	6098	6102	6105	6109	6112	6116
6117	6118	6121	6127	6130	6137	6140	6144	6145	6148	6151	6237	6238
6239	6240	6241	6242	6251	6252	6253	6254	6255	6256	6257	6258	6264
6266	6267	6268	6269	6270	6272	6273	6274	6275	6276	6277	6278	6279
6280	6283	6284	6285	6288	6289	6292	6295	6298	6299	6303	6304	6305
6306	6311	6314	6317	6318	6322	6323	6324	6325	6332	6337	6338	6339
6340	6341	6342	6345	6346	6347	6348	6352	6353	6356	6361	6362	6363
6364	6365	6366	6367	6370	6371	6372	6373	6374	6375	6376	6379	6380
6386	6389	6390	6391	6392	6393	6398	6399	6400	6401	6402	6406	6411
6412	6413	6414	6415	6416	6419	6425	6428	6429	6433	6434	6437	6438
6439	6440	6443	6444	6445	6446	6449	6450	6453	6459	6462	6463	6464
6467	6470	6471	6472	6473	6477	6478	6490	6491	6492	6493	6494	6495
6496	6497	6498	6501	6502	6503	6504	6508	6509	6510	6511	6514	6515
6516	6517	6518	6519	6520	6521	6522	6523	6524	6525	6539	6540	6541
6542	6543	6546	6550	6553	6557	6560	6564	6567	6571	6574	6578	6581
6585	6588	6598	6599	6600	6601	6602	6603	6604	6605	6610	6613	6620
6621	6622	6623	6625	6637	6639	6647	6648	6649	6650	6651	6654	6658
6661	6665	6668	6672	6675	6679	6682	6686	6689	6693	6696	6700	6701
6702	6705	6709	6710	6713	6716	6801	6802	6803	6804	6805	6806	6815
6816	6817	6818	6819	6820	6826	6827	6828	6829	6830	6833	6834	6835
6836	6840	6841	6842	6843	6846	6847	6848	6849	6855	6857	6858	6859
6860	6861	6862	6863	6864	6865	6866	6869	6870	6871	6872	6873	6874

6878	6879	6880	6881	6883	6885	6888	6889	6890	6891	6892	6893	6896
6897	6900	6901	6910	6911	6912	6913	6914	6915	6916	6917	6918	6919
6920	6924	6925	6934	6935	6937	6939	6940	6941	6942	6945	6946	6947
6948	6954	6955	6956	6957	6958	6959	6960	6963	6964	6967	6973	6974
6988	6989	6990	6991	6992	6993	6994	6995	7009	7010	7011	7012	7013
7016	7020	7023	7027	7030	7034	7037	7041	7044	7048	7051	7055	7058
7063	7064	7065	7066	7068	7079	7081	7089	7090	7091	7092	7093	7096
7100	7103	7107	7110	7114	7117	7121	7124	7128	7131	7135	7138	7142
7145	7149	7152	7226	7227	7228	7229	7230	7231	7240	7241	7242	7243
7244	7245	7251	7252	7253	7254	7255	7258	7259	7260	7261	7265	7266
7267	7268	7271	7272	7273	7274	7280	7281	7282	7283	7284	7285	7286
7287	7288	7289	7290	7291	7294	7295	7296	7297	7298	7299	7303	7304
7305	7306	7307	7308	7311	7312	7313	7314	7320	7321	7322	7324	7327
7333	7334	7335	7336	7337	7338	7341	7342	7343	7344	7349	7355	7358
7359	7363	7364	7365	7366	7369	7370	7371	7372	7375	7376	7377	7378
7382	7383	7384	7387	7388	7389	7390	7393	7394	7395	7398	7401	7402
7403	7406	7412	7415	7416	7417	7418	7426	7427	7429	7433	7434	7435
7440	7442	7445	7446	7447	7453	7454	7464	7465	7466	7467	7468	7469
7470	7471	7485	7486	7487	7488	7489	7492	7496	7499	7503	7506	7510
7513	7517	7520	7524	7527	7531	7534	7539	7540	7541	7542	7544	7555
7557	7565	7566	7567	7568	7569	7572	7576	7584	7588	7596	7600	7603
7607	7610	7973	7974	7975	7976	7977	7978	7987	7988	7989	7990	7991
7992	7993	7994	8000	8001	8002	8003	8004	8005	8006	8009	8010	8011
8012	8016	8017	8018	8019	8022	8023	8024	8025	8032	8034	8035	8043
8044	8045	8046	8047	8054	8055	8058	8059	8060	8061	8062	8063	8069
8070	8074	8077	8078	8082	8083	8095	8098	8099	8103	8104	8115	8116
8117	8118	8119	8120	8123	8124	8132	8133	8136	8137	8138	8139	8140
8141	8151	8152	8155	8156	8157	8158	8159	8160	8161	8162	8163	8164
8165	8168	8169	8170	8171	8174	8175	8176	8177	8178	8179	8190	8191
8194	8195	8205	8210	8211	8221	8227	8228	8237	8240	8241	8251	8252
8253	8254	8255	8256	8257	8258	8277	8278	8279	8280	8283	8284	8291
8292	8295	8305	8306	8307	8310	8311	8312	8313	8314	8319	8322	8323
8324	8327	8328	8329	8330	8333	8334	8338	8339	8348	8351	8354	8355
8356	8357	8358	8361	8362	8367	8370	8375	8376	8379	8382	8387	8388
8391	8392	8393	8394	8395	8396	8397	8400	8401	8404	8405	8412	8415
8418	8419	8420	8421	8424	8425	8428	8429	8436	8439	8440	8441	8442
8445	8446	8450	8451	8458	8461	8462	8463	8464	8465	8466	8469	8470
8474	8475	8485	8486	8487	8488	8489	8490	8491	8492	8493	8494	8495
8504	8505	8506	8507	8508	8511	8515	8518	8522	8525	8526	8532	8536
8539	8543	8546	8550	8553	8559	8560	8561	8562	8563	8564	8565	8566
8571	8574	8587	8588	8589	8590	8592	8603	8605	8613	8614	8615	8616
8617	8620	8624	8632	8636	8644	8648	8651	8655	8658	8662	8665	8669
8672	8772	8773	8774	8775	8776	8777	8786	8787	8788	8789	8790	8791
8792	8793	8800	8802	8803	8810	8811	8819	8821	8822	8823	8824	8825
8826	8827	8828	8829	8830	8831	8833	8837	8838	8839	8849	8850	8851
8855	8858	8861	8862	8863	8866	8867	8871	8872	8882	8885	8888	8889
8890	8891	8899	8902	8905	8906	8907	8908	8909	8912	8913	8916	8917
8925	8928	8931	8932	8933	8936	8937	8940	8941	8949	8952	8955	8956
8960	8961	8969	8972	8975	8976	8980	8981	8993	8994	8995	8998	8999
9000	9001	9002	9003	9010	9011	9012	9013	9014	9015	9016	9019	9020
9024	9030	9031	9035	9036	9041	9044	9045	9054	9059	9064	9065	9076
9082	9083	9092	9095	9096	9106	9109	9110	9114	9115	9125	9126	9129
9130	9131	9132	9133	9134	9142	9145	9146	9149	9150	9151	9152	9163
9164	9168	9169	9170	9171	9172	9173	9174	9175	9176	9177	9180	9181
9182	9183	9186	9187	9188	9189	9202	9203	9204	9205	9206	9207	9208
9209	9210	9211	9212	9213	9214	9215	9216	9217	9218	9219	9220	9221

9222	9223	9224	9225	9233	9234	9235	9236	9237	9240	9244	9247	9251
9254	9258	9261	9265	9268	9272	9275	9279	9282	9288	9289	9290	9291
9292	9293	9294	9295	9300	9303	9310	9311	9312	9313	9315	9327	9329
9337	9338	9339	9340	9341	9344	9348	9356	9360	9368	9372	9375	9379
9382	9386	9389	9540	9542	9543	9544	9545	9553	9555	9556	9559	9560
9561	9562	9567	9576	9577	9578	9579	9580	9581	9582	9583	9584	9585
9586	9587	9588	9589	9590	9591	9592	9593	9594	9595	9596	9597	9598
9599	9613	9616	9617	9618	9619	9620	9623	9624	9629	9630	9636	9637
9640	9641	9644	9645	9646	9647	9651	9656	9657	9658	9659	9660	9661
9669	9670	9671	9672	9675	9676	9679	9680	9681	9682	9686	9691	9692
9693	9694	9695	9696	9704	9705	9706	9707	9710	9711	9714	9715	9716
9717	9721	9726	9727	9728	9729	9730	9731	9739	9740	9741	9742	9745
9746	9749	9750	9751	9752	9756	9761	9762	9763	9764	9765	9766	9774
9775	9776	9777	9780	9781	9784	9785	9786	9787	9791	9796	9797	9798
9799	9800	9801	9809	9810	9811	9812	9815	9816	9819	9820	9824	9825
9828	9829	9830	9831	9840	9841	9842	9843	9846	9847	9848	9849	9858
9859	9860	9861	9864	9865	9866	9867	9876	9877	9878	9879	9882	9883
9884	9885	9889	9890	9891	9892	9898	9899	9900	9901	9904	9905	9909
9910	9911	9912	9913	9914	9942	9944	9945	9946	9947	9948	9949	9956
9957	9960	9961	9964	9965	9966	9967	9976	9977	9978	9979	9982	9983
9986	9987	9988	9989	9998	9997	10000	10001	10004	10005	10008	10009	10010
10011	10020	10021	10022	10023	10026	10027	10030	10031	10032	10033	10042	10043
10044	10045	10048	10049	10052	10053	10054	10055	10064	10065	10066	10067	10070
10071	10074	10075	10084	10085	10086	10087	10090	10091	10094	10095	10104	10105
10106	10107	10110	10111	10114	10115	10124	10125	10126	10127	10130	10131	10134
10135	10144	10145	10148	10149	10152	10155	10156	10157	10158	10159	10162	10163
10168	10207	10208	10209	10210	10213	10222	10223	10224	10225	10226	10227	10228
10229	10241	10242	10243	10244	10245	10246	10252	10253	10255	10259	10268	10275
10276	10277	10278	10279	10280	10281	10282	10283	10284	10285	10286	10287	10288
10289	10290	10296	10299	10300	10301	10302	10303	10304	10305	10306	10316	10317
10325	10328	10329	10330	10331	10332	10333	10334	10335	10345	10346	10354	10357
10358	10359	10360	10361	10362	10363	10364	10376	10377	10385	10386	10387	10388
10389	10390	10391	10392	10393	10394	10395	10396	10397	10398	10399	10400	10401
10402	10403	10404	10405	10406	10407	10408	10409	10410	10411	10418	10430	10431
10432	10433	10434	10435	10436	10442	10457	10458	10459	10462	10463	10467	10468
10471	10472	10477	10478	10479	10480	10481	10482	10485	10486	10489	10490	10495
10496	10497	10498	10499	10500	10503	10504	10507	10508	10514	10515	10516	10517
10518	10519	10520	10523	10524	10528	10529	10536	10537	10538	10539	10540	10541
10542	10543	10544	10547	10548	10552	10553	10565	10577	10578	10579	10580	10581
10582	10583	10589	10604	10605	10606	10609	10610	10614	10615	10618	10619	10624
10625	10626	10627	10628	10629	10632	10633	10636	10637	10642	10643	10644	10645
10646	10647	10650	10651	10654	10655	10660	10661	10662	10663	10664	10665	10666
10669	10670	10674	10675	10682	10683	10684	10685	10686	10687	10688	10689	10690
10693	10694	10698	10699	10711	10723	10724	10725	10726	10727	10728	10729	10735
10750	10751	10752	10755	10758	10759	10763	10764	10767	10768	10775	10776	10777
10778	10781	10782	10783	10784	10785	10788	10789	10792	10793	10800	10801	10802
10803	10806	10807	10808	10809	10810	10813	10814	10817	10818	10825	10826	10827
10828	10829	10830	10831	10834	10835	10839	10840	10847	10848	10849	10850	10851
10852	10853	10854	10855	10858	10859	10863	10864	10875	10876	10877	10878	10883
10884	10885	10886	10887	10888	10889	10890	10891	10892	10893	10894	10895	10896
10897	10898	10906	10907	10908	10909	10910	10913	10917	10920	10924	10927	10931
10934	10938	10941	10945	10948	10952	10955	10964	10965	10966	10967	10968	10969
10970	10971	10976	10979	11001	11002	11003	11004	11006	11018	11025	11026	11027
11028	11029	11030	11031	11041	11042	11043	11044	11045	11046	11047	11057	11058
11059	11060	11061	11062	11063	11071	11082	11083	11084	11085	11086	11089	11098
11106	11110	11118	11127	11128	11129	11130	11131	11132	11133	11134	11139	11142

11162	11163	11164	11165	11166	11167	11173	11174	11176	11180	11189	11196	11197
11198	11199	11200	11201	11202	11203	11204	11205	11206	11207	11208	11209	11210
11211	11212	11213	11214	11215	11221	11224	11225	11226	11227	11228	11229	11230
11231	11241	11242	11250	11253	11254	11255	11256	11257	11258	11259	11260	11270
11271	11279	11282	11283	11284	11285	11286	11287	11288	11289	11301	11302	11310
11311	11312	11313	11314	11315	11316	11317	11318	11319	11320	11321	11322	11323
11324	11325	11326	11327	11328	11329	11330	11331	11332	11333	11334	11335	11336
11343	11355	11356	11357	11358	11359	11360	11361	11367	11382	11383	11384	11387
11388	11392	11393	11396	11397	11402	11403	11404	11405	11406	11407	11410	11411
11414	11415	11420	11421	11422	11423	11424	11425	11428	11429	11432	11433	11439
11440	11441	11442	11443	11444	11445	11448	11449	11453	11454	11461	11462	11463
11464	11465	11466	11467	11468	11469	11472	11473	11477	11478	11490	11497	11498
11504	11505	11506	11507	11508	11509	11510	11516	11531	11532	11533	11536	11537
11541	11542	11545	11546	11551	11552	11553	11554	11555	11556	11559	11560	11563
11564	11569	11570	11571	11572	11573	11574	11577	11578	11581	11582	11587	11588
11589	11590	11591	11592	11593	11596	11597	11601	11602	11609	11610	11611	11612
11613	11614	11615	11616	11617	11620	11621	11625	11626	11638	11650	11651	11652
11653	11654	11655	11656	11662	11677	11678	11679	11682	11685	11686	11690	11691
11694	11695	11702	11703	11704	11705	11708	11709	11710	11711	11712	11715	11716
11719	11720	11727	11728	11729	11730	11733	11734	11735	11736	11737	11740	11741
11744	11745	11752	11753	11754	11755	11756	11757	11758	11761	11762	11766	11767
11774	11775	11776	11777	11778	11779	11780	11781	11782	11785	11786	11790	11791
11802	11803	11804	11805	11810	11811	11812	11813	11814	11815	11816	11817	11818
11819	11820	11821	11822	11823	11824	11825	11833	11834	11835	11836	11837	11840
11844	11847	11851	11854	11858	11861	11865	11868	11872	11875	11879	11882	11892
11893	11894	11895	11896	11897	11898	11899	11904	11907	11929	11930	11931	11932
11934	11946	11953	11954	11955	11956	11957	11958	11959	11969	11970	11971	11972
11973	11974	11975	11985	11986	11987	11988	11989	11990	11991	11999	12010	12011
12012	12013	12014	12017	12026	12034	12038	12046	12055	12056	12057	12058	12059
12060	12061	12062	12067	12070	12090	12091	12092	12093	12094	12095	12099	12101
12102	12103	12104	12105	12106	12112	12113	12114	12115	12116	12117	12118	12119
12120	12121	12127	12128	12129	12130	12131	12132	12133	12134	12135	12137	12141
12142	12143	12147	12150	12151	12154	12155	12156	12157	12158	12159	12160	12161
12162	12164	12168	12169	12170	12174	12175	12182	12187	12188	12192	12197	12198
12202	12207	12208	12215	12220	12223	12226	12227	12233	12234	12235	12240	12241
12245	12250	12251	12256	12264	12265	12266	12267	12268	12269	12270	12271	12276
12277	12281	12284	12289	12290	12294	12299	12300	12308	12309	12332	12335	12336
12340	12341	12342	12343	12344	12345	12346	12348	12351	12352	12353	12354	12357
12358	12362	12363	12366	12367	12368	12369	12372	12375	12378	12379	12380	12381
12384	12395	12389	12390	12393	12394	12395	12396	12401	12404	12407	12408	12409
12412	12413	12417	12418	12421	12422	12423	12424	12429	12432	12433	12434	12435
12438	12439	12443	12444	12449	12452	12453	12454	12457	12458	12462	12463	12466
12467	12468	12469	12470	12473	12474	12478	12481	12482	12486	12487	12488	12489
12490	12491	12492	12494	12497	12498	12499	12500	12503	12504	12508	12509	12512
12513	12514	12515	12518	12521	12522	12523	12524	12527	12528	12532	12533	12536
12537	12538	12539	12542	12545	12546	12547	12550	12551	12555	12556	12559	12560
12561	12562	12565	12568	12569	12570	12571	12574	12575	12579	12580	12585	12588
12589	12590	12593	12594	12598	12599	12602	12603	12604	12605	12606	12614	12615
12616	12617	12618	12619	12620	12621	12622	12623	12624	12625	12626	12627	12635
12636	12637	12638	12639	12642	12646	12649	12653	12656	12660	12663	12667	12670
12674	12677	12681	12684	12688	12689	12691	12698	12699	12701	12709	12710	12711
12712	12714	12727	12728	12729	12730	12731	12732	12733	12734	12735	12736	12737
12738	12745	12746	12747	12748	12749	12752	12756	12764	12768	12776	12808	12809
12810	12811	12812	12813	12814	12815	12819	12820	12821	12822	12828	12829	12830
12831	12832	12833	12834	12835	12836	12843	12844	12845	12846	12847	12848	12851
12852	12853	12854	12865	12866	12867	12870	12871	12874	12875	12879	12884	12885

CJKDHAO REF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 408
CROSS REFERENCE TABLE -- USER SYMBOLS

12891	12894	12895	12896	12897	12899	12903	12904	12905	12911	12917	12912	12925
12928	12929	12930	12931	12932	12933	12934	12935	12936	12937	12938	12941	12944
12945	12949	12950	12955	12958	12959	12962	12963	12968	12969	12970	12975	12976
12979	12980	12981	12982	12986	12991	12992	13000	13003	13004	13005	13006	13009
13010	13013	13016	13017	13018	13019	13020	13021	13023	13027	13028	13029	13036
13037	13038	13039	13040	13042	13046	13047	13048	13052	13053	13054	13055	13060
13061	13062	13063	13073	13076	13077	13083	13084	13085	13086	13087	13088	13089
13090	13105	13106	13109	13118	13121	13122	13123	13124	13125	13126	13127	13131
13134	13135	13136	13137	13138	13139	13140	13144	13147	13148	13149	13152	13153
13154	13155	13158	13159	13163	13164	13169	13170	13171	13174	13175	13176	13177
13185	13186	13190	13191	13194	13201	13202	13209	13212	13213	13214	13217	13218
13219	13220	13221	13224	13227	13228	13229	13230	13231	13236	13239	13242	13247
13248	13251	13252	13253	13254	13257	13258	13262	13263	13268	13271	13272	13273
13274	13282	13283	13287	13288	13291	13298	13299	13302	13303	13304	13305	13306
13307	13312	13315	13318	13321	13322	13326	13327	13332	13335	13343	13344	13348
13349	13352	13359	13360	13363	13364	13365	13366	13367	13368	13373	13379	13380
13381	13382	13385	13386	13390	13391	13397	13398	13399	13400	13406	13407	13411
13412	13415	13422	13423	13426	13429	13430	13435	13438	13439	13440	13441	13442
13443	13444	13445	13446	13447	13451	13454	13455	13459	13460	13463	13469	13470
13474	13475	13478	13485	13486	13489	13492	13493	13503	13504	13505	13506	13507
13508	13509	13510	13511	13512	13513	13514	13515	13516	13524	13525	13526	13527
13528	13531	13535	13543	13547	13548	13549	13550	13553	13557	13560	13565	13566
13567	13568	13569	13570	13571	13572	13573	13574	13579	13582	13583	13586	13587
13591	13594	13604	13605	13607	13611	13612	13613	13617	13618	13620	13623	13624
13625	13626	13630	13643	13646	13647	13648	13649	13650	13651	13652	13653	13654
13655	13656	13657	13658	13659	13660	13661	13662	13663	13664	13665	13666	13667
13668	13669	13678	13679	13680	13681	13682	13685	13689	13697	13701	13709	13714
13715	13716	13719	13723	13726	13730	13731	13732	13733	13738	13741	13743	13744
13797	13798	13799	13800	13803	13812	13813	13814	13815	13816	13817	13818	13819
13831	13832	13833	13834	13835	13836	13842	13843	13845	13849	13858	13865	13866
13867	13868	13869	13870	13871	13872	13873	13874	13875	13876	13877	13878	13879
13880	13886	13887	13888	13898	13908	13923	13924	13925	13926	13927	13928	13929
13930	13931	13932	13933	13934	13935	13936	13937	13944	13945	13946	13947	13955
13956	13957	13958	13959	13962	13966	13969	13973	13976	13980	13983	13987	13990
13994	13997	14001	14004	14013	14014	14015	14016	14017	14018	14019	14020	14025
14028	14035	14036	14037	14038	14040	14052	14059	14060	14061	14062	14063	14064
14065	14066	14067	14068	14069	14070	14071	14072	14078	14079	14080	14090	14100
14111	14123	14124	14125	14126	14127	14130	14139	14147	14151	14159	14168	14169
14170	14171	14172	14173	14174	14177	14178	14179	14180	14181	14182	14183	14184
14189	14192	14212	14213	14214	14215	14216	14217	14223	14224	14226	14230	14239
14246	14247	14248	14249	14250	14251	14252	14253	14254	14255	14256	14257	14258
14259	14260	14261	14262	14263	14264	14265	14271	14272	14273	14283	14293	14309
14310	14311	14312	14313	14314	14315	14316	14317	14318	14319	14320	14321	14322
14323	14330	14331	14332	14333	14341	14342	14343	14344	14345	14348	14352	14355
14359	14362	14366	14369	14373	14376	14380	14383	14387	14390	14400	14401	14402
14403	14404	14405	14406	14407	14412	14415	14422	14423	14424	14425	14427	14439
14446	14447	14448	14449	14450	14451	14452	14453	14454	14455	14456	14457	14458
14459	14460	14461	14462	14463	14469	14470	14471	14481	14491	14501	14511	14512
14513	14514	14515	14518	14527	14535	14539	14547	14556	14557	14558	14559	14560
14561	14562	14565	14566	14567	14568	14569	14570	14571	14572	14577	14580	14600
14601	14602	14603	14604	14605	14609	14611	14612	14613	14614	14615	14616	14622
14623	14624	14625	14626	14627	14628	14629	14630	14631	14638	14639	14640	14641
14642	14643	14644	14645	14646	14648	14652	14653	14654	14658	14659	14660	14661
14662	14663	14664	14665	14666	14667	14668	14670	14674	14675	14676	14680	14681
14688	14693	14694	14698	14703	14704	14708	14713	14714	14721	14726	14729	14732
14733	14739	14740	14741	14746	14747	14751	14756	14757	14762	14770	14771	14772

14773	14774	14775	14776	14777	14782	14783	14787	14790	14795	14796	14800	14805
14806	14814	14815	14832	14835	14836	14837	14838	14839	14840	14841	14842	14845
14848	14849	14853	14854	14860	14863	14864	14865	14866	14869	14870	14873	14874
14877	14878	14879	14880	14881	14882	14886	14887	14890	14893	14894	14898	14899
14902	14903	14907	14908	14909	14910	14911	14912	14913	14914	14917	14920	14921
14925	14926	14932	14935	14936	14937	14938	14941	14942	14945	14946	14949	14950
14951	14952	14953	14954	14958	14959	14962	14965	14966	14970	14971	14974	14975
14983	14984	14985	14986	14987	14988	14989	14990	14991	14992	14993	14994	14995
14996	15004	15005	15006	15007	15008	15011	15015	15023	15027	15035	15039	15040
15041	15042	15044	15058	15059	15060	15061	15062	15063	15064	15065	15066	15067
15068	15069	15076	15077	15078	15079	15080	15083	15087	15095	15099	15107	15139
15140	15141	15142	15143	15144	15145	15146	15150	15152	15153	15159	15160	15161
15162	15163	15164	15165	15166	15174	15175	15176	15177	15178	15181	15184	15185
15189	15190	15196	15199	15200	15201	15202	15205	15206	15209	15210	15211	15212
15216	15219	15220	15224	15225	15237	15238	15239	15242	15243	15246	15247	15251
15256	15257	15263	15264	15267	15268	15269	15275	15276	15280	15281	15287	15293
15294	15297	15307	15308	15309	15310	15311	15314	15315	15316	15317	15318	15319
15320	15323	15326	15327	15331	15332	15333	15334	15335	15336	15339	15340	15341
15342	15347	15350	15351	15352	15353	15354	15355	15358	15359	15360	15361	15364
15365	15370	15373	15374	15375	15380	15381	15384	15385	15389	15390	15393	15394
15395	15396	15400	15405	15406	15411	15422	15425	15426	15427	15428	15431	15432
15435	15438	15439	15440	15441	15445	15446	15447	15450	15451	15454	15455	15456
15457	15458	15459	15460	15461	15462	15463	15464	15465	15470	15473	15474	15475
15476	15480	15481	15484	15485	15486	15487	15488	15489	15506	15507	15508	15509
15510	15511	15512	15513	15514	15515	15516	15517	15518	15519	15523	15524	15525
15526	15527	15530	15534	15540	15544	15550	15554	15555	15556	15557	15560	15564
15567	15573	15574	15575	15576	15577	15578	15579	15580	15581	15582	15587	15590
15591	15594	15595	15599	15602	15606	15607	15609	15612	15613	15614	15615	15619
15633	15635	15636	15637	15638	15639	15640	15641	15651	15652	15653	15654	15655
15658	15662	15670	15674	15682	15687	15688	15689	15692	15696	15699	15703	15704
15705	15706	15711	15714	15715	15716	15769	15770	15771	15772	15775	15784	15785
15786	15787	15788	15789	15790	15791	15804	15805	15806	15807	15808	15809	15815
15816	15818	15822	15831	15838	15839	15840	15841	15842	15843	15844	15845	15846
15847	15848	15849	15850	15851	15852	15853	15859	15860	15861	15871	15881	15896
15897	15898	15899	15900	15901	15902	15903	15904	15905	15906	15907	15908	15909
15910	15917	15918	15919	15920	15928	15929	15930	15931	15932	15935	15939	15942
15946	15949	15953	15956	15960	15963	15967	15970	15974	15977	15987	15988	15989
15990	15991	15992	15993	15994	15999	16002	16009	16010	16011	16012	16014	16026
16033	16034	16035	16036	16037	16038	16039	16040	16041	16042	16043	16044	16045
16046	16052	16053	16054	16064	16074	16084	16095	16096	16097	16098	16099	16102
16111	16119	16123	16131	16140	16141	16142	16143	16144	16145	16146	16149	16150
16151	16152	16153	16154	16155	16156	16161	16164	16202	16203	16204	16205	16208
16217	16218	16219	16220	16221	16222	16223	16224	16237	16238	16239	16240	16241
16242	16248	16249	16251	16255	16264	16271	16272	16273	16274	16275	16276	16277
16278	16279	16280	16281	16282	16283	16284	16285	16286	16287	16288	16289	16290
16296	16297	16298	16308	16318	16334	16335	16336	16337	16338	16339	16340	16341
16342	16343	16344	16345	16346	16347	16348	16355	16356	16357	16358	16366	16367
16368	16369	16370	16373	16377	16380	16384	16387	16391	16394	16398	16401	16405
16408	16412	16415	16425	16426	16427	16428	16429	16430	16431	16432	16437	16440
16447	16448	16449	16450	16452	16464	16471	16472	16473	16474	16475	16476	16477
16478	16479	16480	16481	16482	16483	16484	16485	16486	16487	16488	16494	16495
16496	16506	16516	16526	16537	16538	16539	16540	16541	16544	16553	16561	16565
16573	16582	16583	16584	16585	16586	16587	16588	16591	16592	16593	16594	16595
16596	16597	16598	16603	16606	16606	16859	16864	16865	16866	16867	16979	16980
16982	16983	16984	16985	16986	16986	16987	16988	16989	16990	17008	17009	17010
17012	17013	17014	17015	17018	17019	17020	17021	17025	17026	17027	17028	17045

17046	17047	17048	17049	17050	17051	17052	17055	17056	17057	17058	17062	17063
17064	17065	17082	17083	17084	17085	17086	17087	17088	17089	17092	17093	17094
17095	17099	17100	17101	17102	17117	17118	17119	17120	17121	17122	17123	17124
17127	17128	17129	17130	17134	17135	17136	17137	17155	17156	17157	17158	17159
17160	17161	17162	17165	17166	17167	17168	17172	17173	17174	17175	17191	17192
17193	17194	17195	17196	17197	17198	17199	17200	17201	17202	17218	17219	17220
17221	17222	17223	17224	17225	17226	17227	17228	17229	17254	17255	17262	17265
17266	17270	17273	17274	17284	17287	17288	17289	17290	17291	17292	17293	17294
17295	17296	17297	17298	17299	17300	17301	17302	17304	17305	17306	17307	17308
17309	17310	17311	17312	17313	17314	17315	17316	17317	17318	17319	17327	17330
17331	17338	17339	17390	17391	17392	17393	17394	17395	17396	17402	17405	17406
17410	17411	17422	17425	17426	17430	17433	17434	17435	17436	17439	17440	17443
17444	17447	17448	17449	17450	17451	17452	17456	17457	17460	17461	17462	17463
720#	1267	1270	1275	1277	1312	1321	1330	1343	1346	1351	1353	1356
1361	1363	1365	1370	1372	1377	1379	1384	1386	1391	1393	1398	1400
1403	1408	1410	1413	1418	1420	1448	1453	1455	1460	1462	1467	1469
1474	1476	1481	1483	1488	1532	1541	1550	1563	1566	1571	1573	1576
1581	1583	1585	1590	1592	1597	1599	1604	1606	1611	1613	1618	1620
1625	1651	1656	1658	1663	1665	1670	1672	1677	1679	1684	1686	1691
1693	1698	1760	1765	1770	1778	1780	1786	1792	1806	1811	1814	1819
1824	1826	1830	1835	1838	1843	1848	1850	1852	1857	1862	1880	1885
1887	1892	1894	1899	1901	1906	1908	1913	1915	1920	1922	1927	1940
1946	1951	1956	1963	1969	1974	1979	1994	1999	2009	2013	2016	2021
2026	2036	2041	2049	2056	2061	2099	2101	2105	2107	2109	2114	2116
2121	2123	2128	2130	2135	2137	2142	2144	2149	2151	2156	2158	2163
2172	2177	2376	2381	2386	2394	2396	2402	2408	2422	2427	2430	2435
2440	2442	2446	2451	2454	2459	2464	2466	2486	2491	2493	2498	2500
2505	2507	2512	2514	2519	2521	2526	2528	2533	2544	2554	2559	2564
2571	2576	2581	2603	2608	2613	2618	2620	2628	2633	2644	2651	2656
2694	2696	2700	2702	2704	2709	2711	2716	2718	2723	2725	2730	2732
2737	2739	2744	2746	2751	2753	2758	2766	2771	2889	2891	2897	2903
2917	2922	2925	2930	2935	2937	2941	2946	2949	2954	2959	2961	3007
3012	3014	3019	3021	3026	3028	3033	3035	3040	3042	3047	3049	3054
3071	3077	3082	3087	3094	3100	3105	3111	3126	3131	3141	3143	3155
3160	3168	3175	3180	3220	3222	3226	3228	3230	3235	3237	3242	3244
3249	3251	3256	3258	3263	3265	3270	3272	3277	3279	3284	3293	3298
3396	3404	3413	3415	3423	3432	3473	3478	3480	3485	3487	3492	3494
3499	3501	3506	3508	3513	3515	3520	3527	3532	3561	3565	3569	3574
3576	3581	3583	3588	3590	3595	3597	3602	3604	3609	3682	3690	3699
3701	3709	3718	3759	3764	3766	3771	3773	3778	3780	3785	3787	3792
3794	3799	3801	3806	3813	3818	3847	3851	3856	3861	3863	3868	3870
3875	3877	3882	3884	3889	3891	3896	3966	3972	3979	3981	3989	3998
4055	4060	4062	4067	4069	4074	4076	4081	4083	4088	4090	4095	4097
4102	4109	4114	4159	4163	4167	4172	4174	4179	4181	4186	4188	4193
4195	4200	4202	4207	4279	4285	4292	4294	4302	4311	4365	4370	4372
4377	4379	4384	4386	4391	4393	4398	4400	4405	4407	4412	4419	4424
4466	4470	4474	4479	4481	4486	4488	4493	4495	4500	4502	4507	4509
4514	4590	4594	4599	4614	4622	4630	4655	4663	4665	4673	4675	4677
4682	4700	4705	4707	4712	4714	4719	4721	4726	4728	4733	4735	4740
4742	4747	4754	4759	4794	4799	4801	4806	4808	4813	4815	4820	4822
4827	4829	4834	4836	4841	4934	4940	4947	4962	4979	4998	5004	5012
5021	5023	5047	5052	5054	5059	5061	5066	5068	5073	5075	5080	5082
5087	5089	5094	5101	5106	5143	5147	5149	5154	5156	5161	5163	5168
5170	5175	5177	5182	5184	5189	5191	5196	5238	5241	5246	5248	5392
5397	5408	5413	5415	5420	5422	5427	5429	5434	5436	5441	5443	5448
5450	5455	5475	5480	5485	5500	5505	5515	5526	5531	5539	5541	5570

SF\$IF = 000110

5575	5580	5625	5631	5678	5681	5686	5688	5748	5755	5760	5762	5764
5767	5781	5784	5793	5795	5808	5810	5829	5832	5835	5842	5864	5870
5876	5879	5885	5893	5895	5897	5899	5914	5921	5928	5959	5964	5966
5971	5973	5978	5980	5985	5987	5992	5994	5999	6001	6006	6021	6026
6031	6041	6046	6067	6072	6074	6079	6081	6086	6088	6093	6095	6100
6102	6107	6109	6114	6118	6123	6127	6132	6137	6142	6148	6153	6280
6285	6290	6292	6295	6300	6307	6309	6311	6314	6319	6326	6328	6330
6332	6334	6354	6356	6358	6367	6384	6386	6394	6406	6408	6417	6419
6421	6425	6430	6435	6440	6447	6451	6453	6455	6459	6465	6467	6474
6479	6481	6483	6498	6505	6512	6543	6548	6550	6555	6557	6562	6564
6569	6571	6576	6578	65	6585	6590	6605	6610	6615	6625	6630	6651
6656	6658	6663	6665	6670	6672	6677	6679	6684	6686	6691	6693	6698
6702	6707	6713	6718	6830	6837	6844	6866	6875	6885	6894	6942	6948
6950	6960	6965	6967	6969	6975	6977	6979	6981	7013	7018	7020	7025
7027	7032	7034	7039	7041	7046	7048	7053	7055	7060	7068	7073	7093
7098	7100	7105	7107	7112	7114	7119	7121	7126	7128	7133	7135	7140
7142	7147	7149	7154	7255	7262	7269	7291	7300	7318	7324	7327	7329
7338	7345	7347	7349	7351	7355	7360	7367	7384	7391	7396	7398	7404
7406	7408	7412	7419	7421	7423	7442	7447	7449	7455	7457	7489	7494
7496	7501	7503	7508	7510	7515	7517	7522	7524	7529	7531	7536	7544
7549	7569	7574	7576	7578	7580	7586	7588	7590	7592	7598	7600	7605
7607	7612	8006	8013	8020	8063	8065	8071	8074	8079	8084	8086	8095
8100	8105	8165	8180	8205	8207	8212	8221	8223	8229	8237	8242	8295
8300	8307	8315	8319	8324	8330	8335	8340	8342	8344	8348	8351	8359
8363	8367	8370	8372	8377	8379	8382	8384	8389	8397	8402	8406	8408
8412	8415	8421	8426	8430	8432	8436	8442	8447	8452	8454	8458	8466
8471	8476	8478	8508	8513	8515	8520	8522	8527	8529	8534	8536	8541
8543	8548	8550	8555	8566	8571	8576	8592	8597	8617	8622	8624	8626
8628	8634	8636	8638	8640	8646	8648	8653	8655	8660	8662	8667	8669
8674	8855	8858	8863	8868	8873	8875	8877	8882	8885	8892	8894	8899
8902	8909	8914	8918	8920	8925	8928	8933	8938	8942	8944	8949	8952
8957	8962	8964	8969	8972	8977	8982	8984	9016	9021	9024	9026	9032
9037	9039	9041	9046	9054	9056	9059	9061	9066	9068	9076	9078	9084
9092	9097	9106	9111	9116	9142	9153	9177	9190	9237	9242	9244	9249
9251	9256	9258	9263	9265	9270	9272	9277	9279	9284	9295	9300	9305
9315	9320	9341	9346	9348	9350	9352	9358	9360	9362	9364	9370	9372
9377	9379	9384	9386	9391	9553	9557	9562	9564	9567	9573	9603	9605
9613	9620	9625	9627	9651	9653	9662	9686	9688	9697	9721	9723	9732
9756	9758	9767	9791	9793	9802	10152	10159	10164	10166	10210	10213	10218
10220	10296	10313	10318	10325	10342	10347	10354	10373	10378	10418	10423	10459
10464	10469	10482	10487	10500	10505	10520	10525	10530	10544	10549	10554	10565
10570	10606	10611	10616	10629	10634	10647	10652	10666	10671	10676	10690	10695
10700	10711	10716	10752	10755	10760	10765	10769	10778	10785	10790	10794	10803
10810	10815	10819	10831	10836	10841	10855	10860	10865	10910	10915	10917	10922
10924	10929	10931	10936	10938	10943	10945	10950	10952	10957	10971	10976	10981
11006	11011	11086	11091	11098	11100	11102	11108	11110	11112	11114	11120	11134
11139	11144	11221	11238	11243	11250	11267	11272	11279	11298	11303	11343	11348
11384	11389	11394	11407	11412	11425	11430	11445	11450	11455	11469	11474	11479
11490	11495	11533	11538	11543	11556	11561	11574	11579	11593	11598	11603	11617
11622	11627	11638	11643	11679	11682	11687	11692	11696	11705	11712	11717	11721
11730	11737	11742	11746	11758	11763	11768	11782	11787	11792	11837	11842	11844
11849	11851	11856	11858	11863	11865	11870	11872	11877	11879	11884	11899	11904
11909	11934	11939	12014	12019	12026	12028	12030	12036	12038	12040	12042	12048
12062	12067	12072	12147	12152	12182	12184	12189	12192	12194	12199	12202	12204
12209	12211	12213	12215	12217	12220	12223	12228	12230	12235	12237	12242	12245
12247	12252	12254	12256	12271	12281	12284	12286	12291	12294	12296	12301	12303

12305	12310	12312	12314	12316	12332	12337	12348	12354	12359	12364	12370	12372	
12375	12381	12386	12391	12397	12399	12401	12404	12409	12414	12419	12425	12427	
12429	12435	12440	12445	12447	12449	12454	12459	12464	12471	12478	12483	12494	
12500	12505	12510	12516	12518	12524	12529	12534	12540	12542	12547	12552	12557	
12563	12565	12571	12576	12581	12583	12585	12590	12595	12600	12607	12639	12644	
12646	12651	12653	12658	12660	12665	12667	12672	12674	12679	12681	12686	12691	
12696	12701	12706	12714	12719	12749	12754	12756	12758	12760	12766	12768	12770	
12772	12778	12867	12872	12879	12881	12886	12891	12909	12911	12913	12919	12925	
12938	12941	12946	12951	12953	12955	12960	12963	12970	12972	12977	12986	12988	
12993	12995	13000	13006	13011	13013	13033	13055	13067	13069	13071	13073	13078	
13109	13114	13118	13128	13131	13141	13144	13149	13155	13160	13165	13167	13171	
13177	13179	13181	13187	13192	13194	13196	13198	13203	13205	13207	13209	13214	
13222	13224	13232	13234	13236	13239	13242	13244	13249	13254	13259	13264	13266	
13268	13274	13276	13278	13284	13289	13291	13293	13295	13300	13308	13310	13312	
13315	13318	13323	13328	13330	13332	13335	13337	13339	13345	13350	13352	13354	
13356	13361	13369	13371	13373	13382	13387	13392	13400	13402	13408	13413	13415	
13417	13419	13424	13426	13431	13433	13435	13451	13456	13461	13463	13465	13471	
13476	13478	13480	13482	13487	13489	13494	13496	13528	13533	13535	13537	13539	
13545	13550	13555	13557	13562	13574	13579	13584	13591	13596	13620	13627	13630	
13635	13682	13687	13689	13691	13693	13699	13701	13703	13705	13711	13716	13721	
13723	13728	13733	13738	13745	13800	13803	13808	13810	13959	13964	13966	13971	
13973	13978	13980	13985	13987	13992	13994	13999	14001	14006	14020	14025	14030	
14040	14045	14127	14132	14139	14141	14143	14149	14151	14153	14155	14161	14184	
14189	14194	14345	14350	14352	14357	14359	14364	14366	14371	14373	14378	14380	
14385	14387	14392	14407	14412	14417	14427	14432	14515	14520	14527	14529	14531	
14537	14539	14541	14543	14549	14572	14577	14582	14688	14690	14695	14698	14700	
14705	14708	14710	14715	14717	14719	14721	14723	14726	14729	14734	14736	14741	
14743	14748	14751	14753	14758	14760	14762	14777	14787	14790	14792	14797	14800	
14802	14807	14809	14811	14816	14818	14820	14822	14842	14845	14850	14855	14857	
14860	14867	14890	14895	14900	14904	14914	14917	14922	14927	14929	14932	14939	
14962	14967	14972	14976	15008	15013	15015	15017	15019	15025	15027	15029	15031	
15037	15044	15049	15080	15085	15087	15089	15091	15097	15099	15101	15103	15109	
15178	15181	15186	15191	15193	15196	15203	15216	15221	15226	15228	15239	15244	
15251	15253	15258	15269	15271	15277	15282	15287	15289	15295	15297	15299	15301	
15320	15323	15328	15343	15345	15347	15362	15365	15370	15375	15377	15382	15386	
15391	15400	15402	15407	15409	15411	15413	15415	15422	15428	15433	15435	15442	
15465	15470	15477	15482	15493	15495	15497	15527	15532	15534	15536	15542	15544	
15546	15552	15557	15562	15564	15569	15582	15587	15592	15599	15604	15609	15616	
15619	15624	15655	15660	15662	15664	15666	15672	15674	15676	15678	15684	15689	
15694	15696	15701	15706	15711	15718	15772	15775	15780	15782	15932	15937	15939	
15944	15946	15951	15953	15958	15960	15965	15967	15972	15974	15979	15994	15999	
16004	16014	16019	16099	16104	16111	16113	16115	16121	16123	16125	16127	16133	
16156	16161	16166	16205	16208	16213	16215	16370	16375	16377	16382	16384	16389	
16391	16396	16398	16403	16405	16410	16412	16417	16432	16437	16442	16452	16457	
16541	16546	16553	16555	16557	16563	16565	16567	16569	16575	16598	16603	16608	
16859	16861	16870	17262	17267	17270	17275	17277	17284	17320	17327	17335	17340	
17402	17407	17412	17422	17427	17430	17437	17464						
\$F\$INC= 000210	720#	7426	7436	8830	8840	10252	10262	11173	11183	12896	12906	13020	13030
	13039	13049	13604	13614	13842	13852	14223	14233	15815	15825	16248	16258	
\$F\$L00= 000200	720#												
\$F\$NAM= 000160	720#												
\$F\$NO = 000403	720#	1264	1266	1267	1270	1280	1281	1282	1283	1284	1285	1298	1299
	1300	1301	1312	1314	1315	1316	1324	1325	1332	1333	1334	1335	1336
	1337	1343	1345	1346	1356	1358	1365	1367	1372	1374	1379	1381	1386
	1388	1393	1400	1402	1403	1413	1415	1422	1423	1432	1433	1434	1435
	1448	1450	1455	1457	1462	1464	1469	1471	1476	1483	1518	1519	1520

1521	1532	1534	1535	1536	1544	1545	1552	1553	1554	1555	1556	1557
1563	1565	1566	1576	1578	1585	1587	1592	1594	1599	1601	1606	1608
1613	1615	1620	1622	1627	1628	1651	1653	1658	1660	1665	1667	1672
1674	1679	1681	1686	1688	1693	1695	1713	1714	1715	1716	1717	1718
1727	1728	1729	1730	1731	1732	1733	1734	1735	1736	1737	1738	1739
1740	1741	1742	1750	1751	1752	1753	1754	1755	1756	1757	1758	1759
1760	1763	1764	1768	1769	1774	1775	1776	1777	1778	1780	1784	1785
1789	1791	1794	1795	1796	1797	1798	1799	1800	1801	1802	1803	1804
1805	1806	1814	1816	1828	1829	1830	1838	1840	1852	1855	1856	1860
1861	1876	1877	1878	1879	1880	1882	1887	1889	1894	1896	1901	1903
1908	1910	1915	1917	1922	1924	1934	1935	1936	1937	1938	1939	1940
1942	1943	1944	1946	1948	1953	1958	1959	1966	1967	1969	1971	1976
1981	1982	1986	1988	1989	1991	1996	2001	2002	2003	2011	2012	2013
2015	2016	2018	2019	2020	2024	2025	2028	2030	2031	2033	2038	2043
2051	2052	2053	2054	2056	2069	2070	2073	2074	2077	2078	2079	2080
2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2100	2102	2109
2111	2116	2118	2123	2125	2130	2132	2137	2139	2144	2146	2151	2153
2158	2160	2165	2166	2167	2169	2172	2179	2330	2331	2332	2333	2334
2335	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355
2356	2357	2358	2359	2366	2367	2368	2369	2370	2371	2372	2373	2374
2375	2376	2379	2380	2384	2385	2390	2391	2392	2393	2394	2396	2400
2401	2405	2407	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419
2420	2421	2422	2430	2432	2444	2445	2446	2454	2456	2468	2469	2482
2483	2484	2485	2486	2488	2493	2495	2500	2502	2507	2509	2514	2516
2521	2523	2528	2530	2540	2541	2542	2543	2544	2546	2547	2548	2549
2551	2552	2554	2556	2561	2566	2567	2578	2583	2585	2586	2587	2593
2594	2595	2597	2598	2599	2601	2603	2605	2610	2620	2622	2623	2625
2630	2635	2637	2638	2646	2647	2648	2649	2651	2664	2665	2668	2669
2672	2673	2674	2675	2676	2677	2678	2679	2680	2681	2682	2683	2684
2685	2695	2697	2704	2706	2711	2713	2718	2720	2725	2727	2732	2734
2739	2741	2746	2748	2753	2755	2760	2761	2762	2763	2766	2773	2834
2835	2836	2837	2838	2839	2848	2849	2850	2851	2852	2853	2854	2855
2856	2857	2858	2859	2860	2861	2862	2863	2871	2872	2873	2874	2875
2876	2877	2878	2879	2880	2881	2882	2885	2886	2887	2888	2889	2891
2895	2896	2900	2902	2905	2906	2907	2908	2909	2910	2911	2912	2913
2914	2915	2916	2917	2925	2927	2939	2940	2941	2949	2951	2963	2964
2970	2974	2978	2979	2983	2984	2985	2986	2987	2988	2989	2990	3003
3004	3005	3006	3007	3009	3014	3016	3021	3023	3028	3030	3035	3037
3042	3044	3049	3051	3061	3062	3063	3064	3065	3066	3067	3068	3069
3070	3071	3073	3074	3075	3077	3079	3084	3089	3090	3097	3098	3100
3102	3107	3113	3114	3118	3120	3121	3123	3128	3133	3134	3135	3143
3145	3146	3148	3149	3150	3152	3157	3162	3170	3171	3172	3173	3175
3188	3189	3192	3193	3196	3197	3198	3199	3200	3201	3202	3203	3204
3205	3206	3207	3208	3209	3221	3223	3230	3232	3237	3239	3244	3246
3251	3253	3258	3260	3265	3267	3272	3274	3279	3281	3286	3288	3289
3290	3293	3295	3300	3361	3362	3363	3364	3365	3366	3374	3375	3376
3377	3378	3379	3380	3381	3388	3389	3390	3391	3392	3393	3397	3398
3399	3400	3401	3402	3403	3407	3408	3409	3410	3411	3412	3415	3417
3418	3419	3420	3422	3426	3428	3429	3431	3434	3435	3436	3437	3438
3439	3440	3441	3444	3445	3446	3447	3448	3449	3452	3454	3455	3456
3469	3470	3471	3472	3473	3475	3480	3482	3487	3489	3494	3496	3501
3503	3508	3510	3515	3517	3522	3523	3524	3525	3527	3540	3541	3546
3548	3549	3550	3557	3558	3562	3567	3568	3569	3571	3576	3578	3583
3590	3597	3604	3645	3646	3647	3648	3649	3650	3659	3660	3661	3662
3663	3664	3665	3666	3674	3675	3676	3677	3678	3679	3683	3684	3685
3686	3687	3688	3689	3693	3694	3695	3696	3697	3698	3701	3703	3704

3705	3706	3708	3712	3714	3715	3717	3720	3721	3722	3723	3724	3725
3726	3727	3730	3731	3732	3733	3734	3735	3738	3740	3741	3742	3755
3756	3757	3758	3759	3761	3766	3768	3773	3775	3780	3782	3787	3789
3794	3796	3801	3803	3808	3809	3810	3811	3813	3826	3827	3832	3834
3835	3836	3843	3844	3848	3853	3855	3856	3858	3863	3865	3870	3877
3884	3891	3929	3930	3931	3932	3933	3934	3943	3944	3945	3946	3947
3948	3956	3957	3958	3959	3960	3961	3962	3963	3967	3968	3969	3970
3971	3975	3976	3977	3978	3981	3983	3984	3985	3986	3988	3992	3994
3995	3997	4000	4001	4002	4003	4004	4005	4006	4007	4010	4011	4012
4013	4014	4015	4018	4020	4021	4022	4024	4026	4032	4033	4034	4036
4037	4038	4051	4052	4053	4054	4055	4057	4062	4064	4069	4071	4076
4078	4083	4085	4090	4092	4097	4099	4104	4105	4106	4107	4109	4122
4123	4126	4128	4129	4130	4131	4132	4134	4136	4142	4144	4145	4146
4155	4156	4160	4165	4166	4167	4169	4174	4176	4181	4188	4195	4202
4242	4243	4244	4245	4246	4247	4256	4257	4258	4259	4260	4261	4269
4270	4271	4272	4273	4274	4275	4276	4280	4281	4282	4283	4284	4288
4289	4290	4291	4294	4296	4297	4298	4299	4301	4305	4307	4308	4310
4313	4314	4315	4316	4317	4318	4319	4320	4323	4324	4325	4326	4327
4328	4331	4333	4334	4335	4337	4339	4345	4346	4347	4348	4361	4362
4363	4364	4365	4367	4372	4374	4379	4381	4386	4388	4393	4395	4400
4402	4407	4409	4414	4415	4416	4417	4419	4432	4433	4436	4438	4439
4440	4441	4442	4444	4446	4447	4448	4454	4455	4462	4463	4467	4472
4473	4474	4476	4481	4483	4488	4495	4502	4509	4549	4550	4551	4552
4553	4554	4563	4564	4565	4566	4567	4568	4569	4570	4576	4577	4578
4579	4580	4581	4582	4583	4584	4585	4586	4587	4591	4592	4593	4597
4598	4606	4607	4610	4611	4614	4616	4617	4619	4620	4621	4625	4627
4628	4629	4632	4633	4639	4641	4642	4644	4645	4647	4648	4650	4651
4652	4653	4654	4655	4657	4658	4659	4660	4662	4665	4667	4668	4669
4670	4672	4675	4677	4679	4680	4681	4696	4697	4698	4699	4700	4702
4707	4709	4714	4716	4721	4723	4728	4730	4735	4737	4742	4744	4749
4750	4751	4752	4754	4767	4768	4771	4772	4777	4779	4780	4781	4790
4791	4792	4793	4794	4796	4801	4808	4815	4822	4829	4836	4899	4900
4901	4902	4903	4904	4913	4914	4915	4916	4917	4918	4924	4925	4926
4927	4928	4929	4930	4931	4935	4936	4937	4938	4939	4943	4944	4945
4946	4954	4955	4958	4959	4962	4964	4965	4966	4971	4973	4976	4977
4978	4986	4987	4988	4989	4990	4991	4992	4993	4994	4995	4996	4997
4998	5001	5003	5004	5006	5007	5008	5009	5011	5015	5017	5018	5020
5027	5028	5029	5030	5043	5044	5045	5046	5047	5049	5054	5056	5061
5063	5068	5070	5075	5077	5082	5084	5089	5091	5096	5097	5098	5099
5101	5114	5115	5118	5119	5122	5123	5130	5139	5140	5144	5149	5151
5156	5158	5163	5170	5177	5184	5191	5240	5241	5251	5252	5253	5254
5255	5256	5257	5258	5259	5260	5261	5262	5268	5269	5270	5271	5272
5273	5283	5284	5285	5286	5287	5288	5289	5290	5291	5292	5293	5294
5296	5297	5298	5299	5300	5301	5302	5303	5304	5305	5306	5307	5309
5310	5311	5312	5313	5314	5315	5316	5317	5318	5319	5320	5321	5322
5326	5327	5335	5339	5343	5344	5348	5349	5350	5351	5352	5353	5354
5355	5358	5364	5365	5372	5373	5374	5375	5376	5377	5386	5390	5392
5394	5404	5405	5406	5407	5408	5410	5415	5417	5422	5424	5429	5431
5436	5438	5443	5445	5450	5452	5462	5463	5464	5465	5466	5467	5468
5469	5470	5471	5472	5473	5475	5477	5482	5487	5488	5492	5494	5495
5497	5502	5507	5508	5509	5515	5517	5518	5520	5521	5523	5528	5533
5543	5544	5545	5546	5547	5548	5549	5550	5553	5554	5566	5567	5568
5569	5570	5573	5574	5578	5579	5598	5599	5600	5601	5602	5603	5604
5605	5606	5607	5608	5609	5614	5615	5625	5633	5634	5635	5636	5637
5638	5675	5677	5678	5681	5690	5691	5692	5693	5694	5695	5696	5697
5708	5709	5710	5711	5712	5713	5722	5723	5724	5725	5726	5727	5728

5729	5737	5738	5739	5740	5743	5744	5745	5747	5748	5751	5752	5753
5754	5755	5758	5759	5764	5767	5770	5771	5774	5776	5777	5778	5779
5780	5784	5786	5787	5789	5790	5791	5792	5795	5797	5798	5800	5801
5802	5803	5804	5805	5806	5807	5810	5812	5813	5814	5817	5818	5821
5823	5824	5825	5827	5828	5832	5834	5835	5838	5839	5840	5841	5844
5845	5847	5849	5850	5851	5854	5856	5857	5858	5859	5860	5866	5867
5870	5872	5873	5875	5879	5882	5884	5888	5890	5891	5892	5906	5907
5908	5909	5910	5911	5912	5913	5914	5917	5918	5919	5920	5924	5925
5926	5927	5930	5931	5932	5933	5934	5935	5936	5937	5938	5939	5940
5941	5955	5956	5957	5958	5959	5961	5966	5968	5973	5975	5980	5982
5987	5989	5994	5996	6001	6003	6014	6015	6016	6017	6018	6019	6021
6023	6028	6036	6037	6038	6039	6041	6063	6064	6065	6066	6067	6074
6081	6088	6095	6102	6109	6116	6117	6120	6129	6136	6139	6144	6145
6155	6237	6238	6239	6240	6241	6242	6251	6252	6253	6254	6255	6256
6257	6258	6266	6267	6268	6269	6272	6273	6274	6276	6277	6279	6280
6283	6284	6285	6288	6289	6292	6294	6295	6298	6299	6303	6304	6305
6306	6311	6313	6314	6317	6318	6322	6323	6324	6325	6332	6334	6337
6338	6339	6340	6342	6345	6346	6347	6348	6352	6353	6356	6358	6361
6363	6364	6365	6366	6367	6369	6370	6371	6372	6373	6374	6376	6378
6379	6380	6386	6388	6389	6390	6391	6392	6393	6398	6400	6401	6402
6405	6406	6408	6410	6411	6412	6413	6414	6415	6416	6419	6421	6423
6425	6427	6428	6429	6433	6434	6437	6439	6440	6442	6443	6444	6445
6446	6449	6450	6453	6455	6457	6459	6461	6462	6464	6467	6469	6470
6471	6472	6473	6477	6478	6490	6491	6492	6493	6494	6495	6496	6497
6498	6501	6502	6503	6504	6508	6509	6510	6511	6514	6515	6516	6517
6518	6519	6520	6521	6522	6523	6524	6525	6539	6540	6541	6542	6543
6545	6550	6552	6557	6559	6564	6566	6571	6573	6578	6580	6585	6587
6598	6599	6600	6601	6602	6603	6605	6607	6612	6620	6621	6622	6623
6625	6647	6648	6649	6650	6651	6658	6665	6672	6679	6686	6693	6700
6701	6704	6709	6710	6715	6720	6801	6802	6803	6804	6805	6806	6815
6816	6817	6818	6819	6820	6826	6827	6828	6829	6830	6833	6834	6835
6836	6840	6841	6842	6843	6846	6847	6848	6849	6857	6858	6859	6860
6861	6862	6863	6865	6866	6869	6870	6871	6872	6873	6874	6878	6880
6881	6885	6888	6889	6890	6891	6892	6893	6897	6899	6900	6901	6910
6911	6912	6913	6914	6916	6917	6918	6919	6920	6924	6925	6934	6935
6939	6941	6942	6944	6945	6947	6950	6952	6954	6955	6956	6957	6958
6959	6960	6963	6964	6967	6969	6973	6974	6988	6989	6990	6991	6992
6993	6994	6995	7009	7010	7011	7012	7013	7015	7020	7022	7027	7029
7034	7036	7041	7043	7048	7050	7055	7057	7063	7064	7065	7066	7068
7089	7090	7091	7092	7093	7100	7107	7114	7121	7128	7135	7142	7144
7149	7151	7226	7227	7228	7229	7230	7231	7240	7241	7242	7243	7244
7245	7251	7252	7253	7254	7255	7258	7259	7260	7261	7265	7266	7267
7268	7271	7272	7273	7274	7280	7281	7282	7283	7284	7285	7286	7287
7288	7290	7291	7294	7295	7296	7297	7298	7299	7303	7305	7306	7307
7308	7311	7312	7313	7314	7320	7322	7324	7327	7329	7331	7333	7334
7335	7337	7340	7341	7342	7343	7344	7349	7351	7353	7355	7357	7358
7359	7363	7365	7366	7369	7370	7371	7372	7375	7376	7377	7378	7381
7382	7383	7387	7388	7389	7390	7393	7395	7398	7400	7401	7403	7406
7408	7410	7414	7415	7416	7417	7418	7426	7430	7434	7435	7442	7444
7445	7446	7447	7449	7453	7454	7464	7465	7466	7467	7468	7469	7470
7471	7485	7486	7487	7488	7489	7491	7496	7498	7503	7505	7510	7512
7517	7519	7524	7526	7531	7533	7539	7540	7541	7542	7544	7565	7566
7567	7568	7569	7576	7578	7580	7588	7590	7592	7600	7602	7607	7609
7973	7974	7975	7976	7977	7978	7987	7988	7989	7990	7991	7992	7993
7994	8000	8001	8002	8003	8004	8005	8006	8009	8010	8011	8012	8016
8017	8018	8019	8022	8023	8024	8025	8034	8035	8043	8045	8046	8047

8054	8055	8058	8059	8060	8062	8063	8065	8069	8070	8074	8076	8077
8078	8082	8083	8095	8097	8098	8099	8103	8104	8115	8117	8118	8119
8120	8123	8124	8132	8133	8136	8138	8139	8140	8141	8152	8155	8156
8157	8158	8159	8161	8162	8163	8164	8165	8168	8169	8170	8171	8174
8176	8177	8178	8179	8191	8193	8194	8195	8205	8207	8209	8210	8211
8221	8223	8227	8228	8237	8239	8240	8241	8251	8252	8253	8255	8256
8257	8258	8277	8278	8280	8282	8283	8284	8291	8292	8295	8297	8305
8306	8307	8310	8312	8313	8314	8319	8322	8323	8324	8327	8329	8330
8333	8334	8338	8339	8348	8350	8351	8354	8356	8357	8358	8361	8362
8367	8369	8370	8372	8374	8375	8376	8379	8382	8384	8386	8387	8388
8391	8393	8394	8396	8397	8399	8400	8401	8404	8405	8412	8414	8415
8418	8420	8421	8424	8425	8428	8429	8436	8438	8439	8441	8442	8445
8446	8450	8451	8458	8460	8461	8463	8464	8465	8466	8469	8470	8474
8475	8485	8486	8487	8488	8489	8491	8492	8493	8494	8495	8504	8505
8506	8507	8508	8510	8515	8517	8522	8524	8529	8531	8536	8538	8543
8545	8550	8552	8559	8560	8561	8562	8563	8564	8566	8568	8573	8587
8588	8589	8590	8592	8613	8614	8615	8616	8617	8619	8624	8626	8628
8636	8638	8640	8650	8657	8664	8671	8772	8773	8774	8775	8776	8777
8786	8787	8788	8789	8790	8791	8792	8793	8802	8803	8810	8811	8821
8822	8823	8824	8825	8826	8827	8829	8830	8834	8838	8839	8849	8851
8855	8858	8861	8862	8863	8866	8867	8871	8872	8882	8884	8885	8888
8889	8890	8891	8899	8901	8902	8905	8906	8907	8908	8909	8911	8912
8913	8916	8917	8925	8927	8928	8931	8932	8933	8935	8936	8937	8940
8941	8949	8951	8952	8955	8956	8960	8961	8969	8971	8972	8975	8976
8980	8981	8993	8995	8998	9000	9001	9003	9010	9012	9013	9015	9016
9019	9020	9024	9026	9028	9030	9031	9035	9036	9041	9043	9044	9045
9054	9056	9058	9061	9064	9065	9076	9078	9082	9083	9092	9094	9095
9096	9106	9108	9109	9110	9114	9115	9125	9126	9129	9130	9131	9133
9134	9142	9145	9146	9149	9150	9151	9152	9164	9168	9169	9170	9171
9172	9174	9175	9176	9177	9180	9181	9182	9183	9186	9187	9188	9189
9202	9203	9204	9205	9206	9207	9208	9209	9210	9211	9212	9213	9214
9215	9216	9217	9218	9219	9220	9221	9222	9223	9224	9225	9233	9234
9235	9236	9237	9239	9244	9246	9251	9253	9258	9260	9265	9267	9272
9274	9279	9281	9288	9289	9290	9291	9292	9293	9295	9297	9299	9302
9310	9311	9312	9313	9315	9337	9338	9339	9340	9341	9348	9350	9352
9360	9362	9364	9374	9381	9388	9542	9543	9544	9545	9554	9555	9556
9559	9561	9562	9564	9567	9576	9577	9578	9579	9580	9581	9582	9583
9584	9585	9586	9587	9588	9589	9590	9591	9592	9593	9594	9595	9596
9597	9598	9599	9613	9615	9616	9617	9618	9619	9620	9623	9624	9629
9630	9637	9639	9640	9641	9644	9645	9646	9647	9651	9653	9655	9656
9657	9658	9659	9660	9661	9669	9670	9672	9674	9675	9676	9679	9680
9681	9682	9686	9688	9690	9691	9692	9693	9694	9695	9696	9704	9705
9707	9709	9710	9711	9714	9715	9716	9717	9721	9723	9725	9726	9727
9728	9729	9730	9731	9739	9740	9742	9744	9745	9746	9749	9750	9751
9752	9756	9758	9760	9761	9762	9763	9764	9765	9766	9774	9775	9777
9779	9780	9781	9784	9785	9786	9787	9791	9793	9795	9796	9797	9798
9799	9800	9801	9809	9810	9812	9814	9815	9816	9819	9820	9825	9827
9828	9829	9830	9831	9840	9841	9843	9845	9846	9847	9848	9849	9858
9859	9861	9863	9864	9865	9866	9867	9876	9877	9879	9881	9882	9883
9884	9885	9889	9890	9891	9892	9898	9899	9904	9905	9909	9910	9911
9913	9944	9945	9946	9947	9948	9949	9960	9961	9964	9965	9966	9967
9976	9977	9982	9983	9986	9987	9988	9989	9998	9999	10004	10005	10008
10009	10010	10011	10020	10021	10026	10027	10030	10031	10032	10033	10042	10043
10048	10049	10052	10053	10054	10055	10064	10065	10070	10071	10074	10075	10084
10085	10090	10091	10094	10095	10104	10105	10110	10111	10114	10115	10124	10125
10130	10131	10134	10135	10144	10145	10148	10149	10152	10154	10155	10156	10157

10158	10159	10162	10163	10207	10209	10210	10213	10222	10223	10224	10225	10226
10227	10228	10229	10241	10242	10243	10244	10245	10246	10252	10256	10275	10276
10277	10278	10279	10280	10281	10282	10283	10284	10285	10286	10287	10288	10289
10290	10296	10299	10300	10301	10302	10303	10304	10305	10306	10316	10317	10325
10328	10329	10330	10331	10332	10333	10334	10335	10345	10346	10354	10357	10358
10359	10360	10361	10362	10363	10364	10376	10377	10385	10386	10387	10389	10390
10391	10392	10393	10394	10396	10397	10398	10399	10400	10401	10403	10404	10405
10406	10407	10408	10409	10410	10411	10418	10420	10430	10431	10432	10434	10435
10436	10457	10458	10459	10462	10463	10467	10468	10471	10472	10477	10479	10480
10481	10482	10484	10485	10486	10489	10490	10495	10497	10498	10499	10500	10502
10503	10504	10507	10508	10514	10516	10517	10519	10520	10523	10524	10528	10529
10536	10538	10539	10541	10542	10543	10544	10547	10548	10552	10553	10565	10567
10577	10578	10579	10581	10582	10583	10604	10605	10606	10609	10610	10614	10615
10618	10619	10624	10626	10627	10628	10629	10631	10632	10633	10636	10637	10642
10644	10645	10646	10647	10649	10650	10651	10654	10655	10660	10662	10663	10665
10666	10669	10670	10674	10675	10682	10684	10685	10687	10688	10689	10690	10693
10694	10698	10699	10711	10713	10723	10724	10725	10727	10728	10729	10750	10751
10752	10755	10758	10759	10763	10764	10767	10768	10775	10777	10778	10781	10782
10783	10784	10785	10787	10788	10789	10792	10793	10800	10802	10803	10806	10807
10808	10809	10810	10812	10813	10814	10817	10818	10825	10827	10828	10830	10831
10834	10835	10839	10840	10847	10849	10850	10852	10853	10854	10855	10858	10859
10863	10864	10875	10876	10877	10878	10883	10884	10885	10886	10887	10888	10889
10890	10891	10892	10893	10894	10895	10896	10897	10898	10906	10907	10908	10909
10910	10912	10917	10919	10924	10926	10931	10933	10938	10940	10945	10947	10952
10954	10964	10965	10966	10967	10968	10969	10971	10973	10978	11001	11002	11003
11004	11006	11025	11026	11027	11029	11030	11031	11041	11042	11043	11045	11046
11047	11057	11058	11059	11061	11062	11063	11082	11083	11084	11085	11086	11098
11100	11102	11110	11112	11114	11127	11128	11129	11130	11131	11132	11134	11136
11141	11162	11163	11164	11165	11166	11167	11173	11177	11196	11197	11198	11199
11200	11201	11202	11203	11204	11205	11206	11207	11208	11209	11210	11211	11212
11213	11214	11215	11221	11224	11225	11226	11227	11228	11229	11230	11231	11241
11242	11250	11253	11254	11255	11256	11257	11258	11259	11260	11270	11271	11279
11282	11283	11284	11285	11286	11287	11288	11289	11301	11302	11310	11311	11312
11314	11315	11316	11317	11318	11319	11321	11322	11323	11324	11325	11326	11328
11329	11330	11331	11332	11333	11334	11335	11336	11343	11345	11355	11356	11357
11359	11360	11361	11382	11383	11384	11387	11388	11392	11393	11396	11397	11402
11404	11405	11406	11407	11409	11410	11411	11414	11415	11420	11422	11423	11424
11425	11427	11428	11429	11432	11433	11439	11441	11442	11444	11445	11448	11449
11453	11454	11461	11463	11464	11466	11467	11468	11469	11472	11473	11477	11478
11490	11492	11497	11498	11504	11505	11506	11508	11509	11510	11531	11532	11533
11536	11537	11541	11542	11545	11546	11551	11553	11554	11555	11556	11558	11559
11560	11563	11564	11569	11571	11572	11573	11574	11576	11577	11578	11581	11582
11587	11589	11590	11592	11593	11596	11597	11601	11602	11609	11611	11612	11614
11615	11616	11617	11620	11621	11625	11626	11638	11640	11650	11651	11652	11654
11655	11656	11677	11678	11679	11682	11685	11686	11690	11691	11694	11695	11702
11704	11705	11708	11709	11710	11711	11712	11714	11715	11716	11719	11720	11727
11729	11730	11733	11734	11735	11736	11737	11739	11740	11741	11744	11745	11752
11754	11755	11757	11758	11761	11762	11766	11767	11774	11776	11777	11779	11780
11781	11782	11785	11786	11790	11791	11802	11803	11804	11805	11810	11811	11812
11813	11814	11815	11816	11817	11818	11819	11820	11821	11822	11823	11824	11825
11833	11834	11835	11836	11837	11839	11844	11846	11851	11853	11858	11860	11865
11867	11872	11874	11879	11881	11892	11893	11894	11895	11896	11897	11899	11901
11906	11929	11930	11931	11932	11934	11953	11954	11955	11957	11958	11959	11969
11970	11971	11973	11974	11975	11985	11986	11987	11989	11990	11991	12010	12011
12012	12013	12014	12026	12028	12030	12038	12040	12042	12055	12056	12057	12058
12059	12060	12062	12064	12069	12090	12091	12092	12093	12094	12095	12101	12102

12103	12104	12105	12106	12112	12113	12114	12115	12116	12117	12118	12119	12120
12121	12127	12129	12130	12131	12132	12133	12134	12138	12142	12143	12147	12150
12151	12154	12156	12157	12158	12159	12160	12161	12165	12169	12170	12174	12175
12182	12184	12187	12188	12192	12194	12197	12198	12202	12204	12206	12207	12208
12215	12217	12220	12222	12223	12225	12226	12227	12233	12234	12235	12237	12240
12241	12245	12247	12249	12250	12251	12256	12258	12264	12265	12266	12267	12268
12269	12273	12275	12276	12277	12283	12286	12289	12290	12296	12298	12299	12300
12308	12309	12332	12334	12335	12336	12340	12341	12342	12344	12345	12346	12348
12351	12353	12354	12357	12358	12362	12363	12366	12367	12368	12369	12372	12374
12375	12378	12380	12381	12384	12385	12389	12390	12393	12394	12395	12396	12401
12403	12404	12407	12408	12409	12412	12413	12417	12418	12421	12422	12423	12424
12429	12431	12432	12434	12435	12438	12439	12443	12444	12449	12451	12452	12453
12454	12457	12458	12462	12463	12466	12468	12469	12470	12473	12474	12478	12480
12481	12482	12486	12487	12488	12490	12491	12492	12494	12497	12499	12500	12503
12504	12508	12509	12512	12513	12514	12515	12518	12520	12521	12523	12524	12527
12528	12532	12533	12536	12537	12538	12539	12542	12544	12545	12546	12547	12550
12551	12555	12556	12559	12560	12561	12562	12565	12567	12568	12570	12571	12574
12575	12579	12580	12585	12587	12588	12589	12590	12593	12594	12598	12599	12602
12604	12605	12606	12614	12616	12617	12618	12619	12621	12622	12623	12624	12625
12626	12627	12635	12636	12637	12638	12639	12641	12646	12653	12660	12667	12674
12681	12688	12689	12691	12693	12698	12699	12701	12703	12709	12710	12711	12712
12714	12727	12728	12729	12730	12731	12732	12733	12734	12735	12736	12737	12738
12745	12746	12747	12748	12749	12751	12756	12758	12760	12762	12768	12770	12772
12774	12808	12809	12810	12811	12812	12813	12814	12815	12819	12820	12821	12822
12828	12829	12830	12831	12832	12833	12834	12835	12843	12844	12845	12846	12848
12851	12852	12853	12854	12865	12866	12867	12870	12871	12874	12875	12879	12881
12884	12885	12891	12894	12895	12896	12900	12904	12905	12911	12913	12917	12918
12925	12928	12930	12931	12933	12934	12935	12936	12937	12938	12941	12943	12944
12945	12949	12950	12955	12958	12959	12963	12965	12968	12969	12970	12972	12974
12975	12976	12979	12980	12981	12982	12986	12988	12990	12991	12992	13000	13003
13005	13006	13008	13009	13010	13013	13015	13016	13017	13018	13019	13020	13024
13028	13029	13036	13038	13039	13043	13047	13048	13052	13053	13055	13057	13060
13061	13062	13063	13073	13076	13077	13083	13084	13085	13087	13088	13089	13090
13105	13106	13109	13111	13118	13121	13122	13123	13125	13126	13127	13131	13134
13135	13136	13138	13139	13140	13144	13147	13148	13149	13152	13154	13155	13158
13159	13163	13164	13169	13170	13171	13174	13176	13177	13179	13181	13185	13186
13190	13191	13196	13198	13200	13201	13202	13209	13211	13212	13213	13214	13217
13219	13220	13221	13224	13227	13229	13230	13231	13236	13238	13239	13242	13244
13247	13248	13251	13253	13254	13257	13258	13262	13263	13268	13271	13273	13274
13276	13278	13282	13283	13287	13288	13293	13295	13297	13298	13299	13302	13303
13304	13305	13306	13307	13312	13314	13315	13318	13321	13322	13326	13327	13332
13335	13337	13339	13343	13344	13348	13349	13354	13356	13358	13359	13360	13363
13364	13365	13366	13367	13368	13373	13375	13379	13381	13382	13385	13386	13390
13391	13397	13399	13400	13402	13406	13407	13411	13412	13417	13419	13421	13422
13423	13426	13429	13430	13435	13437	13438	13440	13441	13442	13443	13445	13446
13447	13451	13454	13455	13459	13460	13463	13465	13469	13470	13474	13475	13480
13482	13484	13485	13486	13489	13492	13493	13503	13505	13506	13507	13508	13510
13511	13512	13513	13514	13515	13516	13524	13525	13526	13527	13528	13530	13535
13537	13539	13547	13549	13550	13552	13557	13559	13565	13566	13567	13568	13569
13570	13571	13572	13574	13576	13581	13582	13583	13586	13587	13591	13604	13608
13612	13613	13617	13618	13620	13623	13624	13625	13626	13630	13646	13647	13648
13649	13650	13651	13652	13653	13654	13655	13656	13657	13658	13659	13660	13661
13662	13663	13664	13665	13666	13667	13668	13669	13678	13679	13680	13681	13682
13684	13689	13691	13693	13701	13703	13705	13714	13715	13718	13725	13730	13731
13733	13735	13737	13743	13744	13797	13799	13800	13803	13812	13813	13814	13815
13816	13817	13818	13819	13831	13832	13833	13834	13835	13836	13842	13846	13865

13866	13867	13868	13869	13870	13871	13872	13873	13874	13875	13876	13877	13878
13879	13880	13886	13887	13923	13925	13926	13927	13928	13930	13931	13932	13933
13935	13936	13937	13944	13945	13946	13947	13955	13956	13957	13958	13959	13961
13966	13968	13973	13975	13980	13982	13987	13989	13994	13996	14001	14003	14013
14014	14015	14016	14017	14018	14020	14022	14027	14035	14036	14037	14038	14040
14059	14060	14061	14062	14063	14064	14065	14066	14067	14068	14069	14070	14071
14072	14078	14079	14123	14124	14125	14126	14127	14139	14141	14143	14151	14153
14155	14168	14169	14170	14171	14172	14174	14177	14178	14179	14180	14181	14182
14184	14186	14191	14212	14213	14214	14215	14216	14217	14223	14227	14246	14247
14248	14249	14250	14251	14252	14253	14254	14255	14256	14257	14258	14259	14260
14261	14262	14263	14264	14265	14271	14272	14309	14311	14312	14313	14314	14316
14317	14318	14319	14321	14322	14323	14330	14331	14332	14333	14341	14342	14343
14344	14345	14347	14352	14354	14359	14361	14366	14368	14373	14375	14380	14382
14387	14389	14400	14401	14402	14403	14404	14405	14407	14409	14414	14422	14423
14424	14425	14427	14446	14447	14448	14449	14450	14451	14452	14453	14454	14455
14456	14457	14458	14459	14460	14461	14462	14463	14469	14470	14511	14512	14513
14514	14515	14527	14529	14531	14539	14541	14543	14556	14557	14558	14559	14560
14562	14565	14566	14567	14568	14569	14570	14572	14574	14579	14600	14601	14602
14603	14604	14605	14611	14612	14613	14614	14615	14616	14622	14623	14624	14625
14626	14627	14628	14629	14630	14631	14638	14640	14641	14642	14643	14644	14645
14649	14653	14654	14658	14659	14660	14662	14663	14664	14665	14666	14667	14671
14675	14676	14680	14681	14688	14690	14693	14694	14698	14700	14703	14704	14708
14710	14712	14713	14714	14721	14723	14726	14728	14729	14731	14732	14733	14739
14740	14741	14743	14746	14747	14751	14753	14755	14756	14757	14762	14764	14770
14771	14772	14773	14774	14775	14779	14781	14782	14783	14789	14792	14795	14796
14802	14804	14805	14806	14814	14815	14835	14836	14837	14839	14840	14841	14842
14845	14848	14849	14853	14854	14860	14863	14864	14865	14866	14870	14872	14873
14874	14877	14878	14879	14880	14881	14882	14886	14887	14890	14893	14894	14898
14899	14902	14903	14907	14908	14909	14911	14912	14913	14914	14917	14920	14921
14925	14926	14932	14935	14936	14937	14938	14942	14944	14945	14946	14949	14950
14951	14952	14953	14954	14958	14959	14962	14965	14966	14970	14971	14974	14975
14983	14985	14986	14987	14988	14990	14991	14992	14993	14994	14995	14996	15004
15005	15006	15007	15008	15010	15015	15017	15019	15027	15029	15031	15039	15040
15041	15042	15044	15058	15059	15060	15061	15062	15063	15064	15065	15066	15067
15068	15069	15076	15077	15078	15079	15080	15082	15087	15089	15091	15093	15099
15101	15103	15105	15139	15140	15141	15142	15143	15144	15145	15146	15152	15153
15159	15160	15161	15162	15163	15164	15165	15166	15174	15175	15176	15177	15178
15181	15184	15185	15189	15190	15196	15199	15200	15201	15202	15206	15208	15209
15210	15211	15212	15216	15219	15220	15224	15225	15237	15238	15239	15242	15243
15246	15247	15251	15253	15256	15257	15263	15264	15267	15268	15269	15271	15275
15276	15280	15281	15287	15289	15293	15294	15297	15299	15301	15307	15309	15310
15311	15314	15315	15316	15317	15318	15319	15320	15323	15325	15326	15327	15331
15332	15333	15334	15335	15336	15339	15340	15341	15342	15347	15350	15351	15352
15353	15354	15355	15358	15359	15360	15361	15365	15367	15370	15373	15374	15375
15377	15379	15380	15381	15384	15385	15389	15390	15393	15394	15395	15396	15400
15402	15404	15405	15406	15411	15413	15415	15422	15425	15427	15428	15430	15431
15432	15435	15437	15438	15439	15440	15441	15445	15447	15450	15451	15454	15455
15456	15457	15458	15459	15460	15461	15463	15465	15467	15470	15473	15474	15475
15476	15480	15481	15484	15485	15486	15487	15488	15489	15506	15508	15509	15510
15511	15513	15514	15515	15516	15517	15518	15519	15523	15524	15525	15526	15527
15529	15534	15536	15544	15546	15554	15556	15557	15559	15564	15566	15573	15574
15575	15576	15577	15578	15579	15580	15582	15584	15589	15590	15591	15594	15595
15599	15606	15607	15609	15612	15613	15614	15615	15619	15635	15636	15637	15638
15639	15640	15651	15652	15653	15654	15655	15662	15664	15666	15674	15676	15678
15687	15688	15691	15698	15703	15704	15706	15708	15710	15714	15715	15769	15771
15772	15775	15784	15785	15786	15787	15788	15789	15790	15791	15804	15805	15806

15807	15808	15809	15815	15819	15838	15839	15840	15841	15842	15843	15844	15845
15846	15847	15848	15849	15850	15851	15852	15853	15859	15860	15896	15898	15899
15900	15901	15903	15904	15905	15906	15908	15909	15910	15917	15918	15919	15920
15928	15929	15930	15931	15932	15934	15939	15941	15946	15948	15953	15955	15960
15962	15967	15969	15974	15976	15987	15988	15989	15990	15991	15992	15994	15996
16001	16009	16010	16011	16012	16014	16033	16034	16035	16036	16037	16038	16039
16040	16041	16042	16043	16044	16045	16046	16052	16053	16095	16096	16097	16098
16099	16111	16113	16115	16123	16125	16127	16140	16141	16142	16143	16144	16146
16149	16150	16151	16152	16153	16154	16156	16158	16163	16202	16204	16205	16208
16217	16218	16219	16220	16221	16222	16223	16224	16237	16238	16239	16240	16241
16242	16248	16252	16271	16272	16273	16274	16275	16276	16277	16278	16279	16280
16281	16282	16283	16284	16285	16286	16287	16288	16289	16290	16296	16297	16334
16336	16337	16338	16339	16341	16342	16343	16344	16346	16347	16348	16355	16356
16357	16358	16366	16367	16368	16369	16370	16372	16377	16379	16384	16386	16391
16393	16398	16400	16405	16407	16412	16414	16425	16426	16427	16428	16429	16430
16432	16434	16439	16447	16448	16449	16450	16452	16471	16472	16473	16474	16475
16476	16477	16478	16479	16480	16481	16482	16483	16484	16485	16486	16487	16488
16494	16495	16537	16538	16539	16540	16541	16553	16555	16557	16565	16567	16569
16582	16583	16584	16585	16586	16588	16591	16592	16593	16594	16595	16596	16598
16600	16605	16859	16861	16864	16865	16866	16867	16979	16980	16981	16982	16983
16984	16985	16986	16987	16988	16989	16990	17008	17009	17010	17011	17012	17013
17014	17015	17018	17019	17020	17021	17025	17026	17027	17028	17045	17046	17047
17048	17049	17050	17051	17052	17055	17056	17057	17058	17062	17063	17064	17065
17082	17083	17084	17085	17086	17087	17088	17089	17092	17093	17094	17095	17099
17100	17101	17102	17117	17118	17119	17120	17121	17122	17123	17124	17127	17128
17129	17130	17134	17135	17136	17137	17155	17156	17157	17158	17159	17160	17161
17162	17165	17166	17167	17168	17172	17173	17174	17175	17191	17192	17193	17194
17195	17196	17197	17198	17199	17200	17201	17202	17218	17219	17220	17221	17222
17223	17224	17225	17226	17227	17228	17229	17254	17255	17262	17265	17266	17270
17273	17274	17284	17287	17288	17289	17290	17291	17292	17293	17294	17295	17296
17297	17298	17299	17300	17301	17302	17304	17305	17306	17307	17308	17309	17310
17311	17312	17313	17314	17315	17316	17317	17318	17319	17327	17330	17331	17338
17339	17390	17391	17392	17393	17394	17396	17402	17405	17406	17410	17411	17422
17425	17426	17430	17433	17434	17435	17436	17440	17442	17443	17444	17447	17448
17449	17450	17451	17452	17456	17457	17460	17461	17462	17463			
720*	1269	1272	1314	1345	1348	1358	1367	1374	1381	1388	1395	1402
1405	1415	1450	1457	1464	1471	1478	1485	1534	1565	1568	1578	1587
1594	1601	1608	1615	1622	1653	1660	1667	1674	1681	1688	1695	1762
1780	1808	1816	1832	1840	1854	1882	1889	1896	1903	1910	1917	1924
1942	1948	1950	1953	1971	1973	1976	1996	2015	2018	2038	2058	2111
2118	2125	2132	2139	2146	2153	2160	2174	2378	2396	2424	2432	2448
2456	2488	2495	2502	2509	2516	2523	2530	2546	2556	2558	2561	2578
2605	2607	2610	2622	2630	2653	2706	2713	2720	2727	2734	2741	2748
2755	2768	2891	2919	2927	2943	2951	3009	3016	3023	3030	3037	3044
3051	3073	3079	3081	3084	3102	3104	3107	3128	3145	3157	3177	3232
3239	3246	3253	3260	3267	3274	3281	3295	3417	3475	3482	3489	3496
3503	3510	3517	3529	3571	3578	3585	3592	3599	3606	3703	3761	3768
3775	3782	3789	3796	3803	3815	3858	3865	3872	3879	3886	3893	3983
4026	4057	4064	4071	4078	4085	4092	4099	4111	4136	4169	4176	4183
4190	4197	4204	4296	4339	4367	4374	4381	4388	4395	4402	4409	4421
4446	4476	4483	4490	4497	4504	4511	4616	4657	4667	4677	4679	4702
4709	4716	4723	4730	4737	4744	4756	4796	4803	4810	4817	4824	4831
4838	4964	5000	5006	5049	5056	5063	5070	5077	5084	5091	5103	5151
5158	5165	5172	5179	5186	5193	5240	5243	5360	5394	5410	5417	5424
5431	5438	5445	5452	5477	5479	5482	5502	5517	5528	5572	5627	5680
5683	5750	5757	5766	5769	5786	5797	5812	5834	5837	5849	5872	5881

SF\$OR = 000320

5916	5961	5968	5975	5982	5989	5996	6003	6023	6025	6028	6043	6069	
6076	6083	6090	6097	6104	6111	6120	6129	6139	6150	6282	6287	6294	
6297	6313	6316	6334	6336	6344	6358	6360	6369	6378	6388	6408	6410	
6421	6427	6442	6455	6461	6469	6500	6545	6552	6559	6566	6573	6580	
6587	6607	6609	6612	6627	6653	6660	6667	6674	6681	6688	6695	6704	
6715	6832	6868	6887	6899	6944	6950	6962	6969	7015	7022	7029	7036	
7043	7050	7057	7070	7095	7102	7109	7116	7123	7130	7137	7144	7151	
7257	7293	7326	7329	7340	7351	7357	7386	7400	7408	7414	7444	7449	
7491	7498	7505	7512	7519	7526	7533	7546	7571	7578	7580	7590	7592	
7602	7609	8008	8065	8076	8097	8154	8167	8193	8207	8209	8223	8239	
8282	8297	8309	8321	8326	8332	8350	8353	8369	8372	8374	8381	8384	
8386	8399	8414	8417	8423	8438	8444	8460	8468	8510	8517	8524	8531	
8538	8545	8552	8568	8570	8573	8594	8619	8626	8628	8638	8640	8650	
8657	8664	8671	8857	8860	8865	8884	8887	8901	8904	8911	8927	8930	
8935	8951	8954	8971	8974	9018	9026	9043	9056	9058	9061	9063	9078	
9094	9108	9144	9166	9179	9239	9246	9253	9260	9267	9274	9281	9297	
9299	9302	9317	9343	9350	9352	9362	9364	9374	9381	9388	9564	9566	
9569	9615	9622	9639	9653	9655	9674	9688	9690	9709	9723	9725	9744	
9758	9760	9779	9793	9795	9814	9827	9845	9863	9881	9903	9959	9981	
10003	10025	10047	10069	10089	10109	10129	10154	10161	10212	10215	10298	10327	
10356	10420	10461	10484	10502	10522	10546	10567	10608	10631	10649	10668	10692	
10713	10754	10757	10780	10787	10805	10812	10833	10857	10912	10919	10926	10933	
10940	10947	10954	10973	10975	10978	11008	11088	11100	11102	11112	11114	11136	
11138	11141	11223	11252	11281	11345	11386	11409	11427	11447	11471	11492	11535	
11558	11576	11595	11619	11640	11681	11684	11707	11714	11732	11739	11760	11784	
11839	11846	11853	11860	11867	11874	11881	11901	11903	11906	11936	12016	12028	
12030	12040	12042	12064	12066	12069	12149	12184	12186	12194	12196	12204	12206	
12217	12219	12222	12225	12237	12239	12247	12249	12258	12273	12275	12283	12286	
12288	12296	12298	12334	12350	12356	12374	12377	12383	12403	12406	12411	12431	
12437	12451	12456	12480	12496	12502	12520	12526	12544	12549	12567	12573	12587	
12592	12641	12648	12655	12662	12669	12676	12683	12693	12703	12716	12751	12758	
12760	12770	12772	12850	12869	12881	12883	12893	12913	12927	12940	12943	12957	
12965	12967	12972	12974	12988	12990	13002	13008	13015	13057	13059	13075	13111	
13120	13133	13146	13151	13157	13173	13179	13181	13196	13198	13200	13211	13216	
13226	13238	13241	13244	13246	13256	13270	13276	13278	13293	13295	13297	13314	
13317	13320	13334	13337	13339	13354	13356	13358	13375	13384	13402	13417	13419	
13421	13428	13437	13453	13465	13480	13482	13484	13491	13530	13537	13539	13552	
13559	13576	13578	13581	13593	13622	13632	13684	13691	13693	13703	13705	13718	
13725	13735	13737	13740	13802	13805	13961	13968	13975	13982	13989	13996	14003	
14022	14024	14027	14042	14129	14141	14143	14153	14155	14186	14188	14191	14347	
14354	14361	14368	14375	14382	14389	14409	14411	14414	14429	14517	14529	14531	
14541	14543	14574	14576	14579	14690	14692	14700	14702	14710	14712	14723	14725	
14728	14731	14743	14745	14753	14755	14764	14779	14781	14789	14792	14794	14802	
14804	14844	14847	14862	14872	14892	14916	14919	14934	14944	14964	15010	15017	
15019	15029	15031	15046	15082	15089	15091	15101	15103	15180	15183	15198	15208	
15218	15241	15253	15255	15271	15289	15299	15301	15303	15322	15325	15349	15367	
15369	15372	15377	15379	15402	15404	15413	15415	15417	15424	15430	15437	15467	
15469	15472	15529	15536	15546	15559	15566	15584	15586	15589	15601	15611	15621	
15657	15664	15666	15676	15678	15691	15698	15708	15710	15713	15774	15777	15934	
15941	15948	15955	15962	15969	15976	15996	15998	16001	16016	16101	16113	16115	
16125	16127	16158	16160	16163	16207	16210	16372	16379	16386	16393	16400	16407	
16414	16434	16436	16439	16454	16543	16555	16557	16567	16569	16600	16602	16605	
16861	16863	17264	17272	17286	17329	17404	17424	17432	17442				
	720#												
\$F SRTI- 000350	720#	9534	9918	9936	10170	16978	16991	17007	17029	17044	17066	17081	17103
\$F SRTN- 000300	720#												
	17116	17138	17154	17176	17190	17203	17217	17230	17246	17342	17389	17466	

\$F SRTI- 000350
\$F SRTN- 000300

*F\$SEL= 000140

720#	10442	10455	10474	10492	10511	10533	10556	10589	10602	10621	10639	10657
10679	10702	10735	10748	10772	10797	10822	10844	10867	11367	11380	11399	11417
11436	11458	11481	11516	11529	11548	11566	11584	11606	11629	11662	11675	11699
11724	11749	11771	11794									

\$F\$ME= 000330

720#	1269	1272	1314	1345	1348	1358	1367	1374	1381	1388	1395	1402
1405	1415	1450	1457	1464	1471	1478	1485	1534	1565	1568	1578	1587
1594	1601	1608	1615	1622	1653	1660	1667	1674	1681	1688	1695	1762
1783	1808	1816	1832	1840	1854	1882	1889	1896	1903	1910	1917	1924
1942	1953	1976	1996	2015	2018	2038	2058	2111	2118	2125	2132	2139
2146	2153	2160	2174	2378	2399	2424	2432	2448	2456	2488	2495	2502
2509	2516	2523	2530	2546	2561	2578	2610	2622	2630	2653	2706	2713
2720	2727	2734	2741	2748	2755	2768	2894	2919	2927	2943	2951	3009
3016	3023	3030	3037	3044	3051	3073	3084	3107	3128	3145	3157	3177
3232	3239	3246	3253	3260	3267	3274	3281	3295	3417	3475	3482	3489
3496	3503	3510	3517	3529	3571	3578	3585	3592	3599	3606	3703	3761
3768	3775	3782	3789	3796	3803	3815	3858	3865	3872	3879	3886	3893
3983	4057	4064	4071	4078	4085	4092	4099	4111	4169	4176	4183	4190
4197	4204	4296	4367	4374	4381	4388	4395	4402	4409	4421	4476	4483
4490	4497	4504	4511	4616	4657	4667	4679	4702	4709	4716	4723	4730
4737	4744	4756	4796	4803	4810	4817	4824	4831	4838	4964	5000	5006
5049	5056	5063	5070	5077	5084	5091	5103	5151	5158	5165	5172	5179
5186	5193	5240	5243	5394	5410	5417	5424	5431	5438	5445	5452	5482
5502	5517	5528	5572	5627	5680	5683	5750	5757	5766	5769	5786	5797
5812	5834	5837	5872	5881	5916	5961	5968	5975	5982	5989	5996	6003
6028	6043	6069	6076	6083	6090	6097	6104	6111	6120	6129	6139	6150
6282	6287	6294	6297	6313	6316	6336	6360	6369	6388	6410	6424	6427
6442	6458	6461	6469	6500	6545	6552	6559	6566	6573	6580	6587	6612
6627	6653	6660	6667	6674	6681	6688	6695	6704	6715	6832	6868	6887
6944	6953	6962	6972	7015	7022	7029	7036	7043	7050	7057	7070	7095
7102	7109	7116	7123	7130	7137	7144	7151	7257	7293	7326	7332	7340
7354	7357	7386	7400	7411	7414	7444	7452	7491	7498	7505	7512	7519
7526	7533	7546	7571	7583	7595	7602	7609	8008	8068	8076	8097	8167
8209	8226	8239	8297	8309	8321	8326	8332	8350	8353	8369	8374	8381
8386	8399	8414	8417	8423	8438	8444	8460	8468	8510	8517	8524	8531
8538	8545	8552	8573	8594	8619	8631	8643	8650	8657	8664	8671	8857
8860	8865	8884	8887	8901	8904	8911	8927	8930	8935	8951	8954	8971
8974	9018	9029	9043	9058	9063	9081	9094	9108	9144	9179	9239	9246
9253	9260	9267	9274	9281	9302	9317	9343	9355	9367	9374	9381	9388
9566	9569	9615	9622	9655	9690	9725	9760	9795	10154	10161	10212	10215
10298	10327	10356	10420	10461	10484	10502	10522	10546	10567	10608	10631	10649
10668	10692	10713	10754	10757	10780	10787	10805	10812	10833	10857	10912	10919
10926	10933	10940	10947	10954	10978	11008	11088	11105	11117	11141	11223	11252
11281	11345	11386	11409	11427	11447	11471	11492	11535	11558	11576	11595	11619
11640	11681	11684	11707	11714	11732	11739	11760	11784	11839	11846	11853	11860
11867	11874	11881	11906	11936	12016	12033	12045	12069	12149	12186	12196	12206
12219	12222	12225	12239	12249	12258	12283	12288	12298	12334	12350	12356	12374
12377	12383	12403	12406	12411	12431	12437	12451	12456	12480	12496	12502	12520
12526	12544	12549	12567	12573	12587	12592	12641	12648	12655	12662	12669	12676
12683	12693	12703	12716	12751	12763	12775	12869	12883	12893	12916	12927	12940
12943	12957	12974	12990	13002	13008	13015	13075	13111	13120	13133	13146	13151
13157	13173	13184	13200	13211	13216	13226	13238	13241	13246	13256	13270	13281
13297	13314	13317	13320	13334	13342	13358	13375	13384	13405	13421	13428	13437
13453	13468	13484	13491	13530	13542	13552	13559	13581	13593	13622	13632	13684
13696	13708	13718	13725	13740	13802	13805	13961	13968	13975	13982	13989	13996
14003	14027	14042	14129	14146	14158	14191	14347	14354	14361	14368	14375	14382
14389	14414	14429	14517	14534	14546	14579	14692	14702	14712	14725	14728	14731

14745	14755	14764	14789	14794	14804	14844	14847	14862	14892	14916	14919	14934		
14964	15010	15022	15034	15046	15082	15094	15106	15180	15183	15198	15218	15241		
15255	15274	15292	15303	15322	15325	15349	15372	15379	15404	15417	15424	15430		
15437	15472	15529	15539	15549	15559	15566	15589	15601	15611	15621	15657	15669		
15681	15691	15698	15713	15774	15777	15934	15941	15948	15955	15962	15969	15976		
16001	16016	16101	16118	16130	16163	16207	16210	16372	16379	16386	16393	16400		
16407	16414	16439	16454	16543	16560	16572	16605	16863	17264	17272	17286	17329		
17404	17424	17432												
SFSTRU= 000404	720#	10455	10474	10475	10492	10493	10511	10512	10533	10534	10602	10621	10622	
		10639	10640	10657	10658	10679	10680	10748	10772	10773	10797	10798	10822	10823
		10844	10845	11380	11399	11400	11417	11418	11436	11437	11458	11459	11529	11548
		11549	11566	11567	11584	11585	11606	11607	11675	11699	11700	11724	11725	11749
		11750	11771	11772										
SFSUNT= 000130	720#	1302	1424	1522	1629	1992	2003	2034	2043	2170	2179	2574	2587	
		2626	2638	2764	2773	3124	3135	3162	3291	3300	4967	4974	5124	
		5128	5281	5386	5498	5509	5524	5533	6125	6134	6146	6155	6396	6403
		6711	6720	7309	7315	7373	7379	17016	17022	17053	17059	17090	17096	17125
		17131	17163	17169										
SFSWHI= 000120	720#	1780	1945	1946	1948	1960	1968	1969	1971	1983	2396	2553	2554	
		2556	2568	2602	2603	2605	2615	2891	3076	3077	3079	3091	3099	3100
		3102	3115	4023	4024	4029	4133	4134	4139	4336	4337	4342	4443	4444
		4449	4677	5357	5358	5361	5474	5475	5477	5489	5846	5847	5861	6020
		6021	6023	6033	6334	6341	6342	6349	6358	6375	6376	6381	6408	6421
		6455	6604	6605	6607	6617	6896	6897	6921	6950	6969	7329	7351	7408
		7449	7578	7590	8065	8151	8152	8182	8190	8191	8196	8207	8223	8279
		8280	8285	8372	8384	8565	8566	8568	8578	8626	8638	9026	9056	9061
		9078	9163	9164	9192	9294	9295	9297	9307	9350	9362	9564	9636	9637
		9648	9653	9671	9672	9683	9688	9706	9707	9718	9723	9741	9742	9753
		9758	9776	9777	9788	9793	9811	9812	9821	9824	9825	9832	9842	9843
		9850	9860	9861	9868	9878	9879	9886	9900	9901	9906	9956	9957	9968
		9978	9979	9990	10000	10001	10012	10022	10023	10034	10044	10045	10056	10066
		10067	10076	10086	10087	10096	10106	10107	10116	10126	10127	10136	10970	10971
		10973	10983	11100	11112	11133	11134	11136	11146	11898	11899	11901	11911	12028
		12040	12061	12062	12064	12074	12184	12194	12204	12217	12237	12247	12270	12271
		12273	12278	12286	12296	12758	12770	12847	12848	12855	12881	12913	12962	12963
		12965	12972	12983	12988	13054	13055	13057	13064	13179	13196	13244	13276	13293
		13337	13354	13402	13417	13465	13480	13537	13573	13574	13576	13588	13691	13703
		13732	13733	13735	13747	14019	14020	14022	14032	14141	14153	14183	14184	14186
		14196	14406	14407	14409	14419	14529	14541	14571	14572	14574	14584	14690	14700
		14710	14723	14743	14753	14776	14777	14779	14784	14792	14802	14869	14870	14883
		14941	14942	14955	15017	15029	15089	15101	15205	15206	15213	15253	15271	15289
		15299	15364	15365	15367	15377	15397	15402	15413	15464	15465	15467	15490	15536
		15546	15581	15582	15584	15596	15664	15676	15705	15706	15708	15720	15993	15994
		15996	16006	16113	16125	16155	16156	16158	16168	16431	16432	16434	16444	16555
		16567	16597	16598	16600	16610	16861	17439	17440	17453				
SFSYES= 000402	720#	1266	1267	1270	1275	1277	1281	1283	1285	1299	1301	1312	1316	
		1321	1325	1330	1333	1335	1337	1343	1346	1351	1353	1356	1361	1363
		1365	1370	1372	1377	1379	1384	1386	1391	1393	1398	1400	1403	1408
		1410	1413	1418	1420	1423	1433	1435	1448	1453	1455	1460	1462	1467
		1469	1474	1476	1481	1483	1488	1519	1521	1532	1536	1541	1545	1550
		1553	1555	1557	1563	1566	1571	1573	1576	1581	1583	1585	1590	1592
		1597	1599	1604	1606	1611	1613	1618	1620	1625	1628	1651	1656	1658
		1663	1665	1670	1672	1677	1679	1684	1686	1691	1693	1698	1714	1716
		1718	1728	1730	1732	1734	1736	1738	1740	1742	1751	1753	1755	1757
		1759	1760	1764	1765	1769	1770	1775	1777	1778	1780	1785	1786	1791
		1792	1795	1797	1799	1801	1803	1805	1806	1811	1814	1819	1824	1826

1829	1830	1835	1838	1843	1848	1850	1852	1856	1857	1861	1862	1877
1879	1880	1885	1887	1892	1894	1899	1901	1906	1908	1913	1915	1920
1922	1927	1935	1937	1939	1940	1944	1946	1948	1951	1956	1959	1963
1967	1969	1971	1974	1979	1982	1988	1991	1994	1999	2002	2009	2012
2013	2016	2020	2021	2025	2026	2030	2033	2036	2041	2049	2052	2054
2056	2061	2070	2074	2078	2080	2082	2084	2086	2088	2090	2099	2101
2105	2107	2109	2114	2116	2121	2123	2128	2130	2135	2137	2142	2144
2149	2151	2156	2158	2163	2166	2169	2172	2177	2331	2333	2335	2345
2347	2349	2351	2353	2355	2357	2359	2367	2369	2371	2373	2375	2376
2380	2381	2385	2386	2391	2393	2394	2396	2401	2402	2407	2408	2411
2413	2415	2417	2419	2421	2422	2427	2430	2435	2440	2442	2445	2446
2451	2454	2459	2464	2466	2469	2483	2485	2486	2491	2493	2498	2500
2505	2507	2512	2514	2519	2521	2526	2528	2533	2541	2543	2544	2548
2551	2552	2554	2556	2559	2564	2567	2571	2576	2581	2585	2586	2594
2597	2598	2601	2603	2605	2608	2613	2618	2620	2625	2628	2633	2637
2644	2647	2649	2651	2656	2665	2669	2673	2675	2677	2679	2681	2683
2685	2694	2696	2700	2702	2704	2709	2711	2716	2718	2723	2725	2730
2732	2737	2739	2744	2746	2751	2753	2758	2761	2763	2766	2771	2835
2837	2839	2849	2851	2853	2855	2857	2859	2861	2863	2872	2874	2876
2878	2880	2882	2886	2888	2889	2891	2896	2897	2902	2903	2906	2908
2910	2912	2914	2916	2917	2922	2925	2930	2935	2937	2940	2941	2946
2949	2954	2959	2961	2964	2970	2973	2979	2984	2986	2988	2990	3004
3006	3007	3012	3014	3019	3021	3026	3028	3033	3035	3040	3042	3047
3049	3054	3062	3064	3066	3068	3070	3071	3075	3077	3079	3082	3087
3090	3094	3098	3100	3102	3105	3111	3114	3120	3123	3126	3131	3134
3141	3143	3148	3149	3152	3155	3160	3168	3171	3173	3175	3180	3189
3193	3197	3199	3201	3203	3205	3207	3209	3220	3222	3226	3228	3230
3235	3237	3242	3244	3249	3251	3256	3258	3263	3265	3270	3272	3277
3279	3284	3288	3290	3293	3298	3362	3364	3366	3375	3377	3379	3381
3389	3391	3393	3396	3399	3401	3403	3404	3408	3410	3412	3413	3415
3419	3422	3423	3428	3431	3432	3435	3437	3439	3441	3445	3447	3449
3454	3456	3470	3472	3473	3478	3480	3485	3487	3492	3494	3499	3501
3506	3508	3513	3515	3520	3523	3525	3527	3532	3541	3548	3550	3558
3561	3565	3568	3569	3574	3576	3581	3583	3588	3590	3595	3597	3602
3604	3609	3646	3648	3650	3660	3662	3664	3666	3675	3677	3679	3682
3685	3687	3689	3690	3694	3696	3698	3699	3701	3705	3708	3709	3714
3717	3718	3721	3723	3725	3727	3731	3733	3735	3740	3742	3756	3758
3759	3764	3766	3771	3773	3778	3780	3785	3787	3792	3794	3799	3801
3806	3809	3811	3813	3818	3827	3834	3836	3844	3847	3851	3855	3856
3861	3863	3868	3870	3875	3877	3882	3884	3889	3891	3896	3930	3932
3934	3944	3946	3948	3957	3959	3961	3963	3966	3969	3971	3972	3976
3978	3979	3981	3985	3988	3989	3994	3997	3998	4001	4003	4005	4007
4011	4013	4015	4020	4022	4024	4033	4036	4038	4052	4054	4055	4060
4062	4067	4069	4074	4076	4081	4083	4088	4090	4095	4097	4102	4105
4107	4109	4114	4123	4128	4130	4132	4134	4144	4146	4156	4159	4163
4166	4167	4172	4174	4179	4181	4186	4188	4193	4195	4200	4202	4207
4243	4245	4247	4257	4259	4261	4270	4272	4274	4276	4279	4282	4284
4285	4289	4291	4292	4294	4298	4301	4302	4307	4310	4311	4314	4316
4318	4320	4324	4326	4328	4333	4335	4337	4346	4348	4362	4364	4365
4370	4372	4377	4379	4384	4386	4391	4393	4398	4400	4405	4407	4412
4415	4417	4419	4424	4433	4438	4440	4442	4444	4448	4455	4463	4466
4470	4473	4474	4479	4481	4486	4488	4493	4495	4500	4502	4507	4509
4514	4550	4552	4554	4564	4566	4568	4570	4577	4579	4581	4583	4585
4587	4590	4593	4594	4598	4599	4607	4611	4614	4619	4621	4622	4627
4629	4630	4633	4641	4644	4647	4650	4652	4654	4655	4659	4662	4663
4665	4669	4672	4673	4675	4677	4681	4682	4697	4699	4700	4705	4707

4712	4714	4719	4721	4726	4728	4733	4735	4740	4742	4747	4750	4752
4754	4759	4768	4772	4779	4781	4791	4793	4794	4799	4801	4806	4808
4813	4815	4820	4822	4827	4829	4834	4836	4841	4900	4902	4904	4914
4916	4918	4925	4927	4929	4931	4934	4937	4939	4940	4944	4946	4947
4955	4959	4962	4966	4973	4978	4979	4987	4989	4991	4993	4995	4997
4998	5003	5004	5008	5011	5012	5017	5020	5021	5023	5028	5030	5044
5046	5047	5052	5054	5059	5061	5066	5068	5073	5075	5080	5082	5087
5089	5094	5097	5099	5101	5106	5115	5119	5123	5140	5143	5147	5149
5154	5156	5161	5163	5168	5170	5175	5177	5182	5184	5189	5191	5196
5238	5241	5246	5248	5252	5254	5256	5258	5260	5262	5269	5271	5273
5284	5286	5288	5290	5292	5294	5297	5299	5301	5303	5305	5307	5310
5312	5314	5316	5318	5320	5322	5327	5335	5338	5344	5349	5351	5353
5355	5358	5365	5373	5375	5377	5392	5397	5405	5407	5408	5413	5415
5420	5422	5427	5429	5434	5436	5441	5443	5448	5450	5455	5463	5465
5467	5469	5471	5473	5475	5477	5480	5485	5488	5494	5497	5500	5505
5508	5515	5520	5523	5526	5531	5539	5541	5544	5546	5548	5550	5554
5567	5569	5570	5574	5575	5579	5580	5599	5601	5603	5605	5607	5609
5615	5625	5631	5634	5636	5638	5677	5678	5681	5686	5688	5691	5693
5695	5697	5709	5711	5713	5723	5725	5727	5729	5738	5740	5744	5747
5748	5752	5754	5755	5759	5760	5762	5764	5767	5771	5776	5778	5780
5781	5784	5789	5790	5792	5793	5795	5800	5801	5803	5805	5807	5808
5810	5814	5818	5823	5824	5827	5828	5829	5832	5835	5839	5841	5842
5845	5847	5851	5856	5858	5860	5864	5867	5870	5875	5876	5879	5884
5885	5890	5892	5893	5895	5897	5899	5907	5909	5911	5913	5914	5918
5920	5921	5925	5927	5928	5931	5933	5935	5937	5939	5941	5956	5958
5959	5964	5966	5971	5973	5978	5980	5985	5987	5992	5994	5999	6001
6006	6015	6017	6019	6021	6023	6026	6031	6037	6039	6041	6046	6064
6066	6067	6072	6074	6079	6081	6086	6088	6093	6095	6100	6102	6107
6109	6114	6117	6118	6123	6127	6132	6137	6142	6145	6148	6153	6238
6240	6242	6252	6254	6256	6258	6267	6269	6273	6276	6279	6280	6284
6285	6289	6290	6292	6295	6299	6300	6304	6306	6307	6309	6311	6314
6318	6319	6323	6325	6326	6328	6330	6332	6334	6338	6340	6342	6346
6348	6353	6354	6356	6358	6363	6364	6366	6367	6371	6372	6374	6376
6380	6384	6386	6390	6391	6393	6394	6400	6402	6406	6408	6412	6414
6416	6417	6419	6421	6425	6429	6430	6434	6435	6439	6440	6444	6446
6447	6450	6451	6453	6455	6459	6464	6465	6467	6471	6473	6474	6478
6479	6481	6483	6491	6493	6495	6497	6498	6502	6504	6505	6509	6511
6512	6515	6517	6519	6521	6523	6525	6540	6542	6543	6548	6550	6555
6557	6562	6564	6569	6571	6576	6578	6583	6585	6590	6599	6601	6603
6605	6607	6610	6615	6621	6623	6625	6630	6648	6650	6651	6656	6658
6663	6665	6670	6672	6677	6679	6684	6686	6691	6693	6698	6701	6702
6707	6710	6713	6718	6802	6804	6806	6816	6818	6820	6827	6829	6830
6834	6836	6837	6841	6843	6844	6847	6849	6858	6860	6862	6865	6866
6870	6872	6874	6875	6880	6881	6885	6889	6891	6893	6894	6897	6901
6911	6913	6916	6918	6920	6925	6935	6941	6942	6947	6948	6950	6955
6957	6959	6960	6964	6965	6967	6969	6974	6975	6977	6979	6981	6989
6991	6993	6995	7010	7012	7013	7018	7020	7025	7027	7032	7034	7039
7041	7046	7048	7053	7055	7060	7064	7066	7068	7073	7090	7092	7093
7098	7100	7105	7107	7112	7114	7119	7121	7126	7128	7133	7135	7140
7142	7147	7149	7154	7227	7229	7231	7241	7243	7245	7252	7254	7255
7259	7261	7262	7266	7268	7269	7272	7274	7281	7283	7285	7287	7290
7291	7295	7297	7299	7300	7305	7306	7308	7312	7314	7318	7322	7324
7327	7329	7334	7337	7338	7342	7344	7345	7347	7349	7351	7355	7359
7360	7365	7366	7367	7370	7372	7376	7378	7383	7384	7388	7390	7391
7395	7396	7398	7403	7404	7406	7408	7412	7416	7418	7419	7421	7423
7426	7429	7435	7442	7446	7447	7449	7454	7455	7457	7465	7467	7469

7471	7486	7488	7489	7494	7496	7501	7503	7508	7510	7515	7517	7522
7524	7529	7531	7536	7540	7542	7544	7549	7566	7568	7569	7574	7576
7578	7580	7586	7588	7590	7592	7598	7600	7605	7607	7612	7974	7976
7978	7988	7990	7992	7994	8001	8003	8005	8006	8010	8012	8013	8017
8019	8020	8023	8025	8035	8045	8047	8055	8059	8062	8063	8065	8070
8071	8074	8078	8079	8083	8084	8086	8095	8099	8100	8104	8105	8117
8118	8120	8124	8133	8138	8139	8141	8152	8156	8158	8161	8162	8164
8165	8169	8171	8176	8177	8179	8180	8191	8195	8205	8207	8211	8212
8221	8223	8228	8229	8237	8241	8242	8252	8255	8256	8258	8278	8280
8284	8292	8295	8300	8306	8307	8312	8314	8315	8319	8323	8324	8329
8330	8334	8335	8339	8340	8342	8344	8348	8351	8356	8358	8359	8362
8363	8367	8370	8372	8376	8377	8379	8382	8384	8388	8389	8393	8396
8397	8401	8402	8405	8406	8408	8412	8415	8420	8421	8425	8426	8429
8430	8432	8436	8441	8442	8446	8447	8451	8452	8454	8458	8463	8465
8466	8470	8471	8475	8476	8478	8486	8488	8491	8493	8495	8505	8507
8508	8513	8515	8520	8522	8527	8529	8534	8536	8541	8543	8548	8550
8555	8560	8562	8564	8566	8568	8571	8576	8588	8590	8592	8597	8614
8616	8617	8622	8624	8626	8628	8634	8636	8638	8640	8646	8648	8653
8655	8660	8662	8667	8669	8674	8773	8775	8777	8787	8789	8791	8793
8803	8811	8822	8824	8826	8829	8830	8833	8839	8851	8855	8858	8862
8863	8867	8868	8872	8873	8875	8877	8882	8885	8889	8891	8892	8894
8899	8902	8906	8908	8909	8913	8914	8917	8918	8920	8925	8928	8932
8933	8937	8938	8941	8942	8944	8949	8952	8956	8957	8961	8962	8964
8969	8972	8976	8977	8981	8982	8984	8995	9000	9003	9012	9015	9016
9020	9021	9024	9026	9031	9032	9036	9037	9039	9041	9045	9046	9054
9056	9059	9061	9065	9066	9068	9076	9078	9083	9084	9092	9096	9097
9106	9110	9111	9115	9116	9126	9130	9133	9134	9142	9146	9150	9152
9155	9164	9169	9171	9174	9176	9177	9181	9183	9187	9189	9190	9203
9205	9207	9209	9211	9213	9215	9217	9219	9221	9223	9225	9234	9236
9237	9242	9244	9249	9251	9256	9258	9263	9265	9270	9272	9277	9279
9284	9289	9291	9293	9295	9297	9300	9305	9311	9313	9315	9320	9338
9340	9341	9346	9348	9350	9352	9358	9360	9362	9364	9370	9372	9377
9379	9384	9386	9391	9543	9545	9553	9556	9557	9561	9562	9564	9567
9573	9577	9579	9581	9583	9585	9587	9589	9591	9593	9595	9597	9599
9603	9605	9613	9617	9619	9620	9624	9625	9627	9630	9637	9641	9645
9647	9651	9653	9657	9659	9661	9662	9670	9672	9676	9680	9682	9686
9688	9692	9694	9696	9697	9705	9707	9711	9715	9717	9721	9723	9727
9729	9731	9732	9740	9742	9746	9750	9752	9756	9758	9762	9764	9766
9767	9775	9777	9781	9785	9787	9791	9793	9797	9799	9801	9802	9810
9812	9816	9820	9825	9829	9831	9841	9843	9847	9849	9859	9861	9865
9867	9877	9879	9883	9885	9890	9892	9899	9901	9905	9910	9913	9945
9947	9949	9957	9961	9965	9967	9977	9979	9983	9987	9989	9999	10001
10005	10009	10011	10021	10023	10027	10031	10033	10043	10045	10049	10053	10055
10065	10067	10071	10075	10085	10087	10091	10095	10105	10107	10111	10115	10125
10127	10131	10135	10145	10149	10152	10156	10158	10159	10163	10164	10166	10209
10210	10213	10218	10220	10223	10225	10227	10229	10242	10244	10246	10252	10255
10276	10278	10280	10282	10284	10286	10288	10290	10296	10300	10302	10304	10306
10313	10317	10318	10325	10329	10331	10333	10335	10342	10346	10347	10354	10358
10360	10362	10364	10373	10377	10378	10386	10389	10391	10393	10396	10398	10400
10403	10405	10407	10409	10411	10418	10423	10431	10434	10436	10443	10448	10458
10459	10463	10464	10468	10469	10472	10479	10481	10482	10486	10487	10490	10497
10499	10500	10504	10505	10508	10516	10519	10520	10524	10525	10529	10530	10538
10541	10543	10544	10548	10549	10553	10554	10556	10565	10570	10578	10581	10583
10590	10595	10605	10606	10610	10611	10615	10616	10619	10626	10628	10629	10633
10634	10637	10644	10646	10647	10651	10652	10655	10662	10665	10666	10670	10671
10675	10676	10684	10687	10689	10690	10694	10695	10699	10700	10702	10711	10716

CKDHAO KEF11-B CIS DIAGNOSTIC
CKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 K 1 PAGE 427
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0426

10724	10727	10729	10736	10741	10751	10752	10755	10759	10760	10764	10765	10768
10769	10777	10778	10782	10784	10785	10789	10790	10793	10794	10802	10803	10807
10809	10810	10814	10815	10818	10819	10827	10830	10831	10835	10836	10840	10841
10849	10852	10854	10855	10859	10860	10864	10865	10867	10876	10878	10884	10886
10888	10890	10892	10894	10896	10898	10907	10909	10910	10915	10917	10922	10924
10929	10931	10936	10938	10943	10945	10950	10952	10957	10965	10967	10969	10971
10973	10976	10981	11002	11004	11006	11011	11026	11029	11031	11042	11045	11047
11058	11061	11063	11083	11085	11086	11091	11098	11100	11102	11108	11110	11112
11114	11120	11128	11130	11132	11134	11136	11139	11144	11163	11165	11167	11173
11176	11197	11199	11201	11203	11205	11207	11209	11211	11213	11215	11221	11225
11227	11229	11231	11238	11242	11243	11250	11254	11256	11258	11260	11267	11271
11272	11279	11283	11285	11287	11289	11298	11302	11303	11311	11314	11316	11318
11321	11323	11325	11328	11330	11332	11334	11336	11343	11348	11356	11359	11361
11368	11373	11383	11384	11388	11389	11393	11394	11397	11404	11406	11407	11411
11412	11415	11422	11424	11425	11429	11430	11433	11441	11444	11445	11449	11450
11454	11455	11463	11466	11468	11469	11473	11474	11478	11479	11481	11490	11495
11498	11505	11508	11510	11517	11522	11532	11533	11537	11538	11542	11543	11546
11553	11555	11556	11560	11561	11564	11571	11573	11574	11578	11579	11582	11589
11592	11593	11597	11598	11602	11603	11611	11614	11616	11617	11621	11622	11626
11627	11629	11638	11643	11651	11654	11656	11663	11668	11678	11679	11682	11686
11687	11691	11692	11695	11696	11704	11705	11709	11711	11712	11716	11717	11720
11721	11729	11730	11734	11736	11737	11741	11742	11745	11746	11754	11757	11758
11762	11763	11767	11768	11776	11779	11781	11782	11786	11787	11791	11792	11794
11803	11805	11811	11813	11815	11817	11819	11821	11823	11825	11834	11836	11837
11842	11844	11849	11851	11856	11858	11863	11865	11870	11872	11877	11879	11884
11893	11895	11897	11899	11901	11904	11909	11930	11932	11934	11939	11954	11957
11959	11970	11973	11975	11986	11989	11991	12011	12013	12014	12019	12026	12028
12030	12036	12038	12040	12042	12048	12056	12058	12060	12062	12064	12067	12072
12091	12093	12095	12102	12104	12106	12113	12115	12117	12119	12121	12129	12131
12133	12134	12137	12143	12147	12151	12152	12156	12158	12160	12161	12164	12170
12175	12182	12184	12188	12189	12192	12194	12198	12199	12202	12204	12208	12209
12211	12213	12215	12217	12220	12223	12227	12228	12230	12234	12235	12237	12241
12242	12245	12247	12251	12252	12254	12256	12265	12267	12269	12271	12273	12277
12281	12284	12286	12290	12291	12294	12296	12300	12301	12303	12305	12309	12310
12312	12314	12316	12332	12336	12337	12341	12344	12346	12348	12353	12354	12358
12359	12363	12364	12367	12369	12370	12372	12375	12380	12381	12385	12386	12390
12391	12394	12396	12397	12399	12401	12404	12408	12409	12413	12414	12418	12419
12422	12424	12425	12427	12429	12434	12435	12439	12440	12444	12445	12447	12449
12453	12454	12458	12459	12463	12464	12468	12470	12471	12474	12478	12482	12483
12487	12490	12492	12494	12499	12500	12504	12505	12509	12510	12513	12515	12516
12518	12523	12524	12528	12529	12533	12534	12537	12539	12540	12542	12546	12547
12551	12552	12556	12557	12560	12562	12563	12565	12570	12571	12575	12576	12580
12581	12583	12585	12589	12590	12594	12595	12599	12600	12604	12606	12607	12616
12618	12621	12623	12625	12627	12636	12638	12639	12644	12646	12651	12653	12658
12660	12665	12667	12672	12674	12679	12681	12686	12689	12691	12696	12699	12701
12706	12710	12712	12714	12719	12728	12730	12732	12734	12736	12738	12746	12748
12749	12754	12756	12758	12760	12766	12768	12770	12772	12778	12809	12811	12813
12815	12820	12822	12829	12831	12833	12835	12844	12846	12848	12852	12854	12866
12867	12871	12872	12875	12879	12881	12885	12886	12891	12895	12896	12899	12905
12909	12911	12913	12918	12919	12925	12930	12933	12935	12937	12938	12941	12945
12946	12950	12951	12953	12955	12959	12960	12963	12965	12969	12970	12972	12976
12977	12980	12982	12986	12988	12992	12993	12995	13000	13005	13006	13010	13011
13013	13017	13019	13020	13023	13029	13033	13038	13039	13042	13048	13053	13055
13057	13061	13063	13067	13069	13071	13073	13077	13078	13084	13087	13088	13090
13106	13109	13114	13118	13122	13125	13127	13128	13131	13135	13138	13140	13141
13144	13148	13149	13154	13155	13159	13160	13164	13165	13167	13170	13171	13176

13177	13179	13181	13186	13187	13191	13192	13194	13196	13198	13202	13203	13205
13207	13209	13213	13214	13219	13221	13222	13224	13229	13231	13232	13234	13236
13239	13242	13244	13248	13249	13253	13254	13258	13259	13263	13264	13266	13268
13273	13274	13276	13278	13283	13284	13288	13289	13291	13293	13295	13299	13300
13303	13305	13307	13308	13310	13312	13315	13318	13322	13323	13327	13328	13330
13332	13335	13337	13339	13344	13345	13349	13350	13352	13354	13356	13360	13361
13364	13366	13368	13369	13371	13373	13381	13382	13386	13387	13391	13392	13399
13400	13402	13407	13408	13412	13413	13415	13417	13419	13423	13424	13426	13430
13431	13433	13435	13440	13442	13445	13447	13451	13455	13456	13460	13461	13463
13465	13470	13471	13475	13476	13478	13480	13482	13486	13487	13489	13493	13494
13496	13505	13507	13510	13512	13514	13516	13525	13527	13528	13533	13535	13537
13539	13545	13549	13550	13555	13557	13562	13566	13568	13570	13572	13574	13576
13579	13583	13584	13587	13591	13596	13604	13607	13613	13618	13620	13624	13626
13627	13630	13635	13647	13649	13651	13653	13655	13657	13659	13661	13663	13665
13667	13669	13679	13681	13682	13687	13689	13691	13693	13699	13701	13703	13705
13711	13715	13716	13721	13723	13728	13731	13733	13735	13738	13744	13745	13799
13800	13803	13808	13810	13813	13815	13817	13819	13832	13834	13836	13842	13845
13866	13868	13870	13872	13874	13876	13878	13880	13887	13925	13927	13930	13932
13935	13937	13945	13947	13956	13958	13959	13964	13966	13971	13973	13978	13980
13985	13987	13992	13994	13999	14001	14006	14014	14016	14018	14020	14022	14025
14030	14036	14038	14040	14045	14060	14062	14064	14066	14068	14070	14072	14079
14124	14126	14127	14132	14139	14141	14143	14149	14151	14153	14155	14161	14169
14171	14174	14178	14180	14182	14184	14186	14189	14194	14213	14215	14217	14223
14226	14247	14249	14251	14253	14255	14257	14259	14261	14263	14265	14272	14311
14313	14316	14318	14321	14323	14331	14333	14342	14344	14345	14350	14352	14357
14359	14364	14366	14371	14373	14378	14380	14385	14387	14392	14401	14403	14405
14407	14409	14412	14417	14423	14425	14427	14432	14447	14449	14451	14453	14455
14457	14459	14461	14463	14470	14512	14514	14515	14520	14527	14529	14531	14537
14539	14541	14543	14549	14557	14559	14562	14566	14568	14570	14572	14574	14577
14582	14601	14603	14605	14612	14614	14616	14623	14625	14627	14629	14631	14640
14642	14644	14645	14648	14654	14659	14662	14664	14666	14667	14670	14676	14681
14688	14690	14694	14695	14698	14700	14704	14705	14708	14710	14714	14715	14717
14719	14721	14723	14726	14729	14733	14734	14736	14740	14741	14743	14747	14748
14751	14753	14757	14758	14760	14762	14771	14773	14775	14777	14779	14783	14787
14790	14792	14796	14797	14800	14802	14806	14807	14809	14811	14815	14816	14818
14820	14822	14836	14839	14841	14842	14845	14849	14850	14854	14855	14857	14860
14864	14866	14867	14870	14874	14878	14880	14882	14887	14890	14894	14895	14899
14900	14903	14904	14908	14911	14913	14914	14917	14921	14922	14926	14927	14929
14932	14936	14938	14939	14942	14946	14950	14952	14954	14959	14962	14966	14967
14971	14972	14975	14976	14985	14987	14990	14992	14994	14996	15005	15007	15008
15013	15015	15017	15019	15025	15027	15029	15031	15037	15040	15042	15044	15049
15059	15061	15063	15065	15067	15069	15077	15079	15080	15085	15087	15089	15091
15097	15099	15101	15103	15109	15140	15142	15144	15146	15153	15160	15162	15164
15166	15175	15177	15178	15181	15185	15186	15190	15191	15193	15196	15200	15202
15203	15206	15210	15212	15216	15220	15221	15225	15226	15228	15238	15239	15243
15244	15247	15251	15253	15257	15258	15264	15268	15269	15271	15276	15277	15281
15282	15287	15289	15294	15295	15297	15299	15301	15309	15311	15315	15317	15319
15320	15323	15327	15328	15332	15334	15336	15340	15342	15343	15345	15347	15351
15353	15355	15359	15361	15362	15365	15367	15370	15374	15375	15377	15381	15382
15385	15386	15390	15391	15394	15396	15400	15402	15406	15407	15409	15411	15413
15415	15422	15427	15428	15432	15433	15435	15439	15441	15442	15447	15451	15455
15456	15458	15460	15463	15465	15467	15470	15474	15476	15477	15481	15482	15485
15487	15489	15493	15495	15497	15508	15510	15513	15515	15517	15519	15524	15526
15527	15532	15534	15536	15542	15544	15546	15552	15556	15557	15562	15564	15569
15574	15576	15578	15580	15582	15584	15587	15591	15592	15595	15599	15604	15607
15609	15613	15615	15616	15619	15624	15636	15638	15640	15652	15654	15655	15660

15662	15664	15666	15672	15674	15676	15678	15684	15688	15689	15694	15696	15701
15704	15706	15708	15711	15715	15718	15771	15772	15775	15780	15782	15785	15787
15789	15791	15805	15807	15809	15815	15818	15839	15841	15843	15845	15847	15849
15851	15853	15860	15898	15900	15903	15905	15908	15910	15918	15920	15929	15931
15932	15937	15939	15944	15946	15951	15953	15958	15960	15965	15967	15972	15974
15979	15988	15990	15992	15994	15996	15999	16004	16010	16012	16014	16019	16034
16036	16038	16040	16042	16044	16046	16053	16096	16098	16099	16104	16111	16113
16115	16121	16123	16125	16127	16133	16141	16143	16146	16150	16152	16154	16156
16158	16161	16166	16204	16205	16208	16213	16215	16218	16220	16222	16224	16238
16240	16242	16248	16251	16272	16274	16276	16278	16280	16282	16284	16286	16288
16290	16297	16336	16338	16341	16343	16346	16348	16356	16358	16367	16369	16370
16375	16377	16382	16384	16389	16391	16396	16398	16403	16405	16410	16412	16417
16426	16428	16430	16432	16434	16437	16442	16448	16450	16452	16457	16472	16474
16476	16478	16480	16482	16484	16486	16488	16495	16538	16540	16541	16546	16553
16555	16557	16563	16565	16567	16569	16575	16583	16585	16588	16592	16594	16596
16598	16600	16603	16608	16859	16861	16865	16867	16870	16980	16982	16984	16986
16988	16990	17009	17011	17013	17015	17019	17021	17026	17028	17046	17048	17050
17052	17056	17058	17063	17065	17083	17085	17087	17089	17093	17095	17100	17102
17118	17120	17122	17124	17128	17130	17135	17137	17156	17158	17160	17162	17166
17168	17173	17175	17192	17194	17196	17198	17200	17202	17219	17221	17223	17225
17227	17229	17255	17262	17266	17267	17270	17274	17275	17277	17284	17288	17290
17292	17294	17296	17298	17300	17302	17305	17307	17309	17311	17313	17315	17317
17319	17320	17327	17331	17335	17339	17340	17391	17393	17396	17402	17406	17407
17411	17412	17422	17426	17427	17430	17434	17436	17437	17440	17444	17448	17450
17452	17457	17461	17463	17464								
16898#												
17814#	18030											
1024#												
1041#	17491*	17492	17494*									
720#	1267#	1270#	1275#	1277#	1312#	1330#	1343#	1346#	1351#	1356#	1361#	1363#
1365#	1370#	1372#	1377#	1379#	1384#	1386#	1391#	1393#	1398#	1400#	1403#	1408#
1413#	1418#	1420#	1443#	1453#	1455#	1460#	1462#	1467#	1469#	1474#	1476#	1481#
1483#	1488#	1532#	1550#	1563#	1566#	1571#	1576#	1581#	1583#	1585#	1590#	1592#
1597#	1599#	1604#	1606#	1611#	1613#	1618#	1620#	1625#	1651#	1656#	1658#	1663#
1665#	1670#	1672#	1677#	1679#	1684#	1686#	1691#	1693#	1698#	1760#	1770#	1778#
1792#	1806#	1814#	1824#	1826#	1830#	1838#	1848#	1850#	1852#	1862#	1880#	1885#
1887#	1892#	1894#	1899#	1901#	1906#	1908#	1913#	1915#	1920#	1922#	1927#	1940#
1951#	1956#	1974#	1979#	1994#	1999#	2009#	2013#	2016#	2026#	2036#	2041#	2049#
2056#	2061#	2099#	2101#	2105#	2107#	2109#	2114#	2116#	2121#	2123#	2128#	2130#
2135#	2137#	2142#	2144#	2149#	2151#	2156#	2158#	2163#	2172#	2177#	2376#	2386#
2394#	2408#	2422#	2430#	2440#	2442#	2446#	2454#	2464#	2466#	2486#	2491#	2493#
2498#	2500#	2505#	2507#	2512#	2514#	2519#	2521#	2526#	2528#	2533#	2544#	2559#
2564#	2576#	2581#	2608#	2613#	2618#	2620#	2628#	2633#	2644#	2651#	2656#	2694#
2696#	2700#	2702#	2704#	2709#	2711#	2716#	2718#	2723#	2725#	2730#	2732#	2737#
2739#	2744#	2746#	2751#	2753#	2758#	2766#	2771#	2889#	2903#	2917#	2925#	2935#
2937#	2941#	2949#	2959#	2961#	3007#	3012#	3014#	3019#	3021#	3026#	3028#	3033#
3035#	3040#	3042#	3047#	3049#	3054#	3071#	3082#	3087#	3105#	3111#	3126#	3131#
3141#	3143#	3155#	3160#	3168#	3175#	3180#	3220#	3222#	3226#	3228#	3230#	3235#
3237#	3242#	3244#	3249#	3251#	3256#	3258#	3263#	3265#	3270#	3272#	3277#	3279#
3284#	3293#	3298#	3396#	3413#	3415#	3432#	3473#	3478#	3480#	3485#	3487#	3492#
3494#	3499#	3501#	3506#	3508#	3513#	3515#	3520#	3527#	3532#	3561#	3565#	3569#
3574#	3576#	3581#	3583#	3588#	3590#	3595#	3597#	3602#	3604#	3609#	3682#	3699#
3701#	3718#	3759#	3764#	3766#	3771#	3773#	3778#	3780#	3785#	3787#	3792#	3794#
3799#	3801#	3806#	3813#	3818#	3847#	3851#	3856#	3861#	3863#	3868#	3870#	3875#
3877#	3882#	3884#	3889#	3891#	3896#	3966#	3979#	3981#	3998#	4055#	4060#	4062#
4067#	4069#	4074#	4076#	4081#	4083#	4088#	4090#	4095#	4097#	4102#	4109#	4114#

\$GET42 056422
\$GTSWR 061164
\$MIBTS 000300
\$ICAT 000406
\$IFLEV= 177777

UKDHAO KEF11-B CIS DIAGNOSTIC
UKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 430
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0429

4159#	4163#	4167#	4172#	4174#	4179#	4181#	4186#	4188#	4193#	4195#	4200#	4202#
4207#	4279#	4292#	4294#	4311#	4365#	4370#	4372#	4377#	4379#	4384#	4386#	4391#
4393#	4398#	4400#	4405#	4407#	4412#	4419#	4424#	4466#	4470#	4474#	4479#	4481#
4486#	4488#	4493#	4495#	4500#	4502#	4507#	4509#	4514#	4590#	4599#	4614#	4630#
4655#	4663#	4665#	4673#	4675#	4682#	4700#	4705#	4707#	4712#	4714#	4719#	4721#
4726#	4728#	4733#	4735#	4740#	4742#	4747#	4754#	4759#	4794#	4799#	4801#	4806#
4808#	4813#	4815#	4820#	4822#	4827#	4829#	4834#	4836#	4841#	4934#	4947#	4962#
4979#	4998#	5004#	5021#	5023#	5047#	5052#	5054#	5059#	5061#	5066#	5068#	5073#
5075#	5080#	5082#	5087#	5089#	5094#	5101#	5106#	5143#	5147#	5149#	5154#	5156#
5161#	5163#	5168#	5170#	5175#	5177#	5182#	5184#	5189#	5191#	5196#	5238#	5241#
5246#	5248#	5392#	5408#	5413#	5415#	5420#	5422#	5427#	5429#	5434#	5436#	5441#
5443#	5448#	5450#	5455#	5480#	5485#	5500#	5505#	5515#	5526#	5531#	5539#	5541#
5570#	5580#	5625#	5631#	5678#	5681#	5686#	5688#	5748#	5755#	5760#	5762#	5764#
5767#	5784#	5793#	5795#	5808#	5810#	5829#	5832#	5835#	5842#	5864#	5870#	5879#
5893#	5895#	5897#	5899#	5914#	5928#	5959#	5964#	5966#	5971#	5973#	5978#	5980#
5985#	5987#	5992#	5994#	5999#	6001#	6006#	6026#	6031#	6041#	6046#	6067#	6072#
6074#	6079#	6081#	6086#	6088#	6093#	6095#	6100#	6102#	6107#	6109#	6114#	6118#
6123#	6127#	6132#	6137#	6142#	6148#	6153#	6280#	6285#	6290#	6292#	6295#	6307#
6309#	6311#	6314#	6326#	6328#	6330#	6332#	6354#	6356#	6367#	6384#	6386#	6394#
6406#	6417#	6419#	6425#	6435#	6440#	6447#	6451#	6453#	6459#	6465#	6467#	6479#
6481#	6483#	6498#	6512#	6543#	6548#	6550#	6555#	6557#	6562#	6564#	6569#	6571#
6576#	6578#	6583#	6585#	6590#	6610#	6615#	6625#	6630#	6651#	6656#	6658#	6663#
6665#	6670#	6672#	6677#	6679#	6684#	6686#	6691#	6693#	6698#	6702#	6707#	6713#
6718#	6830#	6844#	6866#	6885#	6894#	6942#	6948#	6960#	6965#	6967#	6975#	6977#
6979#	6981#	7013#	7018#	7020#	7025#	7027#	7032#	7034#	7039#	7041#	7046#	7048#
7053#	7055#	7060#	7068#	7073#	7093#	7098#	7100#	7105#	7107#	7112#	7114#	7119#
7121#	7126#	7128#	7133#	7135#	7140#	7142#	7147#	7149#	7154#	7255#	7269#	7291#
7318#	7324#	7327#	7338#	7345#	7347#	7349#	7355#	7367#	7384#	7391#	7396#	7398#
7404#	7406#	7412#	7419#	7421#	7423#	7442#	7447#	7455#	7457#	7489#	7494#	7496#
7501#	7503#	7508#	7510#	7515#	7517#	7522#	7524#	7529#	7531#	7536#	7544#	7549#
7569#	7574#	7576#	7586#	7588#	7598#	7600#	7605#	7607#	7612#	8006#	8020#	8063#
8074#	8084#	8086#	8095#	8105#	8165#	8180#	8205#	8212#	8221#	8229#	8237#	8242#
8295#	8300#	8307#	8315#	8319#	8324#	8330#	8340#	8342#	8344#	8348#	8351#	8359#
8363#	8367#	8370#	8377#	8379#	8382#	8389#	8397#	8402#	8406#	8408#	8412#	8415#
8421#	8426#	8430#	8432#	8436#	8442#	8452#	8454#	8458#	8466#	8476#	8478#	8508#
8513#	8515#	8520#	8522#	8527#	8529#	8534#	8536#	8541#	8543#	8548#	8550#	8555#
8571#	8576#	8592#	8597#	8617#	8622#	8624#	8634#	8636#	8646#	8648#	8653#	8655#
8660#	8662#	8667#	8669#	8674#	8855#	8858#	8863#	8873#	8875#	8877#	8882#	8885#
8892#	8894#	8899#	8902#	8909#	8914#	8918#	8920#	8925#	8928#	8933#	8938#	8942#
8944#	8949#	8952#	8962#	8964#	8969#	8972#	8982#	8984#	9016#	9024#	9037#	9039#
9041#	9046#	9054#	9059#	9066#	9068#	9076#	9084#	9092#	9097#	9106#	9116#	9142#
9153#	9177#	9190#	9237#	9242#	9244#	9249#	9251#	9256#	9258#	9263#	9265#	9270#
9272#	9277#	9279#	9284#	9300#	9305#	9315#	9320#	9341#	9346#	9348#	9358#	9360#
9370#	9372#	9377#	9379#	9384#	9386#	9391#	9553#	9557#	9562#	9567#	9603#	9605#
9613#	9620#	9625#	9627#	9651#	9662#	9686#	9697#	9721#	9732#	9756#	9767#	9791#
9802#	10152#	10159#	10164#	10166#	10210#	10213#	10218#	10220#	10296#	10318#	10325#	10347#
10354#	10378#	10418#	10423#	10459#	10469#	10482#	10487#	10500#	10505#	10520#	10530#	10544#
10554#	10565#	10570#	10606#	10616#	10629#	10634#	10647#	10652#	10666#	10676#	10690#	10700#
10711#	10716#	10752#	10755#	10765#	10769#	10778#	10785#	10790#	10794#	10803#	10810#	10815#
10819#	10831#	10841#	10855#	10865#	10910#	10915#	10917#	10922#	10924#	10929#	10931#	10936#
10938#	10943#	10945#	10950#	10952#	10957#	10976#	10981#	11006#	11011#	11086#	11091#	11098#
11108#	11110#	11120#	11139#	11144#	11221#	11243#	11250#	11272#	11279#	11303#	11343#	11348#
11384#	11394#	11407#	11412#	11425#	11430#	11445#	11455#	11469#	11479#	11490#	11495#	11533#
11543#	11556#	11561#	11574#	11579#	11593#	11603#	11617#	11627#	11638#	11643#	11679#	11682#
11692#	11696#	11705#	11712#	11717#	11721#	11730#	11737#	11742#	11746#	11758#	11768#	11782#
11792#	11837#	11842#	11844#	11849#	11851#	11856#	11858#	11863#	11865#	11870#	11872#	11877#

11879#	11884#	11904#	11909#	11934#	11939#	12014#	12019#	12026#	12036#	12038#	12048#	12067#
12072#	12147#	12152#	12182#	12192#	12202#	12209#	12211#	12213#	12215#	12220#	12223#	12228#
12235#	12245#	12252#	12254#	12256#	12281#	12284#	12294#	12301#	12303#	12310#	12312#	12314#
12316#	12332#	12337#	12348#	12354#	12364#	12370#	12372#	12375#	12381#	12391#	12397#	12399#
12401#	12404#	12409#	12419#	12425#	12427#	12429#	12435#	12445#	12447#	12449#	12454#	12464#
12471#	12478#	12483#	12494#	12500#	12510#	12516#	12518#	12524#	12534#	12540#	12542#	12547#
12557#	12563#	12565#	12571#	12581#	12583#	12585#	12590#	12600#	12607#	12639#	12644#	12646#
12651#	12653#	12658#	12660#	12665#	12667#	12672#	12674#	12679#	12681#	12686#	12691#	12696#
12701#	12706#	12714#	12719#	12749#	12754#	12756#	12766#	12768#	12778#	12867#	12872#	12879#
12886#	12891#	12909#	12911#	12925#	12938#	12941#	12951#	12953#	12955#	12960#	12970#	12977#
12986#	12993#	12995#	13000#	13006#	13011#	13013#	13067#	13069#	13071#	13073#	13078#	13109#
13114#	13118#	13128#	13131#	13141#	13144#	13149#	13155#	13165#	13167#	13171#	13177#	13192#
13194#	13203#	13205#	13207#	13209#	13214#	13222#	13224#	13232#	13234#	13236#	13239#	13242#
13249#	13254#	13264#	13266#	13268#	13274#	13289#	13291#	13300#	13308#	13310#	13312#	13315#
13318#	13328#	13330#	13332#	13335#	13350#	13352#	13361#	13369#	13371#	13373#	13382#	13392#
13400#	13413#	13415#	13424#	13426#	13431#	13433#	13435#	13451#	13461#	13463#	13476#	13478#
13487#	13489#	13494#	13496#	13528#	13533#	13535#	13545#	13550#	13555#	13557#	13562#	13579#
13584#	13591#	13596#	13620#	13627#	13630#	13635#	13682#	13687#	13689#	13699#	13701#	13711#
13716#	13721#	13723#	13728#	13738#	13745#	13800#	13803#	13808#	13810#	13959#	13964#	13966#
13971#	13973#	13978#	13980#	13985#	13987#	13992#	13994#	13999#	14001#	14006#	14025#	14030#
14040#	14045#	14127#	14132#	14139#	14149#	14151#	14161#	14189#	14194#	14345#	14350#	14352#
14357#	14359#	14364#	14366#	14371#	14373#	14378#	14380#	14385#	14387#	14392#	14412#	14417#
14427#	14432#	14515#	14520#	14527#	14537#	14539#	14549#	14577#	14582#	14688#	14698#	14708#
14715#	14717#	14719#	14721#	14726#	14729#	14734#	14741#	14751#	14758#	14760#	14762#	14787#
14790#	14800#	14807#	14809#	14816#	14818#	14820#	14822#	14842#	14845#	14855#	14860#	14867#
14890#	14900#	14904#	14914#	14917#	14927#	14932#	14939#	14962#	14972#	14976#	15008#	15013#
15015#	15025#	15027#	15037#	15044#	15049#	15080#	15085#	15087#	15097#	15099#	15109#	15178#
15181#	15191#	15196#	15203#	15216#	15226#	15228#	15239#	15244#	15251#	15258#	15269#	15282#
15287#	15295#	15297#	15320#	15323#	15343#	15345#	15347#	15362#	15370#	15375#	15382#	15391#
15400#	15407#	15409#	15411#	15422#	15428#	15433#	15435#	15470#	15482#	15493#	15495#	15497#
15527#	15532#	15534#	15542#	15544#	15552#	15557#	15562#	15564#	15569#	15587#	15592#	15599#
15604#	15609#	15616#	15619#	15624#	15655#	15660#	15662#	15672#	15674#	15684#	15689#	15694#
15696#	15701#	15711#	15718#	15722#	15775#	15780#	15782#	15932#	15937#	15939#	15944#	15946#
15951#	15953#	15958#	15960#	15965#	15967#	15972#	15974#	15979#	15999#	16004#	16014#	16019#
16099#	16104#	16111#	16121#	16123#	16133#	16161#	16166#	16205#	16208#	16213#	16215#	16370#
16375#	16377#	16382#	16384#	16389#	16391#	16396#	16398#	16403#	16405#	16410#	16412#	16417#
16437#	16442#	16452#	16457#	16541#	16546#	16553#	16563#	16565#	16575#	16603#	16608#	16859#
16870#	17262#	17270#	17275#	17277#	17284#	17320#	17327#	17340#	17402#	17412#	17422#	17430#
17437#	17464#											
17510	17526	17545#										
1049#	17846											
1267#	1277	1312#	1330	1343#	1363	1365#	1370	1372#	1377	1379#	1384	1386#
1391	1393#	1398	1400#	1420	1448#	1453	1455#	1460	1462#	1467	1469#	1474
1476#	1481	1483#	1488	1532#	1550	1563#	1583	1585#	1590	1592#	1597	1599#
1604	1606#	1611	1613#	1618	1620#	1625	1651#	1656	1658#	1663	1665#	1670
1672#	1677	1679#	1684	1686#	1691	1693#	1698	1760#	1770	1778#	1792	1806#
1826	1830#	1850	1852#	1862	1880#	1885	1887#	1892	1894#	1899	1901#	1906
1908#	1913	1915#	1920	1922#	1927	1940#	2009	2013#	2049	2056#	2061	2099#
2107	2109#	2114	2116#	2121	2123#	2128	2130#	2135	2137#	2142	2144#	2149
2151#	2156	2158#	2163	2172#	2177	2376#	2386	2394#	2408	2422#	2442	2446#
2466	2486#	2491	2493#	2498	2500#	2505	2507#	2512	2514#	2519	2521#	2526
2528#	2533	2544#	2618	2620#	2644	2651#	2656	2694#	2702	2704#	2709	2711#
2716	2718#	2723	2725#	2730	2732#	2737	2739#	2744	2746#	2751	2753#	2758
2766#	2771	2889#	2903	2917#	2937	2941#	2961	3007#	3012	3014#	3019	3021#
3026	3028#	3033	3035#	3040	3042#	3047	3049#	3054	3071#	3141	3143#	3168
3175#	3180	3220#	3228	3230#	3235	3237#	3242	3244#	3249	3251#	3256	3258#

\$ILLUP 060032
 \$INTAG 000423
 \$ISKO = 000001

3263	3265#	3270	3272#	3277	3279#	3284	3293#	3298	3396#	3413	3415#	3432
3473#	3478	3480#	3485	3487#	3492	3494#	3499	3501#	3506	3508#	3513	3515#
3520	3527#	3532	3561#	3565	3569#	3574	3576#	3581	3583#	3588	3590#	3595
3597#	3602	3604#	3609	3682#	3699	3701#	3718	3759#	3764	3766#	3771	3773#
3778	3780#	3785	3787#	3792	3794#	3799	3801#	3806	3813#	3818	3847#	3851
3856#	3861	3863#	3868	3870#	3875	3877#	3882	3884#	3889	3891#	3896	3966#
3979	3981#	3998	4055#	4060	4062#	4067	4069#	4074	4076#	4081	4083#	4088
4090#	4095	4097#	4102	4109#	4114	4159#	4163	4167#	4172	4174#	4179	4181#
4186	4188#	4193	4195#	4200	4202#	4207	4279#	4292	4294#	4311	4365#	4370
4372#	4377	4379#	4384	4386#	4391	4393#	4398	4400#	4405	4407#	4412	4419#
4424	4466#	4470	4474#	4479	4481#	4486	4488#	4493	4495#	4500	4502#	4507
4509#	4514	4590#	4599	4614#	4630	4655#	4663	4665#	4673	4675#	4682	4700#
4705	4707#	4712	4714#	4719	4721#	4726	4728#	4733	4735#	4740	4742#	4747
4754#	4759	4794#	4799	4801#	4806	4808#	4813	4815#	4820	4822#	4827	4829#
4834	4836#	4841	4934#	4947	4962#	4979	4998#	5023	5027#	5052	5054#	5059
5061#	5066	5068#	5073	5075#	5080	5082#	5087	5089#	5094	5101#	5106	5143#
5147	5149#	5154	5156#	5161	5163#	5168	5170#	5175	5177#	5182	5184#	5189
5191#	5196	5238#	5248	5392#	5541	5570#	5580	5625#	5631	5678#	5688	5748#
5762	5764#	5899	5914#	5928	5959#	5964	5966#	5971	5973#	5978	5980#	5985
5987#	5992	5994#	5999	6001#	6006	6026#	6031	6041#	6046	6067#	6072	6074#
6079	6081#	6086	6088#	6093	6095#	6100	6102#	6107	6109#	6114	6118#	6123
6127#	6132	6137#	6142	6148#	6153	6280#	6330	6332#	6354	6356#	6483	6498#
6512	6543#	6548	6550#	6555	6557#	6562	6564#	6569	6571#	6576	6578#	6583
6585#	6590	6610#	6615	6625#	6630	6651#	6656	6658#	6663	6665#	6670	6672#
6677	6679#	6684	6686#	6691	6693#	6698	6702#	6707	6713#	6718	6830#	6844
6866#	6981	7013#	7018	7020#	7025	7027#	7032	7034#	7039	7041#	7046	7048#
7053	7055#	7060	7068#	7073	7093#	7098	7100#	7105	7107#	7112	7114#	7119
7121#	7126	7128#	7133	7135#	7140	7142#	7147	7149#	7154	7255#	7269	7291#
7318	7324#	7423	7442#	7457	7489#	7494	7496#	7501	7503#	7508	7510#	7515
7517#	7522	7524#	7529	7531#	7536	7544#	7549	7569#	7574	7576#	7586	7588#
7598	7600#	7605	7607#	7612	8006#	8020	8063#	8086	8095#	8105	8165#	8180
8205#	8212	8221#	8229	8237#	8242	8295#	8300	8307#	8315	8319#	8344	8348#
8363	8367#	8408	8412#	8432	8436#	8454	8458#	8478	8508#	8513	8515#	8520
8522#	8527	8529#	8534	8536#	8541	8543#	8548	8550#	8555	8571#	8576	8592#
8597	8617#	8622	8624#	8634	8636#	8646	8648#	8653	8655#	8660	8662#	8667
8669#	8674	8855#	8877	8882#	8894	8899#	8920	8925#	8944	8949#	8964	8969#
8984	9016#	9039	9041#	9046	9054#	9068	9076#	9084	9092#	9097	9106#	9116
9142#	9153	9177#	9190	9237#	9242	9244#	9249	9251#	9256	9258#	9263	9265#
9270	9272#	9277	9279#	9284	9300#	9305	9315#	9320	9341#	9346	9348#	9358
9360#	9370	9372#	9377	9379#	9384	9386#	9391	9553#	9557	9562#	9605	9613#
9627	9651#	9662	9686#	9697	9721#	9732	9756#	9767	9791#	9802	10152#	10166
10210#	10220	10296#	10318	10325#	10347	10354#	10378	10418#	10423	10459#	10469	10482#
10487	10500#	10505	10520#	10530	10544#	10554	10565#	10570	10606#	10616	10629#	10634
10647#	10652	10666#	10676	10690#	10700	10711#	10716	10752#	10769	10778#	10794	10803#
10819	10831#	10841	10855#	10865	10910#	10915	10917#	10922	10924#	10929	10931#	10936
10938#	10943	10945#	10950	10952#	10957	10976#	10981	11006#	11011	11086#	11091	11098#
11108	11110#	11120	11139#	11144	11221#	11243	11250#	11272	11279#	11303	11343#	11348
11384#	11394	11407#	11412	11425#	11430	11445#	11455	11469#	11479	11490#	11495	11533#
11543	11556#	11561	11574#	11579	11593#	11603	11617#	11627	11638#	11643	11679#	11696
11705#	11721	11730#	11746	11758#	11768	11782#	11792	11837#	11842	11844#	11849	11851#
11856	11858#	11863	11865#	11870	11872#	11877	11879#	11884	11904#	11909	11934#	11939
12014#	12019	12026#	12036	12038#	12048	12067#	12072	12147#	12152	12182#	12213	12215#
12316	12332#	12337	12348#	12370	12372#	12399	12401#	12427	12429#	12447	12449#	12471
12478#	12483	12494#	12516	12518#	12540	12542#	12563	12565#	12583	12585#	12607	12639#
12644	12646#	12651	12653#	12658	12660#	12665	12667#	12672	12674#	12679	12681#	12686
12691#	12696	12701#	12706	12714#	12719	12749#	12754	12756#	12766	12768#	12778	12867#

\$ISK1 = 000001

12872	12879#	12886	12891#	12909	12911#	13071	13073#	13078	13109#	13114	13118#	13128
13131#	13141	13144#	13207	13209#	13234	13236#	13310	13312#	13371	13373#	13433	13435#
13496	13528#	13533	13535#	13545	13550#	13555	13557#	13562	13579#	13584	13591#	13596
13620#	13627	13630#	13635	13682#	13687	13689#	13699	13701#	13711	13716#	13721	13723#
13728	13738#	13745	13800#	13810	13959#	13964	13966#	13971	13973#	13978	13980#	13985
13987#	13992	13994#	13999	14001#	14006	14025#	14030	1404#	14045	14127#	14132	14139#
14149	14151#	14161	14189#	14194	14345#	14350	14352#	14357	14359#	14364	14366#	14371
14373#	14378	14380#	14385	14387#	14392	14412#	14417	14427#	14432	14515#	14520	14527#
14537	14539#	14549	14577#	14582	14688#	14719	14721#	14822	14842#	14904	14914#	14976
15008#	15013	15015#	15025	15027#	15037	15044#	15049	15080#	15085	15087#	15097	15099#
15109	15178#	15228	15239#	15244	15251#	15258	15269#	15282	15287#	15295	15297#	15409
15411#	15497	15527#	15532	15534#	15542	15544#	15552	15557#	15562	15564#	15569	15587#
15592	15599#	15604	15609#	15616	15619#	15624	15655#	15660	15662#	15672	15674#	15684
15689#	15694	15696#	15701	15711#	15718	15772#	15782	15932#	15937	15939#	15944	15946#
15951	15953#	15958	15960#	15965	15967#	15972	15974#	15979	15999#	16004	16014#	16019
16099#	16104	16111#	16121	16123#	16133	16161#	16166	16205#	16215	16370#	16375	16377#
16382	16384#	16389	16391#	16396	16398#	16403	16405#	16410	16412#	16417	16437#	16442
16452#	16457	16541#	16546	16553#	16563	16565#	16575	16603#	16608	16859#	16870	17262#
17277	17284#	17320	17327#	17340	17402#	17412	17422#	17464				

\$ISK2 = 000001

1270#	1275	1346#	1351	1356#	1361	1403#	1408	1413#	1418	1566#	1571	1576#
1581	1814#	1824	1838#	1848	1951#	1956	1974#	1979	1994#	1999	2016#	2026
2036#	2041	2101#	2105	2430#	2440	2454#	2464	2559#	2564	2576#	2581	2608#
2613	2628#	2633	2696#	2700	2925#	2935	2949#	2959	3082#	3087	3105#	3111
3126#	3131	3155#	3160	3222#	3226	5004#	5021	5241#	5246	5408#	5413	5415#
5420	5422#	5427	5429#	5434	5436#	5441	5443#	5448	5450#	5455	5480#	5485
5500#	5505	5515#	5539	5681#	5686	5755#	5760	5767#	5897	6285#	6290	6292#
6309	6311#	6328	6367#	6384	6386#	6394	6406#	6417	6419#	6451	6453#	6481
6885#	6894	6942#	6979	7327#	7347	7349#	7396	7398#	7404	7406#	7421	7447#
7455	8074#	8084	8324#	8342	8351#	8359	8370#	8377	8379#	8406	8415#	8430
8442#	8452	8466#	8476	8858#	8875	8885#	8892	8902#	8918	8928#	8942	8952#
8962	8972#	8982	9024#	9037	9059#	9066	9567#	9603	9620#		10159#	10164
10213#	10218	10755#	10765	10785#	10790	10810#	10815	11682#	11692	11712#	11717	11737#
11742	12192#	12211	12220#	12314	12354#	12364	12375#	12397	12404#	12425	12435#	12445
12454#	12464	12500#	12510	12524#	12534	12547#	12557	12571#	12581	12590#	12600	12925#
12995	13000#	13069	13149#	13167	13171#	13205	13214#	13222	13224#	13232	13239#	13266
13268#	13308	13315#	13330	13332#	13369	13382#	13392	13400#	13413	13415#	13424	13426#
13431	13451#	13461	13463#	13476	13478#	13487	13489#	13494	13803#	13808	14698#	14717
14726#	14820	14845#	14855	14860#	14867	14890#	14900	14917#	14927	14932#	14939	14962#
14972	15181#	15191	15196#	15203	15216#	15226	15320#	15345	15347#	15362	15370#	15391
15400#	15407	15422#	15495	15775#	15780	16208#	16213	17270#	17275	17430#	17437	

\$ISK3 = 000001

5526#	5531	5784#	5793	5795#	5808	5810#	5829	5832#	5864	5870#	5895	6295#
6307	6314#	6326	6425#	6435	6440#	6447	6459#	6465	6467#	6479	6948#	6977
7338#	7345	7355#	7367	7384#	7391	7412#	7419	8330#	8340	8382#	8389	8397#
8402	8421#	8426	8863#	8873	8909#	8914	8933#	8938	12202#	12209	12223#	12228
12235#	12254	12256#	12312	12381#	12391	12409#	12419	12938#	12953	12955#	12960	12970#
12977	12986#	12993	13006#	13011	13013#	13067	13155#	13165	13177#	13192	13194#	13203
13242#	13249	13254#	13264	13274#	13289	13291#	13300	13318#	13328	13335#	13350	13352#
13361	14708#	14715	14729#	14734	14741#	14760	14762#	14818	15323#	15343	15375#	15382
15428#	15433	15435#	15493									
5835#	5842	5879#	5893	6960#	6965	6967#	6975	12245#	12252	12281#	12310	12941#
12951	14751#	14758	14787#	14816	15470#	15482						

\$ISK4 = 000001

\$ISK5 = 000001

\$LF 000444

\$LFLG 062151

\$LO - 000000

12284#	12303	14790#	14809									
12294#	12301	14800#	14807									
1060#	17923											
17980*	17986#											
10444#	10448	10454	10455#	10475#	10493#	10512#	10534#	10558#	10591#	10595	10601	10602#

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 434
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0433

10622#	10640#	10658#	10680#	10704#	10737#	10741	10747	10748#	10773#	10798#	10823#	10845#
10869#	11369#	11373	11379	11380#	11400#	11418#	11437#	11459#	11483#	11518#	11522	11528
11529#	11549#	11567#	11585#	11607#	11631#	11664#	11668	11674	11675#	11700#	11725#	11750#
11772#	11796#											
720#	1268	1269	1271	1272	1275	1276	1277	1278	1302	1303	1313	1314
1321	1322	1323	1330	1331	1344	1345	1347	1348	1351	1352	1353	1354
1355	1357	1358	1361	1362	1363	1364	1366	1367	1370	1371	1373	1374
1377	1378	1380	1381	1384	1385	1387	1388	1391	1392	1394	1395	1398
1399	1401	1402	1404	1405	1408	1409	1410	1411	1412	1414	1415	1418
1419	1420	1421	1425	1426	1449	1450	1453	1454	1456	1457	1460	1461
1463	1464	1467	1468	1470	1471	1474	1475	1477	1478	1481	1482	1484
1485	1488	1489	1522	1523	1533	1534	1541	1542	1543	1550	1551	1564
1565	1567	1568	1571	1572	1573	1574	1575	1577	1578	1581	1582	1583
1584	1586	1587	1590	1591	1593	1594	1597	1598	1600	1601	1604	1605
1607	1608	1611	1612	1614	1615	1618	1619	1621	1622	1625	1626	1630
1631	1652	1653	1656	1657	1659	1660	1663	1664	1666	1667	1670	1671
1673	1674	1677	1678	1680	1681	1684	1685	1687	1688	1691	1692	1694
1695	1698	1699	1761	1762	1765	1766	1767	1770	1771	1779	1780	1781
1782	1783	1786	1787	1788	1792	1793	1807	1808	1811	1812	1813	1815
1816	1819	1820	1821	1824	1825	1826	1827	1831	1832	1835	1836	1837
1839	1840	1843	1844	1845	1848	1849	1850	1851	1853	1854	1857	1858
1859	1862	1863	1881	1882	1885	1886	1888	1889	1892	1893	1895	1896
1899	1900	1902	1903	1906	1907	1909	1910	1913	1914	1916	1917	1920
1921	1923	1924	1927	1928	1941	1942	1945	1946	1947	1948	1949	1950
1952	1953	1956	1957	1960	1961	1962	1963	1964	1965	1968	1969	1970
1971	1972	1973	1975	1976	1979	1980	1983	1984	1985	1992	1993	1995
1996	1999	2000	2004	2005	2006	2007	2008	2009	2010	2014	2015	2017
2018	2021	2022	2023	2026	2027	2034	2035	2037	2038	2041	2042	2044
2045	2046	2047	2048	2049	2050	2057	2058	2061	2062	2099	2100	2101
2102	2105	2106	2107	2108	2110	2111	2114	2115	2117	2118	2121	2122
2124	2125	2128	2129	2131	2132	2135	2136	2138	2139	2142	2143	2145
2146	2149	2150	2152	2153	2156	2157	2159	2160	2163	2164	2170	2171
2173	2174	2177	2178	2180	2181	2182	2183	2184	2377	2378	2381	2382
2383	2386	2387	2395	2396	2397	2398	2399	2402	2403	2404	2408	2409
2423	2424	2427	2428	2429	2431	2432	2435	2436	2437	2440	2441	2442
2443	2447	2448	2451	2452	2453	2455	2456	2459	2460	2461	2464	2465
2466	2467	2487	2488	2491	2492	2494	2495	2498	2499	2501	2502	2505
2506	2508	2509	2512	2513	2515	2516	2519	2520	2522	2523	2526	2527
2529	2530	2533	2534	2545	2546	2553	2554	2555	2556	2557	2558	2560
2561	2564	2565	2568	2569	2570	2571	2572	2573	2574	2575	2577	2578
2581	2582	2588	2589	2590	2591	2592	2602	2603	2604	2605	2606	2607
2609	2610	2613	2614	2615	2616	2617	2618	2619	2621	2622	2626	2627
2629	2630	2633	2634	2639	2640	2641	2642	2643	2644	2645	2652	2653
2656	2657	2694	2695	2696	2697	2700	2701	2702	2703	2705	2706	2709
2710	2712	2713	2716	2717	2719	2720	2723	2724	2726	2727	2730	2731
2733	2734	2737	2738	2740	2741	2744	2745	2747	2748	2751	2752	2754
2755	2758	2759	2764	2765	2767	2768	2771	2772	2774	2775	2776	2777
2778	2890	2891	2892	2893	2894	2897	2898	2899	2903	2904	2918	2919
2922	2923	2924	2926	2927	2930	2931	2932	2935	2936	2937	2938	2942
2943	2946	2947	2948	2950	2951	2954	2955	2956	2959	2960	2961	2962
2971	2972	2973	2974	2975	2976	2977	2980	2981	2982	3008	3009	3012
3013	3015	3016	3019	3020	3022	3023	3026	3027	3029	3030	3033	3034
3036	3037	3040	3041	3043	3044	3047	3048	3050	3051	3054	3055	3072
3073	3076	3077	3078	3079	3080	3081	3083	3084	3087	3088	3091	3092
3093	3094	3095	3096	3099	3100	3101	3102	3103	3104	3106	3107	3111
3112	3115	3116	3117	3124	3125	3127	3128	3131	3132	3136	3137	3138

\$L OCTA= 000001

3139	3140	3141	3142	3144	3145	3153	3154	3156	3157	3160	3161	3163
3164	3165	3166	3167	3168	3169	3176	3177	3180	3181	3220	3221	3222
3223	3226	3227	3228	3229	3231	3232	3235	3236	3238	3239	3242	3243
3245	3246	3249	3250	3252	3253	3256	3257	3259	3260	3263	3264	3266
3267	3270	3271	3273	3274	3277	3278	3280	3281	3284	3285	3291	3292
3294	3295	3298	3299	3301	3302	3303	3304	3305	3396	3397	3404	3405
3406	3413	3414	3416	3417	3423	3424	3425	3432	3433	3474	3475	3478
3479	3481	3482	3485	3486	3488	3489	3492	3493	3495	3496	3499	3500
3502	3503	3506	3507	3509	3510	3513	3514	3516	3517	3520	3521	3528
3529	3532	3533	3561	3562	3565	3566	3570	3571	3574	3575	3577	3578
3581	3582	3584	3585	3588	3589	3591	3592	3595	3596	3598	3599	3602
3603	3605	3606	3609	3610	3682	3683	3690	3691	3692	3699	3700	3702
3703	3709	3710	3711	3718	3719	3760	3761	3764	3765	3767	3768	3771
3772	3774	3775	3778	3779	3781	3782	3785	3786	3788	3789	3792	3793
3795	3796	3799	3800	3802	3803	3806	3807	3814	3815	3818	3819	3847
3848	3851	3852	3857	3858	3861	3862	3864	3865	3868	3869	3871	3872
3875	3876	3878	3879	3882	3883	3885	3886	3889	3890	3892	3893	3896
3897	3966	3967	3972	3973	3974	3979	3980	3982	3983	3989	3990	3991
3998	3999	4023	4024	4025	4026	4029	4030	4031	4056	4057	4060	4061
4063	4064	4067	4068	4070	4071	4074	4075	4077	4078	4081	4082	4084
4085	4088	4089	4091	4092	4095	4096	4098	4099	4102	4103	4110	4111
4114	4115	4133	4134	4135	4136	4139	4140	4141	4159	4160	4163	4164
4168	4169	4172	4173	4175	4176	4179	4180	4182	4183	4186	4187	4189
4190	4193	4194	4196	4197	4200	4201	4203	4204	4207	4208	4279	4280
4285	4286	4287	4292	4293	4295	4296	4302	4303	4304	4311	4312	4336
4337	4338	4339	4342	4343	4344	4366	4367	4370	4371	4373	4374	4377
4378	4380	4381	4384	4385	4387	4388	4391	4392	4394	4395	4398	4399
4401	4402	4405	4406	4408	4409	4412	4413	4420	4421	4424	4425	4443
4444	4445	4446	4449	4450	4451	4466	4467	4470	4471	4475	4476	4479
4480	4482	4483	4486	4487	4489	4490	4493	4494	4496	4497	4500	4501
4503	4504	4507	4508	4510	4511	4514	4515	4590	4591	4594	4595	4596
4599	4600	4615	4616	4622	4623	4624	4630	4631	4656	4657	4663	4664
4666	4667	4673	4674	4676	4677	4678	4679	4682	4683	4701	4702	4705
4706	4708	4709	4712	4713	4715	4716	4719	4720	4722	4723	4726	4727
4729	4730	4733	4734	4736	4737	4740	4741	4743	4744	4747	4748	4755
4756	4759	4760	4795	4796	4799	4800	4802	4803	4806	4807	4809	4810
4813	4814	4816	4817	4820	4821	4823	4824	4827	4828	4830	4831	4834
4835	4837	4838	4841	4842	4934	4935	4940	4941	4942	4947	4948	4963
4964	4967	4968	4975	4976	4979	4980	4999	5000	5005	5006	5012	5013
5014	5021	5022	5023	5024	5048	5049	5052	5053	5055	5056	5059	5060
5062	5063	5066	5067	5069	5070	5073	5074	5076	5077	5080	5081	5083
5084	5087	5088	5090	5091	5094	5095	5102	5103	5106	5107	5124	5125
5129	5130	5143	5144	5147	5148	5150	5151	5154	5155	5157	5158	5161
5162	5164	5165	5168	5169	5171	5172	5175	5176	5178	5179	5182	5183
5185	5186	5189	5190	5192	5193	5196	5197	5239	5240	5242	5243	5246
5247	5248	5249	5281	5282	5336	5337	5338	5339	5340	5341	5342	5345
5346	5347	5357	5358	5359	5360	5361	5362	5363	5387	5388	5389	5390
5391	5393	5394	5397	5398	5399	5409	5410	5413	5414	5416	5417	5420
5421	5423	5424	5427	5428	5430	5431	5434	5435	5437	5438	5441	5442
5444	5445	5448	5449	5451	5452	5455	5456	5474	5475	5476	5477	5478
5479	5481	5482	5485	5486	5489	5490	5491	5498	5499	5501	5502	5505
5506	5510	5511	5512	5513	5514	5516	5517	5524	5525	5527	5528	5531
5532	5534	5535	5536	5537	5538	5539	5540	5541	5542	5571	5572	5575
5576	5577	5580	5581	5626	5627	5631	5632	5679	5680	5682	5683	5686
5687	5688	5689	5749	5750	5756	5757	5760	5761	5762	5763	5765	5766
5768	5769	5781	5782	5783	5785	5786	5793	5794	5796	5797	5808	5809

5811	5812	5829	5830	5833	5834	5836	5837	5842	5843	5846	5847	5848
5849	5861	5862	5863	5864	5865	5871	5872	5876	5877	5878	5880	5881
5885	5886	5887	5893	5894	5895	5896	5897	5898	5899	5900	5915	5916
5921	5922	5923	5928	5929	5960	5961	5964	5965	5967	5968	5971	5972
5974	5975	5978	5979	5981	5982	5985	5986	5988	5989	5992	5993	5995
5996	5999	6000	6002	6003	6006	6007	6020	6021	6022	6023	6024	6025
6027	6028	6031	6032	6033	6034	6035	6042	6043	6046	6047	6068	6069
6072	6073	6075	6076	6079	6080	6082	6083	6086	6087	6089	6090	6093
6094	6096	6097	6100	6101	6103	6104	6107	6108	6110	6111	6114	6115
6119	6120	6123	6124	6125	6126	6128	6129	6132	6133	6135	6136	6138
6139	6142	6143	6146	6147	6149	6150	6153	6154	6156	6157	6158	6159
6160	6281	6282	6286	6287	6290	6291	6293	6294	6296	6297	6300	6301
6302	6307	6308	6309	6310	6312	6313	6315	6316	6319	6320	6321	6326
6327	6328	6329	6330	6331	6333	6334	6335	6336	6341	6342	6343	6344
6349	6350	6351	6354	6355	6357	6358	6359	6360	6368	6369	6375	6376
6377	6378	6381	6382	6383	6384	6385	6387	6388	6394	6395	6396	6397
6404	6405	6407	6408	6409	6410	6417	6418	6420	6421	6422	6423	6424
6426	6427	6430	6431	6432	6435	6436	6441	6442	6447	6448	6451	6452
6454	6455	6456	6457	6458	6460	6461	6465	6466	6468	6469	6474	6475
6476	6479	6480	6481	6482	6483	6484	6499	6500	6505	6506	6507	6512
6513	6544	6545	6548	6549	6551	6552	6555	6556	6558	6559	6562	6563
6565	6566	6569	6570	6572	6573	6576	6577	6579	6580	6583	6584	6586
6587	6590	6591	6604	6605	6606	6607	6608	6609	6611	6612	6615	6616
6617	6618	6619	6626	6627	6630	6631	6652	6653	6656	6657	6659	6660
6663	6664	6666	6667	6670	6671	6673	6674	6677	6678	6680	6681	6684
6685	6687	6688	6691	6692	6694	6695	6698	6699	6703	6704	6707	6708
6711	6712	6714	6715	6718	6719	6721	6722	6723	6724	6725	6831	6832
6837	6838	6839	6844	6845	6867	6868	6875	6876	6877	6886	6887	6894
6895	6896	6897	6898	6899	6921	6922	6923	6943	6944	6949	6950	6951
6952	6953	6961	6962	6965	6966	6968	6969	6970	6971	6972	6975	6976
6977	6978	6979	6980	6981	6982	7014	7015	7018	7019	7021	7022	7025
7026	7028	7029	7032	7033	7035	7036	7039	7040	7042	7043	7046	7047
7049	7050	7053	7054	7056	7057	7060	7061	7069	7070	7073	7074	7094
7095	7098	7099	7101	7102	7105	7106	7108	7109	7112	7113	7115	7116
7119	7120	7122	7123	7126	7127	7129	7130	7133	7134	7136	7137	7140
7141	7143	7144	7147	7148	7150	7151	7154	7155	7256	7257	7262	7263
7264	7269	7270	7292	7293	7300	7301	7302	7309	7310	7316	7317	7318
7319	7325	7326	7328	7329	7330	7331	7332	7339	7340	7345	7346	7347
7348	7350	7351	7352	7353	7354	7356	7357	7360	7361	7362	7367	7368
7373	7374	7380	7381	7385	7386	7391	7392	7396	7397	7399	7400	7404
7405	7407	7408	7409	7410	7411	7413	7414	7419	7420	7421	7422	7423
7424	7427	7428	7429	7430	7431	7432	7433	7436	7437	7438	7443	7444
7448	7449	7450	7451	7452	7455	7456	7457	7458	7490	7491	7494	7495
7497	7498	7501	7502	7504	7505	7508	7509	7511	7512	7515	7516	7518
7519	7522	7523	7525	7526	7529	7530	7532	7533	7536	7537	7545	7546
7549	7550	7570	7571	7574	7575	7577	7578	7579	7580	7581	7582	7583
7586	7587	7589	7590	7591	7592	7593	7594	7595	7598	7599	7601	7602
7605	7606	7608	7609	7612	7613	8007	8008	8013	8014	8015	8020	8021
8064	8065	8066	8067	8068	8071	8072	8073	8075	8076	8079	8080	8081
8084	8085	8086	8087	8096	8097	8100	8101	8102	8105	8106	8151	8152
8153	8154	8166	8167	8180	8181	8182	8183	8184	8190	8191	8192	8193
8196	8197	8198	8206	8207	8208	8209	8212	8213	8222	8223	8224	8225
8226	8229	8230	8238	8239	8242	8243	8279	8280	8281	8282	8285	8286
8287	8296	8297	8300	8301	8308	8309	8315	8316	8320	8321	8325	8326
8331	8332	8335	8336	8337	8340	8341	8342	8343	8344	8345	8349	8350
8352	8353	8359	8360	8363	8364	8368	8369	8371	8372	8373	8374	8377

8378	8380	8381	8383	8384	8385	8386	8389	8390	8398	8399	8402	8403
8406	8407	8408	8409	8413	8414	8416	8417	8422	8423	8426	8427	8430
8431	8432	8433	8437	8438	8443	8444	8447	8448	8449	8452	8453	8454
8455	8459	8460	8467	8468	8471	8472	8473	8476	8477	8478	8479	8509
8510	8513	8514	8516	8517	8520	8521	8523	8524	8527	8528	8530	8531
8534	8535	8537	8538	8541	8542	8544	8545	8548	8549	8551	8552	8555
8556	8565	8566	8567	8568	8569	8570	8572	8573	8576	8577	8578	8579
8580	8593	8594	8597	8598	8618	8619	8622	8623	8625	8626	8627	8628
8629	8630	8631	8634	8635	8637	8638	8639	8640	8641	8642	8643	8646
8647	8649	8650	8653	8654	8656	8657	8660	8661	8663	8664	8667	8668
8670	8671	8674	8675	8831	8832	8833	8834	8835	8836	8837	8840	8841
8842	8856	8857	8859	8860	8864	8865	8868	8869	8870	8873	8874	8875
8876	8877	8878	8883	8884	8886	8887	8892	8893	8894	8895	8900	8901
8903	8904	8910	8911	8914	8915	8918	8919	8920	8921	8926	8927	8929
8930	8934	8935	8938	8939	8942	8943	8944	8945	8950	8951	8953	8954
8957	8958	8959	8962	8963	8964	8965	8970	8971	8973	8974	8977	8978
8979	8982	8983	8984	8985	9017	9018	9021	9022	9023	9025	9026	9027
9028	9029	9032	9033	9034	9037	9038	9039	9040	9042	9043	9046	9047
9055	9056	9057	9058	9060	9061	9062	9063	9066	9067	9068	9069	9077
9078	9079	9080	9081	9084	9085	9093	9094	9097	9098	9107	9108	9111
9112	9113	9116	9117	9143	9144	9153	9154	9163	9164	9165	9166	9178
9179	9190	9191	9192	9193	9194	9238	9239	9242	9243	9245	9246	9249
9250	9252	9253	9256	9257	9259	9260	9263	9264	9266	9267	9270	9271
9273	9274	9277	9278	9280	9281	9284	9285	9294	9295	9296	9297	9298
9299	9301	9302	9305	9306	9307	9308	9309	9316	9317	9320	9321	9342
9343	9346	9347	9349	9350	9351	9352	9353	9354	9355	9358	9359	9361
9362	9363	9364	9365	9366	9367	9370	9371	9373	9374	9377	9378	9380
9381	9384	9385	9387	9388	9391	9392	9533	9553	9554	9557	9558	9563
9564	9565	9566	9568	9569	9573	9574	9575	9603	9604	9605	9606	9614
9615	9621	9622	9625	9626	9627	9628	9636	9637	9638	9639	9648	9649
9650	9652	9653	9654	9655	9662	9663	9671	9672	9673	9674	9683	9684
9685	9687	9688	9689	9690	9697	9698	9706	9707	9708	9709	9718	9719
9720	9722	9723	9724	9725	9732	9733	9741	9742	9743	9744	9753	9754
9755	9757	9758	9759	9760	9767	9768	9776	9777	9778	9779	9788	9789
9790	9792	9793	9794	9795	9802	9803	9811	9812	9813	9814	9821	9822
9823	9824	9825	9826	9827	9832	9833	9834	9842	9843	9844	9845	9850
9851	9852	9860	9861	9862	9863	9868	9869	9870	9878	9879	9880	9881
9886	9887	9888	9900	9901	9902	9903	9906	9907	9908	9918	9919	9920
9935	9956	9957	9958	9959	9968	9969	9970	9978	9979	9980	9981	9990
9991	9992	10000	10001	10002	10003	10012	10013	10014	10022	10023	10024	10025
10034	10035	10036	10044	10045	10046	10047	10056	10057	10058	10066	10067	10068
10069	10076	10077	10078	10086	10087	10088	10089	10096	10097	10098	10106	10107
10108	10109	10116	10117	10118	10126	10127	10128	10129	10136	10137	10138	10153
10154	10160	10161	10164	10165	10166	10167	10170	10171	10172	10211	10212	10214
10215	10218	10219	10220	10221	10253	10254	10255	10256	10257	10258	10259	10262
10263	10264	10297	10298	10313	10314	10315	10318	10319	10326	10327	10342	10343
10344	10347	10348	10355	10356	10373	10374	10375	10378	10379	10419	10420	10423
10424	10443	10444	10445	10447	10448	10449	10450	10451	10452	10453	10455	10456
10460	10461	10464	10465	10466	10469	10470	10474	10475	10476	10483	10484	10487
10488	10492	10493	10494	10501	10502	10505	10506	10511	10512	10513	10521	10522
10525	10526	10527	10530	10531	10533	10534	10535	10545	10546	10549	10550	10551
10554	10555	10556	10557	10558	10566	10567	10570	10571	10590	10591	10592	10594
10595	10596	10597	10598	10599	10600	10602	10603	10607	10608	10611	10612	10613
10616	10617	10621	10622	10623	10630	10631	10634	10635	10639	10640	10641	10648
10649	10652	10653	10657	10658	10659	10667	10668	10671	10672	10673	10676	10677
10679	10680	10681	10691	10692	10695	10696	10697	10700	10701	10702	10703	10704

10712	10713	10716	10717	10736	10737	10738	10740	10741	10742	10743	10744	10745
10746	10748	10749	10753	10754	10756	10757	10760	10761	10762	10765	10766	10769
10770	10772	10773	10774	10779	10780	10786	10787	10790	10791	10794	10795	10797
10798	10799	10804	10805	10811	10812	10815	10816	10819	10820	10822	10823	10824
10832	10833	10836	10837	10838	10841	10842	10844	10845	10846	10856	10857	10860
10861	10862	10865	10866	10867	10868	10869	10911	10912	10915	10916	10918	10919
10922	10923	10925	10926	10929	10930	10932	10933	10936	10937	10939	10940	10943
10944	10946	10947	10950	10951	10953	10954	10957	10958	10970	10971	10972	10973
10974	10975	10977	10978	10981	10982	10983	10984	10985	11007	11008	11011	11012
11087	11088	11091	11092	11099	11100	11101	11102	11103	11104	11105	11108	11109
11111	11112	11113	11114	11115	11116	11117	11120	11121	11133	11134	11135	11136
11137	11138	11140	11141	11144	11145	11146	11147	11148	11174	11175	11176	11177
11178	11179	11180	11183	11184	11185	11222	11223	11238	11239	11240	11243	11244
11251	11252	11267	11268	11269	11272	11273	11280	11281	11298	11299	11300	11303
11304	11344	11345	11348	11349	11368	11369	11370	11372	11373	11374	11375	11376
11377	11378	11380	11381	11385	11386	11389	11390	11391	11394	11395	11399	11400
11401	11408	11409	11412	11413	11417	11418	11419	11426	11427	11430	11431	11436
11437	11438	11446	11447	11450	11451	11452	11455	11456	11458	11459	11460	11470
11471	11474	11475	11476	11479	11480	11481	11482	11483	11491	11492	11495	11496
11517	11518	11519	11521	11522	11523	11524	11525	11526	11527	11529	11530	11534
11535	11538	11539	11540	11543	11544	11548	11549	11550	11557	11558	11561	11562
11566	11567	11568	11575	11576	11579	11580	11584	11585	11586	11594	11595	11598
11599	11600	11603	11604	11606	11607	11608	11618	11619	11622	11623	11624	11627
11628	11629	11630	11631	11639	11640	11643	11644	11663	11664	11665	11667	11668
11669	11670	11671	11672	11673	11675	11676	11680	11681	11683	11684	11687	11688
11689	11692	11693	11696	11697	11699	11700	11701	11706	11707	11713	11714	11717
11718	11721	11722	11724	11725	11726	11731	11732	11738	11739	11742	11743	11746
11747	11749	11750	11751	11759	11760	11763	11764	11765	11768	11769	11771	11772
11773	11783	11784	11787	11788	11789	11792	11793	11794	11795	11796	11838	11839
11842	11843	11845	11846	11849	11850	11852	11853	11856	11857	11859	11860	11863
11864	11866	11867	11870	11871	11873	11874	11877	11878	11880	11881	11884	11885
11898	11899	11900	11901	11902	11903	11905	11906	11909	11910	11911	11912	11913
11935	11936	11939	11940	12015	12016	12019	12020	12027	12028	12029	12030	12031
12032	12033	12036	12037	12039	12040	12041	12042	12043	12044	12045	12048	12049
12061	12062	12063	12064	12065	12066	12068	12069	12072	12073	12074	12075	12076
12135	12136	12137	12138	12139	12140	12141	12144	12145	12146	12148	12149	12152
12153	12162	12163	12164	12165	12166	12167	12168	12171	12172	12173	12183	12184
12185	12186	12189	12190	12191	12193	12194	12195	12196	12199	12200	12201	12203
12204	12205	12206	12209	12210	12211	12212	12213	12214	12216	12217	12218	12219
12221	12222	12224	12225	12228	12229	12230	12231	12232	12236	12237	12238	12239
12242	12243	12244	12246	12247	12248	12249	12252	12253	12254	12255	12257	12258
12270	12271	12272	12273	12274	12275	12278	12279	12280	12282	12283	12285	12286
12287	12288	12291	12292	12293	12295	12296	12297	12298	12301	12302	12303	12304
12305	12306	12307	12310	12311	12312	12313	12314	12315	12316	12317	12333	12334
12337	12338	12349	12350	12355	12356	12359	12360	12361	12364	12365	12370	12371
12373	12374	12376	12377	12382	12383	12386	12387	12388	12391	12392	12397	12398
12399	12400	12402	12403	12405	12406	12410	12411	12414	12415	12416	12419	12420
12425	12426	12427	12428	12430	12431	12436	12437	12440	12441	12442	12445	12446
12447	12448	12450	12451	12455	12456	12459	12460	12461	12464	12465	12471	12472
12479	12480	12483	12484	12495	12496	12501	12502	12505	12506	12507	12510	12511
12516	12517	12519	12520	12525	12526	12529	12530	12531	12534	12535	12540	12541
12543	12544	12548	12549	12552	12553	12554	12557	12558	12563	12564	12566	12567
12572	12573	12576	12577	12578	12581	12582	12583	12584	12586	12587	12591	12592
12595	12596	12597	12600	12601	12607	12608	12640	12641	12644	12645	12647	12648
12651	12652	12654	12655	12658	12659	12661	12662	12665	12666	12668	12669	12672
12673	12675	12676	12679	12680	12682	12683	12686	12687	12692	12693	12696	12697

12702	12703	12706	12707	12715	12716	12719	12720	12750	12751	12754	12755	12757
12758	12759	12760	12761	12762	12763	12766	12767	12769	12770	12771	12772	12773
12774	12775	12778	12779	12847	12848	12849	12850	12855	12856	12857	12868	12869
12872	12873	12880	12881	12882	12883	12886	12887	12892	12893	12897	12898	12899
12900	12901	12902	12903	12906	12907	12908	12909	12910	12912	12913	12914	12915
12916	12919	12920	12921	12926	12927	12939	12940	12942	12943	12946	12947	12948
12951	12952	12953	12954	12956	12957	12960	12961	12962	12963	12964	12965	12966
12967	12971	12972	12973	12974	12977	12978	12983	12984	12985	12987	12988	12989
12990	12993	12994	12995	12996	13001	13002	13007	13008	13011	13012	13014	13015
13021	13022	13023	13024	13025	13026	13027	13030	13031	13032	13033	13034	13035
13040	13041	13042	13043	13044	13045	13046	13049	13050	13051	13054	13055	13056
13057	13058	13059	13064	13065	13066	13067	13068	13069	13070	13071	13072	13074
13075	13078	13079	13110	13111	13114	13115	13119	13120	13128	13129	13132	13133
13141	13142	13145	13146	13150	13151	13156	13157	13160	13161	13162	13165	13166
13167	13168	13172	13173	13178	13179	13180	13181	13182	13183	13184	13187	13188
13189	13192	13193	13195	13196	13197	13198	13199	13200	13203	13204	13205	13206
13207	13208	13210	13211	13215	13216	13222	13223	13225	13226	13232	13233	13234
13235	13237	13238	13240	13241	13243	13244	13245	13246	13249	13250	13255	13256
13259	13260	13261	13264	13265	13266	13267	13269	13270	13275	13276	13277	13278
13279	13280	13281	13284	13285	13286	13289	13290	13292	13293	13294	13295	13296
13297	13300	13301	13308	13309	13310	13311	13313	13314	13316	13317	13319	13320
13323	13324	13325	13328	13329	13330	13331	13333	13334	13336	13337	13338	13339
13340	13341	13342	13345	13346	13347	13350	13351	13353	13354	13355	13356	13357
13358	13361	13362	13369	13370	13371	13372	13374	13375	13383	13384	13387	13388
13389	13392	13393	13401	13402	13403	13404	13405	13408	13409	13410	13413	13414
13416	13417	13418	13419	13420	13421	13424	13425	13427	13428	13431	13432	13433
13434	13436	13437	13452	13453	13456	13457	13458	13461	13462	13464	13465	13466
13467	13468	13471	13472	13473	13476	13477	13479	13480	13481	13482	13483	13484
13487	13488	13490	13491	13494	13495	13496	13497	13529	13530	13533	13534	13536
13537	13538	13539	13540	13541	13542	13545	13546	13551	13552	13555	13556	13558
13559	13562	13563	13573	13574	13575	13576	13577	13578	13580	13581	13584	13585
13588	13589	13590	13592	13593	13596	13597	13605	13606	13607	13608	13609	13610
13611	13614	13615	13616	13621	13622	13627	13628	13631	13632	13635	13636	13683
13684	13687	13688	13690	13691	13692	13693	13694	13695	13696	13699	13700	13702
13703	13704	13705	13706	13707	13708	13711	13712	13717	13718	13721	13722	13724
13725	13728	13729	13732	13733	13734	13735	13736	13737	13739	13740	13745	13746
13747	13748	13749	13801	13802	13804	13805	13808	13809	13810	13811	13843	13844
13845	13846	13847	13848	13849	13852	13853	13854	13960	13961	13964	13965	13967
13968	13971	13972	13974	13975	13978	13979	13981	13982	13985	13986	13988	13989
13992	13993	13995	13996	13999	14000	14002	14003	14006	14007	14019	14020	14021
14022	14023	14024	14026	14027	14030	14031	14032	14033	14034	14041	14042	14045
14046	14128	14129	14132	14133	14140	14141	14142	14143	14144	14145	14146	14149
14150	14152	14153	14154	14155	14156	14157	14158	14161	14162	14183	14184	14185
14186	14187	14188	14190	14191	14194	14195	14196	14197	14198	14224	14225	14226
14227	14228	14229	14230	14233	14234	14235	14346	14347	14350	14351	14353	14354
14357	14358	14360	14361	14364	14365	14367	14368	14371	14372	14374	14375	14378
14379	14381	14382	14385	14386	14388	14389	14392	14393	14406	14407	14408	14409
14410	14411	14413	14414	14417	14418	14419	14420	14421	14428	14429	14432	14433
14516	14517	14520	14521	14528	14529	14530	14531	14532	14533	14534	14537	14538
14540	14541	14542	14543	14544	14545	14546	14549	14550	14571	14572	14573	14574
14575	14576	14578	14579	14582	14583	14584	14585	14586	14646	14647	14648	14649
14650	14651	14652	14655	14656	14657	14668	14669	14670	14671	14672	14673	14674
14677	14678	14679	14689	14690	14691	14692	14695	14696	14697	14699	14700	14701
14702	14705	14706	14707	14709	14710	14711	14712	14715	14716	14717	14718	14719
14720	14722	14723	14724	14725	14727	14728	14730	14731	14734	14735	14736	14737
14738	14742	14743	14744	14745	14748	14749	14750	14752	14753	14754	14755	14758

14759	14760	14761	14763	14764	14776	14777	14778	14779	14780	14781	14784	14785
14786	14788	14789	14791	14792	14793	14794	14797	14798	14799	14801	14802	14803
14804	14807	14808	14809	14810	14811	14812	14813	14816	14817	14818	14819	14820
14821	14822	14823	14843	14844	14846	14847	14850	14851	14852	14855	14856	14857
14858	14859	14861	14862	14867	14868	14869	14870	14871	14872	14883	14884	14885
14891	14892	14895	14896	14897	14900	14901	14904	14905	14915	14916	14918	14919
14922	14923	14924	14927	14928	14929	14930	14931	14933	14934	14939	14940	14941
14942	14943	14944	14955	14956	14957	14963	14964	14967	14968	14969	14972	14973
14976	14977	15009	15010	15013	15014	15016	15017	15018	15019	15020	15021	15022
15025	15026	15028	15029	15030	15031	15032	15033	15034	15037	15038	15045	15046
15049	15050	15081	15082	15085	15086	15088	15089	15090	15091	15092	15093	15094
15097	15098	15100	15101	15102	15103	15104	15105	15106	15109	15110	15179	15180
15182	15183	15186	15187	15188	15191	15192	15193	15194	15195	15197	15198	15203
15204	15205	15206	15207	15208	15213	15214	15215	15217	15218	15221	15222	15223
15226	15227	15228	15229	15240	15241	15244	15245	15252	15253	15254	15255	15258
15259	15270	15271	15272	15273	15274	15277	15278	15279	15282	15283	15288	15289
15290	15291	15292	15295	15296	15298	15299	15300	15301	15302	15303	15321	15322
15324	15325	15328	15329	15330	15343	15344	15345	15346	15348	15349	15362	15363
15364	15365	15366	15367	15368	15369	15371	15372	15376	15377	15378	15379	15382
15383	15386	15387	15388	15391	15392	15397	15398	15399	15401	15402	15403	15404
15407	15408	15409	15410	15412	15413	15414	15415	15416	15417	15423	15424	15429
15430	15433	15434	15436	15437	15442	15443	15444	15464	15465	15466	15467	15468
15469	15471	15472	15477	15478	15479	15482	15483	15490	15491	15492	15493	15494
15495	15496	15497	15498	15528	15529	15532	15533	15535	15536	15537	15538	15539
15542	15543	15545	15546	15547	15548	15549	15552	15553	15558	15559	15562	15563
15565	15566	15569	15570	15581	15582	15583	15584	15585	15586	15588	15589	15592
15593	15596	15597	15598	15600	15601	15604	15605	15610	15611	15616	15617	15620
15621	15624	15625	15656	15657	15660	15661	15663	15664	15665	15666	15667	15668
15669	15672	15673	15675	15676	15677	15678	15679	15680	15681	15684	15685	15690
15691	15694	15695	15697	15698	15701	15702	15705	15706	15707	15708	15709	15710
15712	15713	15718	15719	15720	15721	15722	15723	15724	15726	15727	15780	15781
15782	15783	15816	15817	15818	15819	15820	15821	15822	15825	15826	15827	15933
15934	15937	15938	15940	15941	15944	15945	15947	15948	15951	15952	15954	15955
15958	15959	15961	15962	15965	15966	15968	15969	15972	15973	15975	15976	15979
15980	15993	15994	15995	15996	15997	15998	16000	16001	16004	16005	16006	16007
16008	16015	16016	16019	16020	16100	16101	16104	16105	16112	16113	16114	16115
16116	16117	16118	16121	16122	16124	16125	16126	16127	16128	16129	16130	16133
16134	16155	16156	16157	16158	16159	16160	16162	16163	16166	16167	16168	16169
16170	16206	16207	16209	16210	16213	16214	16215	16216	16249	16250	16251	16252
16253	16254	16255	16258	16259	16260	16371	16372	16375	16376	16378	16379	16382
16383	16385	16386	16389	16390	16392	16393	16396	16397	16399	16400	16403	16404
16406	16407	16410	16411	16413	16414	16417	16418	16431	16432	16433	16434	16435
16436	16438	16439	16442	16443	16444	16445	16446	16453	16454	16457	16458	16542
16543	16546	16547	16554	16555	16556	16557	16558	16559	16560	16563	16564	16566
16567	16568	16569	16570	16571	16572	16575	16576	16597	16598	16599	16600	16601
16602	16604	16605	16608	16609	16610	16611	16612	16860	16861	16862	16863	16870
16871	16977	16991	16992	16993	17006	17016	17017	17023	17024	17029	17030	17031
17043	17053	17054	17060	17061	17066	17067	17068	17080	17090	17091	17097	17098
17103	17104	17105	17115	17125	17126	17132	17133	17138	17139	17140	17153	17163
17164	17170	17171	17176	17177	17178	17189	17203	17204	17205	17216	17230	17231
17232	17245	17263	17264	17267	17268	17269	17271	17272	17275	17276	17277	17278
17285	17286	17320	17321	17328	17329	17335	17336	17337	17340	17341	17342	17343
17344	17388	17403	17404	17407	17408	17409	17412	17413	17423	17424	17427	17428
17429	17431	17432	17437	17438	17439	17440	17441	17442	17453	17454	17455	17464
17465	17466	17467	17468									
1042*	1192*	17498*	17501									

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 441
CROSS REFERENCE TABLE -- USER SYMBOLS

\$LPERR 000412
\$LSKO = 000000

\$LS*IN= 000001

1043#													
10454#	10455	10475	10493	10512	10534	10558	10601#	10602	10622	10640	10658	10680	
10704	10747#	10748	10773	10798	10823	10845	10869	11379#	11380	11400	11418	11437	
11459	11483	11528#	11529	11549	11567	11585	11607	11631	11674#	11675	11700	11725	
11750	11772	11796											
720#	1234	1261	1264	1265	1267	1268	1270	1271	1273	1280	1282	1284	
1294	1296	1298	1300	1310	1312	1313	1315	1317	1319	1321	1324	1326	
1328	1332	1334	1336	1343	1344	1346	1347	1349	1353	1356	1357	1359	
1365	1366	1368	1372	1373	1375	1379	1380	1382	1386	1387	1389	1393	
1394	1396	1400	1401	1403	1404	1406	1410	1413	1414	1416	1422	1424	
1425	1432	1434	1436	1438	1440	1442	1444	1446	1448	1449	1451	1455	
1456	1458	1462	1463	1465	1469	1470	1472	1476	1477	1479	1483	1484	
1486	1490	1514	1516	1518	1520	1530	1532	1533	1535	1537	1539	1541	
1544	1546	1548	1552	1554	1556	1563	1564	1566	1567	1569	1573	1576	
1577	1579	1585	1586	1588	1592	1593	1595	1599	1600	1602	1606	1607	
1609	1613	1614	1616	1620	1621	1623	1627	1629	1630	1637	1639	1641	
1643	1645	1647	1649	1651	1652	1654	1658	1659	1661	1665	1666	1668	
1672	1673	1675	1679	1680	1682	1686	1687	1689	1693	1694	1696	1700	
1709	1711	1713	1715	1717	1720	1727	1729	1731	1733	1735	1737	1739	
1741	1750	1752	1754	1756	1758	1760	1761	1763	1765	1768	1772	1774	
1776	1778	1779	1780	1781	1784	1786	1789	1790	1794	1796	1798	1800	
1802	1804	1806	1807	1809	1811	1814	1815	1817	1819	1822	1828	1830	
1831	1833	1835	1838	1839	1841	1843	1846	1852	1853	1855	1857	1860	
1869	1876	1878	1880	1881	1883	1887	1888	1890	1894	1895	1897	1901	
1902	1904	1908	1909	1911	1915	1916	1918	1922	1923	1925	1934	1936	
1938	1940	1941	1943	1946	1947	1948	1949	1951	1952	1954	1958	1960	
1963	1966	1969	1970	1971	1972	1974	1975	1977	1981	1983	1986	1987	
1989	1990	1994	1995	1997	2001	2003	2004	2005	2006	2011	2013	2014	
2016	2017	2019	2021	2024	2028	2029	2031	2032	2036	2037	2039	2043	
2044	2045	2046	2051	2053	2056	2057	2059	2069	2071	2073	2075	2077	
2079	2081	2083	2085	2087	2089	2091	2093	2095	2097	2099	2101	2103	
2109	2110	2112	2116	2117	2119	2123	2124	2126	2130	2131	2133	2137	
2138	2140	2144	2145	2147	2151	2152	2154	2158	2159	2161	2165	2167	
2168	2172	2173	2175	2179	2180	2181	2182	2185	2326	2328	2330	2332	
2334	2337	2344	2346	2348	2350	2352	2354	2356	2358	2366	2368	2370	
2372	2374	2376	2377	2379	2381	2384	2388	2390	2392	2394	2395	2396	
2397	2400	2402	2405	2406	2410	2412	2414	2416	2418	2420	2422	2423	
2425	2427	2430	2431	2433	2435	2438	2444	2446	2447	2449	2451	2454	
2455	2457	2459	2462	2468	2475	2482	2484	2486	2487	2489	2493	2494	
2496	2500	2501	2503	2507	2508	2510	2514	2515	2517	2521	2522	2524	
2528	2529	2531	2540	2542	2544	2545	2547	2549	2550	2551	2554	2555	
2556	2557	2559	2560	2562	2566	2568	2571	2576	2577	2579	2583	2584	
2585	2587	2588	2589	2590	2593	2595	2596	2597	2599	2600	2603	2604	
2605	2606	2608	2609	2611	2615	2620	2621	2623	2624	2628	2629	2631	
2635	2636	2638	2639	2640	2641	2646	2648	2651	2652	2654	2664	2666	
2668	2670	2672	2674	2676	2678	2680	2682	2684	2686	2688	2690	2692	
2694	2696	2698	2704	2705	2707	2711	2712	2714	2718	2719	2721	2725	
2726	2728	2732	2733	2735	2739	2740	2742	2746	2747	2749	2753	2754	
2756	2760	2762	2766	2767	2769	2773	2774	2775	2776	2779	2830	2832	
2834	2836	2838	2841	2848	2850	2852	2854	2856	2858	2860	2862	2871	
2873	2875	2877	2879	2881	2883	2885	2887	2889	2890	2891	2892	2895	
2897	2900	2901	2905	2907	2909	2911	2913	2915	2917	2918	2920	2922	
2925	2926	2928	2930	2933	2939	2941	2942	2944	2946	2949	2950	2952	
2954	2957	2963	2970	2971	2973	2975	2976	2978	2980	2983	2985	2987	
2989	2996	3003	3005	3007	3008	3010	3014	3015	3017	3021	3022	3024	
3028	3029	3031	3035	3036	3038	3042	3043	3045	3049	3050	3052	3061	

3063	3065	3067	3069	3071	3072	3074	3077	3078	3079	3080	3082	3083
3085	3089	3091	3094	3097	3100	3101	3102	3103	3105	3106	3108	3113
3115	31 3	3119	3121	3122	3126	3127	3129	3133	3135	3136	3137	3138
3143	3144	3146	3147	3148	3150	3151	3155	3156	3158	3162	3163	3164
3165	3170	3172	3175	3176	3178	3188	3190	3192	3194	3196	3198	3200
3202	3204	3206	3208	3210	3212	3214	3216	3218	3220	3222	3224	3230
327	3233	3237	3238	3240	3244	3245	3247	3251	3252	3254	3258	3259
3261	3265	3266	3268	3272	3273	3275	3279	3280	3282	3286	3287	3289
3293	3294	3296	3300	3301	3302	3303	3306	3357	3359	3361	3363	3365
3368	3374	3376	3378	3380	3388	3390	3392	3394	3396	3398	3400	3402
3404	3407	3409	3411	3415	3416	3418	3420	3421	3423	3426	3427	3429
3430	3434	3436	3438	3440	3442	3444	3446	3448	3450	3452	3453	3455
3462	3469	3471	3473	3474	3476	3480	3481	3483	3487	3488	3490	3494
3495	3497	3501	3502	3504	3508	3509	3511	3515	3516	3518	3522	3524
3527	3528	3530	3540	3542	3544	3546	3547	3549	3551	3553	3555	3557
3559	3561	3563	3567	3569	3570	3572	3576	3577	3579	3583	3584	3586
3590	3591	3593	3597	3598	3600	3604	3605	3607	3611	3641	3643	3645
3647	3649	3652	3659	3661	3663	3665	3674	3676	3678	3680	3682	3684
3686	3688	3690	3693	3695	3697	3701	3702	3704	3706	3707	3709	3712
3713	3715	3716	3720	3722	3724	3726	3728	3730	3732	3734	3736	3738
3739	3741	3748	3755	3757	3759	3760	3762	3766	3767	3769	3773	3774
3776	3780	3781	3783	3787	3788	3790	3794	3795	3797	3801	3802	3804
3808	3810	3813	3814	3816	3826	3828	3830	3832	3833	3835	3837	3839
3841	3843	3845	3847	3849	3853	3854	3856	3857	3859	3863	3864	3866
3870	3871	3873	3877	3878	3880	3884	3885	3887	3891	3892	3894	3898
3925	3927	3929	3931	3933	3936	3943	3945	3947	3956	3958	3960	3962
3964	3966	3968	3970	3972	3975	3977	3981	3982	3984	3986	3987	3989
3992	3993	3995	3996	4000	4002	4004	4006	4008	4010	4012	4014	4016
4018	4019	4021	4024	4025	4027	4029	4032	4034	4035	4037	4044	4051
4053	4055	4056	4058	4062	4063	4065	4069	4070	4072	4076	4077	4079
4083	4084	4086	4090	4091	4093	4097	4098	4100	4104	4106	4109	4110
4112	4122	4124	4126	4127	4129	4131	4134	4135	4137	4139	4142	4143
4145	4147	4149	4151	4153	4155	4157	4159	4161	4165	4167	4168	4170
4174	4175	4177	4181	4182	4184	4188	4189	4191	4195	4196	4198	4202
4203	4205	4209	4238	4240	4242	4244	4246	4249	4256	4258	4260	4269
4271	4273	4275	4277	4279	4281	4283	4285	4288	4290	4294	4295	4297
4299	4300	4302	4305	4306	4308	4309	4313	4315	4317	4319	4321	4323
4325	4327	4329	4331	4332	4334	4337	4338	4340	4342	4345	4347	4354
4361	4363	4365	4366	4368	4372	4373	4375	4379	4380	4382	4386	4387
4389	4393	4394	4396	4400	4401	4403	4407	4408	4410	4414	4416	4419
4420	4422	4432	4434	4436	4437	4439	4441	4444	4445	4447	4449	4452
4454	4456	4458	4460	4462	4464	4466	4468	4472	4474	4475	4477	4481
4482	4484	4488	4489	4491	4495	4496	4498	4502	4503	4505	4509	4510
4512	4516	4545	4547	4549	4551	4553	4556	4563	4565	4567	4569	4576
4578	4580	4582	4584	4586	4588	4590	4592	4594	4597	4606	4608	4610
4612	4614	4615	4617	4618	4620	4622	4625	4626	4628	4632	4639	4640
4642	4643	4645	4646	4648	4649	4651	4653	4655	4656	4658	4660	4661
4665	4666	4668	4670	4671	4675	4676	4677	4678	4680	4689	4696	4698
4700	4701	4703	4707	4708	4710	4714	4715	4717	4721	4722	4724	4728
4729	4731	4735	4736	4738	4742	4743	4745	4749	4751	4754	4755	4757
4767	4769	4771	4773	4775	4777	4778	4780	4782	4784	4786	4788	4790
4792	4794	4795	4797	4801	4802	4804	4808	4809	4811	4815	4816	4818
4822	4823	4825	4829	4830	4832	4836	4837	4839	4843	4895	4897	4899
4901	4903	4906	4913	4915	4917	4924	4926	4928	4930	4932	4934	4936
4938	4940	4943	4945	4954	4956	4958	4960	4962	4963	4965	4969	4971
4972	4974	4975	4977	4986	4988	4990	4992	4994	4996	4998	4999	5001

5002	5004	5005	5007	5009	5010	5012	5015	5016	5018	5019	5025	5027
5029	5036	5043	5045	5047	5048	5050	5054	5055	5057	5061	5062	5064
5068	5069	5071	5075	5076	5078	5082	5083	5085	5089	5090	5092	5096
5098	5101	5102	5104	5114	5116	5118	5120	5122	5126	5128	5129	5131
5133	5135	5137	5139	5141	5143	5145	5149	5150	5152	5156	5157	5159
5163	5164	5166	5170	5171	5173	5177	5178	5180	5184	5185	5187	5191
5192	5194	5198	5234	5236	5238	5239	5241	5242	5244	5251	5253	5255
5257	5259	5261	5268	5270	5272	5279	5283	5285	5287	5289	5291	5293
5296	5298	5300	5302	5304	5306	5309	5311	5313	5315	5317	5319	5321
5324	5326	5328	5335	5336	5338	5340	5341	5343	5345	5348	5350	5352
5354	5358	5359	5361	5364	5366	5368	5370	5372	5374	5376	5378	5380
5382	5384	5386	5387	5388	5389	5392	5393	5395	5397	5404	5406	5408
5409	5411	5415	5416	5418	5422	5423	5425	5429	5430	5432	5436	5437
5439	5443	5444	5446	5450	5451	5453	5462	5464	5466	5468	5470	5472
5475	5476	5477	5478	5480	5481	5483	5487	5489	5492	5493	5495	5496
5500	5501	5503	5507	5509	5510	5511	5512	5515	5516	5518	5519	5521
5522	5526	5527	5529	5533	5534	5535	5536	5543	5545	5547	5549	5551
5553	5555	5557	5564	5566	5568	5570	5571	5573	5575	5578	5582	5594
5596	5598	5600	5602	5604	5606	5608	5610	5612	5614	5616	5623	5625
5626	5628	5629	5633	5635	5637	5642	5672	5675	5676	5678	5679	5681
5682	5684	5690	5692	5694	5696	5704	5706	5708	5710	5712	5715	5722
5724	5726	5728	5735	5737	5739	5741	5743	5745	5746	5748	5749	5751
5753	5755	5756	5758	5764	5765	5767	5768	5770	5772	5774	5775	5777
5779	5781	5784	5785	5787	5788	5789	5791	5795	5796	5798	5799	5800
5802	5804	5806	5810	5811	5813	5815	5817	5819	5821	5822	5823	5825
5826	5827	5832	5833	5835	5836	5838	5840	5844	5847	5848	5850	5852
5854	5855	5857	5859	5861	5866	5868	5870	5871	5873	5874	5876	5879
5880	5882	5883	5885	5888	5889	5891	5906	5908	5910	5912	5914	5915
5917	5919	5921	5924	5926	5930	5932	5934	5936	5938	5940	5947	5955
5957	5959	5960	5962	5966	5967	5969	5973	5974	5976	5980	5981	5983
5987	5988	5990	5994	5995	5997	6001	6002	6004	6014	6016	6018	6021
6022	6023	6024	6026	6027	6029	6033	6036	6038	6041	6042	6044	6053
6055	6057	6059	6061	6063	6065	6067	6068	6070	6074	6075	6077	6081
6082	6084	6088	6089	6091	6095	6096	6098	6102	6103	6105	6109	6110
612	6116	6118	6119	6121	6127	6128	6130	6134	6135	6137	6138	6140
6144	6148	6149	6151	6155	6156	6157	6158	6161	6233	6235	6237	6239
6241	6244	6251	6253	6255	6257	6264	6266	6268	6270	6272	6274	6275
6277	6278	6280	6281	6283	6285	6286	6288	6292	6293	6295	6296	6298
6300	6303	6305	6311	6312	6314	6315	6317	6319	6322	6324	6332	6333
6334	6335	6337	6339	6342	6343	6345	6347	6349	6352	6356	6357	6358
6359	6361	6362	6363	6365	6367	6368	6370	6371	6373	6376	6377	6379
6381	6386	6387	6389	6390	6392	6398	6399	6401	6403	6404	6406	6407
6408	6409	6411	6413	6415	6419	6420	6421	6422	6425	6426	6428	6430
6433	6437	6438	6440	6441	6443	6445	6449	6453	6454	6455	6456	6459
6460	6462	6463	6467	6468	6470	6472	6474	6477	6490	6492	6494	6496
6498	6499	6501	6503	6505	6508	6510	6514	6516	6518	6520	6522	6524
6531	6539	6541	6543	6544	6546	6550	6551	6553	6557	6558	6560	6564
6565	6567	6571	6572	6574	6578	6579	6581	6585	6586	6588	6598	6600
6602	6605	6606	6607	6608	6610	6611	6613	6617	6620	6622	6625	6626
6628	6637	6639	6641	6643	6645	6647	6649	6651	6652	6654	6658	6659
6661	6665	6666	6668	6672	6673	6675	6679	6680	6682	6686	6687	6689
6693	6694	6696	6700	6702	6703	6705	6709	6713	6714	6716	6720	6721
6722	6723	6726	6797	6799	6801	6803	6805	6808	6815	6817	6819	6826
6828	6830	6831	6833	6835	6837	6840	6842	6846	6848	6855	6857	6859
6861	6863	6864	6866	6867	6869	6871	6873	6875	6878	6879	6880	6883
6885	6886	6888	6890	6892	6897	6898	6900	6902	6904	6906	6908	6910

6912	6914	6915	6917	6919	6921	6924	6926	6928	6930	6932	6934	6937
6939	6940	6942	6943	6945	6946	6948	6949	6950	6951	6954	6956	6958
6960	6961	6963	6967	6968	6969	6970	6973	6988	6990	6992	6994	7001
7009	7011	7013	7014	7016	7020	7021	7023	7027	7028	7030	7034	7035
7037	7041	7042	7044	7048	7049	7051	7055	7056	7058	7063	7065	7068
7069	7071	7079	7081	7083	7085	7087	7089	7091	7093	7094	7096	7100
7101	7103	7107	7108	7110	7114	7115	7117	7121	7122	7124	7128	7129
7131	7135	7136	7138	7142	7143	7145	7149	7150	7152	7156	7222	7224
7226	7228	7230	7233	7240	7242	7244	7251	7253	7255	7256	7258	7260
7262	7265	7267	7271	7273	7280	7282	7284	7286	7288	7289	7291	7292
7294	7296	7298	7300	7303	7304	7305	7307	7311	7313	7315	7316	7320
7321	7324	7325	7327	7328	7329	7330	7333	7335	7336	7338	7339	7341
7343	7349	7350	7351	7352	7355	7356	7358	7360	7363	7364	7365	7369
7371	7375	7377	7379	7380	7382	7384	7385	7387	7389	7393	7394	7398
7399	7401	7402	7406	7407	7408	7409	7412	7413	7415	7417	7426	7427
7429	7431	7432	7434	7436	7440	7442	7443	7445	7447	7448	7449	7450
7453	7464	7466	7468	7470	7477	7485	7487	7489	7490	7492	7496	7497
7499	7503	7504	7506	7510	7511	7513	7517	7518	7520	7524	7525	7527
7531	7532	7534	7539	7541	7544	7545	7547	7555	7557	7559	7561	7563
7565	7567	7569	7570	7572	7576	7577	7578	7579	7580	7581	7584	7588
7589	7590	7591	7592	7593	7596	7600	7601	7603	7607	7608	7610	7614
7969	7971	7973	7975	7977	7980	7987	7989	7991	7993	8000	8002	8004
8006	8007	8009	8011	8013	8016	8018	8022	8024	8032	8034	8043	8044
8046	8054	8056	8058	8060	8061	8063	8064	8065	8066	8069	8071	8074
8075	8077	8079	8082	8095	8096	8098	8100	8103	8115	8116	8117	8119
8121	8123	8132	8134	8136	8137	8138	8140	8152	8153	8155	8157	8159
8160	8161	8163	8165	8166	8168	8170	8172	8174	8175	8176	8178	8182
8191	8192	8194	8196	8205	8206	8207	8208	8210	8221	8222	8223	8224
8227	8237	8238	8240	8251	8253	8254	8255	8257	8268	8270	8272	8277
8280	8281	8283	8285	8291	8293	8295	8296	8298	8305	8307	8308	8310
8311	8313	8319	8320	8322	8324	8325	8327	8328	8330	8331	8333	8335
8338	8348	8349	8351	8352	8354	8355	8357	8361	8367	8368	8370	8371
8372	8373	8375	8379	8380	8382	8383	8384	8385	8387	8391	8392	8394
8395	8397	8398	8400	8404	8412	8413	8415	8416	8418	8419	8421	8422
8424	8428	8436	8437	8439	8440	8442	8443	8445	8447	8450	8458	8459
8461	8462	8464	8466	8467	8469	8471	8474	8485	8487	8489	8490	8492
8494	8496	8504	8506	8508	8509	8511	8515	8516	8518	8522	8523	8525
8529	8530	8532	8536	8537	8539	8543	8544	8546	8550	8551	8553	8559
8561	8563	8566	8567	8568	8569	8571	8572	8574	8578	8582	8584	8587
8589	8592	8593	8595	8603	8605	8607	8609	8611	8613	8615	8617	8618
8620	8624	8625	8626	8627	8628	8629	8632	8636	8637	8638	8639	8640
8641	8644	8648	8649	8651	8655	8656	8658	8662	8663	8665	8669	8670
8672	8676	8768	8770	8772	8774	8776	8779	8786	8788	8790	8792	8800
8802	8810	8819	8821	8823	8825	8827	8828	8830	8831	8833	8835	8836
8838	8840	8849	8850	8855	8856	8858	8859	8861	8863	8864	8866	8868
8871	8882	8883	8885	8886	8888	8890	8899	8900	8902	8903	8905	8907
8909	8910	8912	8916	8925	8926	8928	8929	8931	8933	8934	8936	8940
8949	8950	8952	8953	8955	8957	8960	8969	8970	8972	8973	8975	8977
8980	8993	8994	8996	8998	8999	9001	9002	9010	9011	9013	9014	9016
9017	9019	9021	9024	9025	9026	9027	9030	9032	9035	9041	9042	9044
9054	9055	9056	9057	9059	9060	9061	9062	9064	9076	9077	9078	9079
9082	9092	9093	9095	9106	9107	9109	9111	9114	9125	9127	9129	9131
9132	9133	9142	9143	9145	9147	9149	9151	9164	9165	9168	9170	9172
9173	9175	9177	9178	9180	9182	9184	9186	9188	9192	9202	9204	9206
9208	9210	9212	9214	9216	9218	9220	9222	9224	9226	9233	9235	9237
9238	9240	9244	9245	9247	9251	9252	9254	9258	9259	9261	9265	9266

9268	9272	9273	9275	9279	9280	9282	9288	9290	9292	9295	9296	9297
9298	9300	9301	9303	9307	9310	9312	9315	9316	9318	9327	9329	9331
9333	9335	9337	9339	9341	9342	9344	9348	9349	9350	9351	9352	9353
9356	9360	9361	9362	9363	9364	9365	9368	9372	9373	9375	9379	9380
9382	9386	9387	9389	9393	9540	9542	9544	9551	9553	9555	9559	9560
9562	9563	9564	9565	9567	9568	9571	9573	9576	9578	9580	9582	9584
9586	9588	9590	9592	9594	9596	9598	9601	9613	9614	9616	9618	9620
9621	9623	9629	9637	9638	9640	9642	9644	9646	9648	9651	9652	9653
9654	9656	9658	9660	9669	9672	9673	9675	9677	9679	9681	9683	9686
9687	9688	9689	9691	9693	9695	9704	9707	9708	9710	9712	9714	9716
9718	9721	9722	9723	9724	9726	9728	9730	9739	9742	9743	9745	9747
9749	9751	9753	9756	9757	9758	9759	9761	9763	9765	9774	9777	9778
9780	9782	9784	9786	9788	9791	9792	9793	9794	9796	9798	9800	9809
9812	9813	9815	9817	9819	9821	9825	9826	9828	9830	9832	9840	9843
9844	9846	9848	9850	9858	9861	9862	9864	9866	9868	9876	9879	9880
9882	9884	9886	9889	9891	9898	9901	9902	9904	9906	9909	9911	9912
9914	9916	9920	9942	9944	9946	9948	9957	9958	9960	9962	9964	9966
9968	9976	9979	9980	9982	9984	9986	9988	9990	9998	10001	10002	10004
10006	10008	10010	10012	10020	10023	10024	10026	10028	10030	10032	10034	10042
10045	10046	10048	10050	10052	10054	10056	10064	10067	10068	10070	10072	10074
10076	10084	10087	10088	10090	10092	10094	10096	10104	10107	10108	10110	10112
10114	10116	10124	10127	10128	10130	10132	10134	10136	10144	10146	10148	10150
10152	10153	10155	10157	10159	10160	10162	10168	10172	10175	10204	10207	10208
10210	10211	10213	10214	10216	10222	10224	10226	10228	10237	10239	10241	10243
10245	10252	10253	10255	10257	10258	10260	10262	10266	10268	10275	10277	10279
10281	10283	10285	10287	10289	10296	10297	10299	10301	10303	10305	10307	10309
10311	10313	10316	10325	10326	10328	10330	10332	10334	10336	10338	10340	10342
10345	10354	10355	10357	10359	10361	10363	10365	10367	10369	10371	10373	10376
10385	10387	10388	10390	10392	10394	10395	10397	10399	10401	10402	10404	10406
10408	10410	10414	10416	10418	10419	10421	10430	10432	10433	10435	10442	10443
10444	10445	10446	10448	10449	10450	10451	10452	10453	10457	10459	10460	10462
10464	10467	10471	10474	10477	10478	10480	10482	10483	10485	10489	10492	10495
10496	10498	10500	10501	10503	10507	10511	10514	10515	10517	10518	10520	10521
10523	10525	10528	10533	10536	10537	10539	10540	10542	10544	10545	10547	10549
10552	10561	10563	10565	10566	10568	10577	10579	10580	10582	10589	10590	10591
10592	10593	10595	10596	10597	10598	10599	10600	10604	10606	10607	10609	10611
10614	10618	10621	10624	10625	10627	10629	10630	10632	10636	10639	10642	10643
10645	10647	10648	10650	10654	10657	10660	10661	10663	10664	10666	10667	10669
10671	10674	10679	10682	10683	10685	10686	10688	10690	10691	10693	10695	10698
10707	10709	10711	10712	10714	10723	10725	10726	10728	10735	10736	10737	10738
10739	10741	10742	10743	10744	10745	10746	10750	10752	10753	10755	10756	10758
10760	10763	10767	10772	10775	10776	10778	10779	10781	10783	10785	10786	10788
10792	10797	10800	10801	10803	10804	10806	10808	10810	10811	10813	10817	10822
10825	10826	10828	10829	10831	10832	10834	10836	10839	10844	10847	10848	10850
10851	10853	10855	10856	10858	10860	10863	10875	10877	10879	10881	10883	10885
10887	10889	10891	10893	10895	10897	10899	10906	10908	10910	10911	10913	10917
10918	10920	10924	10925	10927	10931	10932	10934	10938	10939	10941	10945	10946
10948	10952	10953	10955	10964	10966	10968	10971	10972	10973	10974	10976	10977
10979	10983	10987	10989	10992	10994	10997	10999	11001	11003	11006	11007	11009
11018	11025	11027	11028	11030	11032	11034	11041	11043	11044	11046	11048	11050
11057	11059	11060	11062	11064	11066	11068	11071	11074	11076	11078	11080	11082
11084	11086	11087	11089	11098	11099	11100	11101	11102	11103	11106	11110	11111
11112	11113	11114	11115	11118	11127	11129	11131	11134	11135	11136	11137	11139
11140	11142	11146	11158	11160	11162	11164	11166	11173	11174	11176	11178	11179
11181	11183	11187	11189	11196	11198	11200	11202	11204	11206	11208	11210	11212
11214	11221	11222	11224	11226	11228	11230	11232	11234	11236	11238	11241	11250

11251	11253	11255	11257	11259	11261	11263	11265	11267	11270	11279	11280	11282
11284	11286	11288	11290	11292	11294	11296	11298	11301	11310	11312	11313	11315
11317	11319	11320	11322	11324	11326	11327	11329	11331	11333	11335	11339	11341
11343	11344	11346	11355	11357	11358	11360	11367	11368	11369	11370	11371	11373
11374	11375	11376	11377	11378	11382	11384	11385	11387	11389	11392	11396	11399
11402	11403	11405	11407	11408	11410	11414	11417	11420	11421	11423	11425	11426
11428	11432	11436	11439	11440	11442	11443	11445	11446	11448	11450	11453	11458
11461	11462	11464	11465	11467	11469	11470	11472	11474	11477	11486	11488	11490
11491	11493	11497	11504	11506	11507	11509	11516	11517	11518	11519	11520	11522
11523	11524	11525	11526	11527	11531	11533	11534	11536	11538	11541	11545	11548
11551	11552	11554	11556	11557	11559	11563	11566	11569	11570	11572	11574	11575
11577	11581	11584	11587	11588	11590	11591	11593	11594	11596	11598	11601	11606
11609	11610	11612	11613	11615	11617	11618	11620	11622	11625	11634	11636	11638
11639	11641	11650	11652	11653	11655	11662	11663	11664	11665	11666	11668	11669
11670	11671	11672	11673	11677	11679	11680	11682	11683	11685	11687	11690	11694
11699	11702	11703	11705	11706	11708	11710	11712	11713	11715	11719	11724	11727
11728	11730	11731	11733	11735	11737	11738	11740	11744	11749	11752	11753	11755
11756	11758	11759	11761	11763	11766	11771	11774	11775	11777	11778	11780	11782
11783	11785	11787	11790	11802	11804	11806	11808	11810	11812	11814	11816	11818
11820	11822	11824	11826	11833	11835	11837	11838	11840	11844	11845	11847	11851
11852	11854	11858	11859	11861	11865	11866	11868	11872	11873	11875	11879	11880
11882	11892	11894	11896	11899	11900	11901	11902	11904	11905	11907	11911	11915
11917	11920	11922	11925	11927	11929	11931	11934	11935	11937	11946	11953	11955
11956	11958	11960	11962	11969	11971	11972	11974	11976	11978	11985	11987	11988
11990	11992	11994	11996	11999	12002	12004	12006	12008	12010	12012	12014	12015
12017	12026	12027	12028	12029	12030	12031	12034	12038	12039	12040	12041	12042
12043	12046	12055	12057	12059	12062	12063	12064	12065	12067	12068	12070	12074
12086	12088	12090	12092	12094	12097	12099	12101	12103	12105	12112	12114	12116
12118	12120	12127	12128	12130	12132	12134	12135	12137	12139	12140	12142	12144
12147	12148	12150	12154	12155	12157	12159	12161	12162	12164	12166	12167	12169
12171	12174	12182	12183	12184	12185	12187	12189	12192	12193	12194	12195	12197
12199	12202	12203	12204	12205	12207	12215	12216	12217	12218	12220	12221	12223
12224	12226	12230	12233	12235	12236	12237	12238	12240	12242	12245	12246	12247
12248	12250	12256	12257	12264	12266	12268	12271	12272	12273	12274	12276	12278
12281	12282	12284	12285	12286	12287	12289	12291	12294	12295	12296	12297	12299
12305	12308	12330	12332	12333	12335	12340	12342	12343	12345	12348	12349	12351
12352	12354	12355	12357	12359	12362	12366	12368	12372	12373	12375	12376	12378
12379	12381	12382	12384	12386	12389	12393	12395	12401	12402	12404	12405	12407
12409	12410	12412	12414	12417	12421	12423	12429	12430	12432	12433	12435	12436
12438	12440	12443	12449	12450	12452	12454	12455	12457	12459	12462	12466	12467
12469	12473	12476	12478	12479	12481	12486	12488	12489	12491	12494	12495	12497
12498	12500	12501	12503	12505	12508	12512	12514	12518	12519	12521	12522	12524
12525	12527	12529	12532	12536	12538	12542	12543	12545	12547	12548	12550	12552
12555	12559	12561	12565	12566	12568	12569	12571	12572	12574	12576	12579	12585
12586	12588	12590	12591	12593	12595	12598	12602	12603	12605	12614	12615	12617
12619	12620	12622	12624	12626	12628	12635	12637	12639	12640	12642	12646	12647
12649	12653	12654	12656	12660	12661	12663	12667	12668	12670	12674	12675	12677
12681	12682	12684	12688	12691	12692	12694	12698	12701	12702	12704	12709	12711
12714	12715	12717	12727	12729	12731	12733	12735	12737	12739	12741	12743	12745
12747	12749	12750	12752	12756	12757	12758	12759	12760	12761	12764	12768	12769
12770	12771	12772	12773	12776	12780	12804	12806	12808	12810	12812	12814	12817
12819	12821	12828	12830	12832	12834	12836	12843	12845	12848	12849	12851	12853
12855	12865	12867	12868	12870	12874	12879	12880	12881	12882	12884	12891	12892
12894	12896	12897	12899	12901	12902	12904	12906	12911	12912	12913	12914	12917
12919	12925	12926	12928	12929	12931	12932	12934	12936	12938	12939	12941	12942
12944	12946	12949	12955	12956	12958	12963	12964	12965	12966	12968	12970	12971

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 447
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0446

12972	12973	12975	12979	12981	12983	12986	12987	12988	12989	12991	13000	13001
13003	13004	13006	13007	13009	13013	13014	13016	13018	13020	13021	13023	13025
13026	13028	13030	13033	13036	13037	13039	13040	13042	13044	13045	13047	13049
13052	13055	13056	13057	13058	13060	13062	13064	13073	13074	13076	13083	13085
13086	13087	13089	13099	13101	13103	13105	13107	13109	13110	13112	13118	13119
13121	13123	13124	13126	13131	13132	13134	13136	13137	13139	13144	13145	13147
13149	13150	13152	13153	13155	13156	13158	13160	13163	13169	13171	13172	13174
13175	13177	13178	13179	13180	13181	13182	13185	13187	13190	13194	13195	13196
13197	13198	13199	13201	13209	13210	13212	13214	13215	13217	13218	13220	13224
13225	13227	13228	13230	13236	13237	13239	13240	13242	13243	13244	13245	13247
13251	13252	13254	13255	13257	13259	13262	13268	13269	13271	13272	13274	13275
13276	13277	13278	13279	13282	13284	13287	13291	13292	13293	13294	13295	13296
13298	13302	13304	13306	13312	13313	13315	13316	13318	13319	13321	13323	13326
13332	13333	13335	13336	13337	13338	13339	13340	13343	13345	13348	13352	13353
13354	13355	13356	13357	13359	13363	13365	13367	13373	13374	13379	13380	13382
13383	13385	13387	13390	13397	13398	13400	13401	13402	13403	13406	13408	13411
13415	13416	13417	13418	13419	13420	13422	13426	13427	13429	13435	13436	13438
13439	13441	13443	13444	13446	13451	13452	13454	13456	13459	13463	13464	13465
13466	13469	13471	13474	13478	13479	13480	13481	13482	13483	13485	13489	13490
13492	13503	13504	13506	13508	13509	13511	13513	13515	13517	13524	13526	13528
13529	13531	13535	13536	13537	13538	13539	13540	13543	13547	13548	13550	13551
13553	13557	13558	13560	13565	13567	13569	13571	13574	13575	13576	13577	13579
13580	13582	13586	13588	13591	13592	13594	13599	13601	13604	13605	13607	13609
13610	13612	13614	13617	13620	13621	13623	13625	13630	13631	13633	13643	13646
13648	13650	13652	13654	13656	13658	13660	13662	13664	13666	13668	13670	13672
13674	13676	13678	13680	13682	13683	13685	13689	13690	13691	13692	13693	13694
13697	13701	13702	13703	13704	13705	13706	13709	13714	13716	13717	13719	13723
13724	13726	13730	13733	13734	13735	13736	13738	13739	13741	13743	13747	13750
13765	13794	13797	13798	13800	13801	13803	13804	13806	13812	13814	13816	13818
13827	13829	13831	13833	13835	13842	13843	13845	13847	13848	13850	13852	13856
13858	13865	13867	13869	13871	13873	13875	13877	13879	13886	13888	13889	13890
13891	13892	13893	13894	13895	13896	13898	13899	13900	13901	13902	13903	13904
13905	13906	13908	13909	13910	13911	13912	13913	13914	13915	13916	13923	13924
13926	13928	13929	13931	13933	13934	13936	13944	13946	13948	13955	13957	13959
13960	13962	13966	13967	13969	13973	13974	13976	13980	13981	13983	13987	13988
13990	13994	13995	13997	14001	14002	14004	14013	14015	14017	14020	14021	14022
14023	14025	14026	14028	14032	14035	14037	14040	14041	14043	14052	14059	14061
14063	14065	14067	14069	14071	14078	14080	14081	14082	14083	14084	14085	14086
14087	14088	14090	14091	14092	14093	14094	14095	14096	14097	14098	14100	14101
14102	14103	14104	14105	14106	14107	14108	14111	14115	14117	14119	14121	14123
14125	14127	14128	14130	14139	14140	14141	14142	14143	14144	14147	14151	14152
14153	14154	14155	14156	14159	14168	14170	14172	14173	14175	14177	14179	14181
14184	14185	14186	14187	14189	14190	14192	14196	14208	14210	14212	14214	14216
14223	14224	14226	14228	14229	14231	14233	14237	14239	14246	14248	14250	14252
14254	14256	14258	14260	14262	14264	14271	14273	14274	14275	14276	14277	14278
14279	14280	14281	14283	14284	14285	14286	14287	14288	14289	14290	14291	14293
14294	14295	14296	14297	14298	14299	14300	14301	14309	14310	14312	14314	14315
14317	14319	14320	14322	14330	14332	14334	14341	14343	14345	14346	14348	14352
14353	14355	14359	14360	14362	14366	14367	14369	14373	14374	14376	14380	14381
14383	14387	14388	14390	14400	14402	14404	14407	14408	14409	14410	14412	14413
14415	14419	14422	14424	14427	14428	14430	14439	14446	14448	14450	14452	14454
14456	14458	14460	14462	14469	14471	14472	14473	14474	14475	14476	14477	14478
14479	14481	14482	14483	14484	14485	14486	14487	14488	14489	14491	14492	14493
14494	14495	14496	14497	14498	14499	14501	14503	14505	14507	14509	14511	14513
14515	14516	14518	14527	14528	14529	14530	14531	14532	14535	14539	14540	14541
14542	14543	14544	14547	14556	14558	14560	14561	14563	14565	14567	14569	14572

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 448
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0447

14573	14574	14575	14577	14578	14580	14584	14596	14598	14600	14602	14604	14607
14609	14611	14613	14615	14622	14624	14626	14628	14630	14638	14639	14641	14643
14645	14646	14648	14650	14651	14653	14655	14658	14660	14661	14663	14665	14667
14668	14670	14672	14673	14675	14677	14680	14688	14689	14690	14691	14693	14695
14698	14699	14700	14701	14703	14705	14708	14709	14710	14711	14713	14721	14722
14723	14724	14726	14727	14729	14730	14732	14736	14739	14741	14742	14743	14744
14746	14748	14751	14752	14753	14754	14756	14762	14763	14770	14772	14774	14777
14778	14779	14780	14782	14784	14787	14788	14790	14791	14792	14793	14795	14797
14800	14801	14802	14803	14805	14811	14814	14832	14835	14837	14838	14840	14842
14843	14845	14846	14848	14850	14853	14857	14860	14861	14863	14865	14870	14871
14873	14875	14877	14879	14881	14883	14886	14888	14890	14891	14893	14895	14898
14902	14907	14909	14910	14912	14914	14915	14917	14918	14920	14922	14925	14929
14932	14933	14935	14937	14942	14943	14945	14947	14949	14951	14953	14955	14958
14960	14962	14963	14965	14967	14970	14974	14983	14984	14986	14988	14989	14991
14993	14995	14997	15004	15006	15008	15009	15011	15015	15016	15017	15018	15019
15020	15023	15027	15028	15029	15030	15031	15032	15035	15039	15041	15044	15045
15047	15058	15060	15062	15064	15066	15068	15070	15072	15074	15076	15078	15080
15081	15083	15087	15088	15089	15090	15091	15092	15095	15099	15100	15101	15102
15103	15104	15107	15111	15135	15137	15139	15141	15143	15145	15148	15150	15152
15159	15161	15163	15165	15174	15176	15178	15179	15181	15182	15184	15186	15189
15193	15196	15197	15199	15201	15206	15207	15209	15211	15213	15216	15217	15219
15221	15224	15237	15239	15240	15242	15246	15251	15252	15253	15254	15256	15263
15265	15267	15269	15270	15271	15272	15275	15277	15280	15287	15288	15289	15290
15293	15297	15298	15299	15300	15301	15302	15307	15308	15310	15312	15314	15316
15318	15320	15321	15323	15324	15326	15328	15331	15333	15335	15337	15339	15341
15347	15348	15350	15352	15354	15356	15358	15360	15365	15366	15367	15368	15370
15371	15373	15375	15376	15377	15378	15380	15384	15386	15389	15393	15395	15397
15400	15401	15402	15403	15405	15411	15412	15413	15414	15415	15416	15422	15423
15425	15426	15428	15429	15431	15435	15436	15438	15440	15442	15445	15446	15448
15450	15452	15454	15455	15457	15459	15461	15462	15465	15466	15467	15468	15470
15471	15473	15475	15477	15480	15484	15486	15488	15490	15506	15507	15509	15511
15512	15514	15516	15518	15520	15523	15525	15527	15528	15530	15534	15535	15536
15537	15540	15544	15545	15546	15547	15550	15554	15555	15557	15558	15560	15564
15565	15567	15573	15575	15577	15579	15582	15583	15584	15585	15587	15588	15590
15594	15596	15599	15600	15602	15606	15609	15610	15612	15614	15619	15620	15622
15633	15635	15637	15639	15641	15643	15645	15647	15649	15651	15653	15655	15656
15658	15662	15663	15664	15665	15666	15667	15670	15674	15675	15676	15677	15678
15679	15682	15687	15689	15690	15692	15696	15697	15699	15703	15706	15707	15708
15709	15711	15712	15714	15716	15720	15723	15738	15766	15769	15770	15772	15773
15775	15776	15778	15784	15786	15788	15790	15800	15802	15804	15806	15808	15815
15816	15818	15820	15821	15823	15825	15829	15831	15838	15840	15842	15844	15846
15848	15850	15852	15859	15861	15862	15863	15864	15865	15866	15867	15868	15869
15871	15872	15873	15874	15875	15876	15877	15878	15879	15881	15882	15883	15884
15885	15886	15887	15888	15889	15896	15897	15899	15901	15902	15904	15906	15907
15909	15917	15919	15921	15928	15930	15932	15933	15935	15939	15940	15942	15946
15947	15949	15953	15954	15956	15960	15961	15963	15967	15968	15970	15974	15975
15977	15987	15989	15991	15994	15995	15996	15997	15999	16000	16002	16006	16009
16011	16014	16015	16017	16026	16033	16035	16037	16039	16041	16043	16045	16052
16054	16055	16056	16057	16058	16059	16060	16061	16062	16064	16065	16066	16067
16068	16069	16070	16071	16072	16074	16075	16076	16077	16078	16079	16080	16081
16082	16084	16087	16089	16091	16093	16095	16097	16099	16100	16102	16111	16112
16113	16114	16115	16116	16119	16123	16124	16125	16126	16127	16128	16131	16140
16142	16144	16145	16147	16149	16151	16153	16156	16157	16158	16159	16161	16162
16164	16168	16172	16199	16202	16203	16205	16206	16208	16209	16211	16217	16219
16221	16223	16233	16235	16237	16239	16241	16248	16249	16251	16253	16254	16256
16258	16262	16264	16271	16273	16275	16277	16279	16281	16283	16285	16287	16289

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 449
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0448

16296	16298	16299	16300	16301	16302	16303	16304	16305	16306	16308	16309	16310
16311	16312	16313	16314	16315	16316	16318	16319	16320	16321	16322	16323	16324
16325	16326	16334	16335	16337	16339	16340	16342	16344	16345	16347	16355	16357
16359	16366	16368	16370	16371	16373	16377	16378	16380	16384	16385	16387	16391
16392	16394	16398	16399	16401	16405	16406	16408	16412	16413	16415	16425	16427
16429	16432	16433	16434	16435	16437	16438	16440	16444	16447	16449	16452	16453
16455	16464	16471	16473	16475	16477	16479	16481	16483	16485	16487	16494	16496
16497	16498	16499	16500	16501	16502	16503	16504	16506	16507	16508	16509	16510
16511	16512	16513	16514	16516	16517	16518	16519	16520	16521	16522	16523	16524
16526	16529	16531	16533	16535	16537	16539	16541	16542	16544	16553	16554	16555
16556	16557	16558	16561	16565	16566	16567	16568	16569	16570	16573	16582	16584
16586	16587	16589	16591	16593	16595	16598	16599	16600	16601	16603	16604	16606
16610	16613	16859	16860	16861	16862	16864	16866	16868	16979	16981	16983	16985
16987	16989	16993	17008	17010	17012	17014	17018	17020	17022	17023	17025	17027
17031	17045	17047	17049	17051	17055	17057	17059	17060	17062	17064	17068	17082
17084	17086	17088	17092	17094	17096	17097	17099	17101	17105	17117	17119	17121
17123	17127	17129	17131	17132	17134	17136	17140	17155	17157	17159	17161	17165
17167	17169	17170	17172	17174	17178	17191	17193	17195	17197	17199	17201	17205
17218	17220	17222	17224	17226	17228	17232	17254	17262	17263	17265	17267	17270
17271	17273	17284	17285	17287	17289	17291	17293	17295	17297	17299	17301	17304
17306	17308	17310	17312	17314	17316	17318	17327	17328	17330	17335	17338	17344
17390	17392	17394	17395	17402	17403	17405	17407	17410	17422	17423	17425	17427
17430	17431	17433	17435	17440	17441	17443	17445	17447	17449	17451	17453	17456
17458	17460	17462	17468	18623	18625							
720#	1275	1277	1302	1322	1330	1351	1354	1361	1363	1370	1377	1384
1391	1398	1408	1411	1418	1420	1453	1460	1467	1474	1481	1488	1522
1542	1550	1571	1574	1581	1583	1590	1597	1604	1611	1618	1625	1656
1663	1670	1677	1684	1691	1698	1766	1770	1782	1787	1792	1812	1820
1824	1826	1836	1844	1848	1850	1858	1862	1885	1892	1899	1906	1913
1920	1927	1945	1956	1961	1964	1968	1979	1984	1992	1999	2007	2009
2022	2026	2034	2041	2047	2049	2061	2105	2107	2114	2121	2128	2135
2142	2149	2156	2163	2170	2177	2183	2382	2386	2398	2403	2408	2428
2436	2440	2442	2452	2460	2464	2466	2491	2498	2505	2512	2519	2526
2533	2553	2564	2569	2572	2574	2581	2591	2602	2613	2616	2618	2626
2633	2642	2644	2656	2700	2702	2709	2716	2723	2730	2737	2744	2751
2758	2764	2771	2777	2893	2898	2903	2923	2931	2935	2937	2947	2955
2959	2961	2972	2974	2981	3012	3019	3026	3033	3040	3047	3054	3076
3087	3092	3095	3099	3111	3116	3124	3131	3139	3141	3153	3160	3166
3168	3180	3226	3228	3235	3242	3249	3256	3263	3270	3277	3284	3291
3298	3304	3405	3413	3424	3432	3478	3485	3492	3499	3506	3513	3520
3532	3565	3574	3581	3588	3595	3602	3609	3691	3699	3710	3718	3764
3771	3778	3785	3792	3799	3806	3818	3851	3861	3868	3875	3882	3889
3896	3973	3979	3990	3998	4023	4030	4060	4067	4074	4081	4088	4095
4102	4114	4133	4140	4163	4172	4179	4186	4193	4200	4207	4286	4292
4303	4311	4336	4343	4370	4377	4384	4391	4398	4405	4412	4424	4443
4450	4470	4479	4486	4493	4500	4507	4514	4595	4599	4623	4630	4663
4673	4682	4705	4712	4719	4726	4733	4740	4747	4759	4799	4806	4813
4820	4827	4834	4841	4941	4947	4967	4979	5013	5021	5023	5052	5059
5066	5073	5080	5087	5094	5106	5124	5147	5154	5161	5168	5175	5182
5189	5196	5246	5248	5281	5337	5339	5346	5357	5362	5390	5398	5413
5420	5427	5434	5441	5448	5455	5474	5485	5490	5498	5505	5513	5524
5531	5537	5539	5541	5576	5580	5631	5686	5688	5760	5762	5782	5793
5808	5829	5842	5846	5862	5864	5877	5886	5893	5895	5897	5899	5922
5928	5964	5971	5978	5985	5992	5999	6006	6020	6031	6034	6046	6072
6079	6086	6093	6100	6107	6114	6123	6125	6132	6142	6146	6153	6159
6290	6301	6307	6309	6320	6326	6328	6330	6341	6350	6354	6375	6382

SLSTTA- 000001

6384	6394	6396	6417	6423	6431	6435	6447	6451	6457	6465	6475	6479
6481	6483	6506	6512	6548	6555	6562	6569	6576	6583	6590	6604	6615
6618	6630	6656	6663	6670	6677	6684	6691	6698	6707	6711	6718	6724
6838	6844	6876	6894	6896	6922	6952	6965	6971	6975	6977	6979	6981
7018	7025	7032	7039	7046	7053	7060	7073	7098	7105	7112	7119	7126
7133	7140	7147	7154	7263	7269	7301	7309	7318	7331	7345	7347	7353
7361	7367	7373	7391	7396	7404	7410	7419	7421	7423	7428	7430	7437
7451	7455	7457	7494	7501	7508	7515	7522	7529	7536	7549	7574	7582
7586	7594	7598	7605	7612	8014	8020	8067	8072	8080	8084	8086	8101
8105	8151	8180	8183	8190	8197	8212	8225	8229	8242	8279	8286	8300
8315	8336	8340	8342	8344	8359	8363	8377	8389	8402	8406	8408	8426
8430	8432	8448	8452	8454	8472	8476	8478	8513	8520	8527	8534	8541
8548	8555	8565	8576	8579	8597	8622	8630	8634	8642	8646	8653	8660
8667	8674	8832	8834	8841	8869	8873	8875	8877	8892	8894	8914	8918
8920	8938	8942	8944	8958	8962	8964	8978	8982	8984	9022	9028	9033
9037	9039	9046	9066	9068	9080	9084	9097	9112	9116	9153	9163	9190
9193	9242	9249	9256	9263	9270	9277	9284	9294	9305	9308	9320	9346
9354	9358	9366	9370	9377	9384	9391	9533	9557	9574	9603	9605	9625
9627	9636	9649	9662	9671	9684	9697	9706	9719	9732	9741	9754	9767
9776	9789	9802	9811	9822	9824	9833	9842	9851	9860	9869	9878	9887
9900	9907	9918	9919	9935	9956	9969	9978	9991	10000	10013	10022	10035
10044	10057	10066	10077	10086	10097	10106	10117	10126	10137	10164	10166	10170
10171	10218	10220	10254	10256	10263	10314	10318	10343	10347	10374	10378	10423
10447	10455	10465	10469	10475	10487	10493	10505	10512	10526	10530	10534	10550
10554	10556	10557	10570	10594	10602	10612	10616	10622	10634	10640	10652	10658
10672	10676	10680	10696	10700	10702	10703	10716	10740	10748	10761	10765	10769
10773	10790	10794	10798	10815	10819	10823	10837	10841	10845	10861	10865	10867
10868	10915	10922	10929	10936	10943	10950	10957	10970	10981	10984	11011	11091
11104	11108	11116	11120	11133	11144	11147	11175	11177	11184	11239	11243	11268
11272	11299	11303	11348	11372	11380	11390	11394	11400	11412	11418	11430	11437
11451	11455	11459	11475	11479	11481	11482	11495	11521	11529	11539	11543	11549
11561	11567	11579	11585	11599	11603	11607	11623	11627	11629	11630	11643	11667
11675	11688	11692	11696	11700	11717	11721	11725	11742	11746	11750	11764	11768
11772	11788	11792	11794	11795	11842	11849	11856	11863	11870	11877	11884	11898
11909	11912	11939	12019	12032	12036	12044	12048	12061	12072	12075	12136	12138
12145	12152	12163	12165	12172	12190	12200	12209	12211	12213	12228	12231	12243
12252	12254	12270	12279	12292	12301	12303	12306	12310	12312	12314	12316	12337
12360	12364	12370	12387	12391	12397	12399	12415	12419	12425	12427	12441	12445
12447	12460	12464	12471	12483	12506	12510	12516	12530	12534	12540	12553	12557
12563	12577	12581	12583	12596	12600	12607	12644	12651	12658	12665	12672	12679
12686	12696	12706	12719	12754	12762	12766	12774	12778	12847	12856	12872	12886
12898	12900	12907	12909	12915	12920	12947	12951	12953	12960	12962	12977	12984
12993	12995	13011	13022	13024	13031	13034	13041	13043	13050	13054	13065	13067
13069	13071	13078	13114	13128	13141	13161	13165	13167	13183	13188	13192	13203
13205	13207	13222	13232	13234	13249	13260	13264	13266	13280	13285	13289	13300
13308	13310	13324	13328	13330	13341	13346	13350	13361	13369	13371	13388	13392
13404	13409	13413	13424	13431	13433	13457	13461	13467	13472	13476	13487	13494
13496	13533	13541	13545	13555	13562	13573	13584	13589	13596	13606	13608	13615
13627	13635	13687	13695	13699	13707	13711	13721	13728	13732	13745	13748	13808
13810	13844	13846	13853	13964	13971	13978	13985	13992	13999	14006	14019	14030
14033	14045	14132	14145	14149	14157	14161	14183	14194	14197	14225	14227	14234
14350	14357	14364	14371	14378	14385	14392	14406	14417	14420	14432	14520	14533
14537	14545	14549	14571	14582	14585	14647	14649	14656	14669	14671	14678	14696
14706	14715	14717	14719	14734	14737	14749	14758	14750	14776	14785	14798	14807
14809	14812	14816	14818	14820	14822	14851	14855	14858	14867	14869	14884	14896
14900	14904	14923	14927	14930	14939	14941	14956	14968	14972	14976	15013	15021

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 451
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0450

15025	15033	15037	15049	15085	15093	15097	15105	15109	15187	15191	15194	15203
15205	15214	15222	15226	15228	15244	15258	15273	15278	15282	15291	15295	15329
15343	15345	15362	15364	15382	15387	15391	15398	15407	15409	15433	15443	15464
15478	15482	15491	15493	15495	15497	15532	15538	15542	15548	15552	15562	15569
15581	15592	15597	15604	15616	15624	15660	15668	15672	15680	15684	15694	15701
15705	15718	15721	15780	15782	15817	15819	15826	15937	15944	15951	15958	15965
15972	15979	15993	16004	16007	16019	16104	16117	16121	16129	16133	16155	16166
16169	16213	16215	16250	16252	16259	16375	16382	16389	16396	16403	16410	16417
16431	16442	16445	16457	16546	16559	16563	16571	16575	16597	16608	16611	16870
16977	16991	16992	17006	17016	17029	17030	17043	17053	17066	17067	17080	17090
17103	17104	17115	17125	17138	17139	17153	17163	17176	17177	17189	17203	17204
17216	17230	17231	17245	17268	17275	17277	17320	17336	17340	17342	17343	17388
17408	17412	17428	17437	17439	17454	17464	17466	17467				
1025	1029	1135#										
1156#												
1025#												
17945*	17951	17981*	17985#									
17818	17937#											
1142#	17961*	17964										
1143#	17966*											
1136#	17959	17967*	17974	17978*								
17814	17935#											
1157#												
17495	17503#											
720#	1267#	1270#	1275#	1277#	1302#	1312#	1321	1330#	1343#	1346#	1351#	1353
1356#	1361#	1363#	1365#	1370#	1372#	1377#	1379#	1384#	1386#	1391#	1393#	1398#
1400#	1403#	1408#	1410	1413#	1418#	1420#	1424#	1448#	1453#	1455#	1460#	1462#
1467#	1469#	1474#	1476#	1481#	1483#	1488#	1522#	1532#	1541	1550#	1563#	1566#
1571#	1573	1576#	1581#	1583#	1585#	1590#	1592#	1597#	1599#	1604#	1606#	1611#
1613#	1618#	1620#	1625#	1629#	1651#	1656#	1658#	1663#	1665#	1670#	1672#	1677#
1679#	1684#	1686#	1691#	1693#	1698#	1760#	1765	1770#	1778#	1786	1792#	1806#
1811	1814#	1819	1824#	1826#	1830#	1835	1838#	1843	1848#	1850#	1852#	1857
1862#	1880#	1885#	1887#	1892#	1894#	1899#	1901#	1906#	1908#	1913#	1915#	1920#
1922#	1927#	1940#	1945#	1951#	1956#	1960#	1963	1968#	1974#	1979#	1983#	1992#
1994#	1999#	2003#	2009#	2013#	2016#	2021	2026#	2034#	2036#	2041#	2043#	2049#
2056#	2061#	2099#	2101#	2105#	2107#	2109#	2114#	2116#	2121#	2123#	2128#	2130#
2135#	2137#	2142#	2144#	2149#	2151#	2156#	2158#	2163#	2170#	2172#	2177#	2179#
2376#	2381	2386#	2394#	2402	2408#	2422#	2427	2430#	2435	2440#	2442#	2446#
2451	2454#	2459	2464#	2466#	2486#	2491#	2493#	2498#	2500#	2505#	2507#	2512#
2514#	2519#	2521#	2526#	2528#	2533#	2544#	2553#	2559#	2564#	2568#	2571	2574#
2576#	2581#	2587#	2602#	2608#	2613#	2615#	2618#	2620#	2626#	2628#	2633#	2638#
2644#	2651#	2656#	2694#	2696#	2700#	2702#	2704#	2709#	2711#	2716#	2718#	2723#
2725#	2730#	2732#	2737#	2739#	2744#	2746#	2751#	2753#	2758#	2764#	2766#	2771#
2773#	2889#	2897	2903#	2917#	2922	2925#	2930	2935#	2937#	2941#	2946	2949#
2954	2959#	2961#	2970#	2980#	3007#	3012#	3014#	3019#	3021#	3026#	3028#	3033#
3035#	3040#	3042#	3047#	3049#	3054#	3071#	3076#	3082#	3087#	3091#	3094	3099#
3105#	3111#	3115#	3124#	3126#	3131#	3135#	3141#	3143#	3153#	3155#	3160#	3162#
3168#	3175#	3180#	3220#	3222#	3226#	3228#	3230#	3235#	3237#	3242#	3244#	3249#
3251#	3256#	3258#	3263#	3265#	3270#	3272#	3277#	3279#	3284#	3291#	3293#	3298#
3300#	3396#	3404	3413#	3415#	3423	3432#	3473#	3478#	3480#	3485#	3487#	3492#
3494#	3499#	3501#	3506#	3508#	3513#	3515#	3520#	3527#	3532#	3561#	3565#	3569#
3574#	3576#	3581#	3583#	3588#	3590#	3595#	3597#	3602#	3604#	3609#	3682#	3690
3699#	3701#	3709	3718#	3759#	3764#	3766#	3771#	3773#	3778#	3780#	3785#	3787#
3792#	3794#	3799#	3801#	3806#	3813#	3818#	3847#	3851#	3856#	3861#	3863#	3868#
3870#	3875#	3877#	3882#	3884#	3889#	3891#	3896#	3966#	3972	3979#	3981#	3989
3998#	4023#	4029#	4055#	4060#	4062#	4067#	4069#	4074#	4076#	4081#	4083#	4088#

SMAIL 000632
SMAMS1 000662
SMBADR 000302
SMFLG 062150
SMNEW 061717
SMMSGAD 000646
SMMSGLG 000650
SMMSGTY 000632
SMSWR 061706
SMTYP1 000663
SMXCNT 057660
SNESTL= 177777

4090#	4095#	4097#	4102#	4109#	4114#	4133#	4139#	4159#	4163#	4167#	4172#	4174#
4179#	4181#	4186#	4188#	4193#	4195#	4200#	4202#	4207#	4279#	4285	4292#	4294#
4302	4311#	4336#	4342#	4365#	4370#	4372#	4377#	4379#	4384#	4386#	4391#	4393#
4398#	4400#	4405#	4407#	4412#	4419#	4424#	4443#	4449#	4466#	4470#	4474#	4479#
4481#	4486#	4488#	4493#	4495#	4500#	4502#	4507#	4509#	4514#	4590#	4594	4599#
4614#	4622	4630#	4655#	4663#	4665#	4673#	4675#	4682#	4700#	4705#	4707#	4712#
4714#	4719#	4721#	4726#	4728#	4733#	4735#	4740#	4742#	4747#	4754#	4759#	4794#
4799#	4801#	4806#	4808#	4813#	4815#	4820#	4822#	4827#	4829#	4834#	4836#	4841#
4934#	4940	4947#	4962#	4967#	4974#	4979#	4998#	5004#	5012	5021#	5023#	5047#
5052#	5054#	5059#	5061#	5066#	5068#	5073#	5075#	5080#	5082#	5087#	5089#	5094#
5101#	5106#	5124#	5128#	5143#	5147#	5149#	5154#	5156#	5161#	5163#	5168#	5170#
5175#	5177#	5182#	5184#	5189#	5191#	5196#	5238#	5241#	5246#	5248#	5281#	5335#
5345#	5357#	5361#	5386#	5392#	5397	5408#	5413#	5415#	5420#	5422#	5427#	5429#
5434#	5436#	5441#	5443#	5448#	5450#	5455#	5474#	5480#	5485#	5489#	5498#	5500#
5505#	5509#	5515#	5524#	5526#	5531#	5533#	5539#	5541#	5570#	5575	5580#	5625#
5631#	5678#	5681#	5686#	5688#	5748#	5755#	5760#	5762#	5764#	5767#	5781	5784#
5793#	5795#	5808#	5810#	5829#	5832#	5835#	5842#	5846#	5861#	5864#	5870#	5876
5879#	588 ^c	5893#	5895#	5897#	5899#	5914#	5921	5928#	5959#	5964#	5966#	5971#
5973#	5978#	5980#	5985#	5987#	5992#	5994#	5999#	6001#	6006#	6020#	6026#	6031#
6033#	6041#	6046#	6067#	6072#	6074#	6079#	6081#	6086#	6088#	6093#	6095#	6100#
6102#	6107#	6109#	6114#	6118#	6123#	6125#	6127#	6132#	6134#	6137#	6142#	6146#
6148#	6153#	6155#	6280#	6285#	6290#	6292#	6295#	6300	6307#	6309#	6311#	6314#
6319	6326#	6328#	6330#	6332#	6341#	6349#	6354#	6356#	6367#	6375#	6381#	6384#
6386#	6394#	6396#	6403#	6406#	6417#	6419#	6425#	6430	6435#	6440#	6447#	6451#
6453#	6459#	6465#	6467#	6474	6479#	6481#	6483#	6498#	6505	6512#	6543#	6548#
6550#	6555#	6557#	6562#	6564#	6569#	6571#	6576#	6578#	6583#	6585#	6590#	6604#
6610#	6615#	6617#	6625#	6630#	6651#	6656#	6658#	6663#	6665#	6670#	6672#	6677#
6679#	6684#	6686#	6691#	6693#	6698#	6702#	6707#	6711#	6713#	6718#	6720#	6830#
6837	6844#	6866#	6875	6885#	6894#	6896#	6921#	6942#	6948#	6960#	6965#	6967#
6975#	6977#	6979#	6981#	7013#	7018#	7020#	7025#	7027#	7032#	7034#	7039#	7041#
7046#	7048#	7053#	7055#	7060#	7068#	7073#	7093#	7098#	7100#	7105#	7107#	7112#
7114#	7119#	7121#	7126#	7128#	7133#	7135#	7140#	7142#	7147#	7149#	7154#	7255#
7262	7269#	7291#	7300	7309#	7315#	7318#	7324#	7327#	7338#	7345#	7347#	7349#
7355#	7360	7367#	7373#	7379#	7384#	7391#	7396#	7398#	7404#	7406#	7412#	7419#
7421#	7423#	7426#	7436#	7442#	7447#	7455#	7457#	7489#	7494#	7496#	7501#	7503#
7508#	7510#	7515#	7517#	7522#	7524#	7529#	7531#	7536#	7544#	7549#	7569#	7574#
7576#	7586#	7588#	7598#	7600#	7605#	7607#	7612#	8006#	8013	8020#	8063#	8071
8074#	8079	8084#	8086#	8095#	8100	8105#	8151#	8165#	8180#	8182#	8190#	8196#
8205#	8212#	8221#	8229#	8237#	8242#	8279#	8285#	8295#	8300#	8307#	8315#	8319#
8324#	8330#	8335	8340#	8342#	8344#	8348#	8351#	8359#	8363#	8367#	8370#	8377#
8379#	8382#	8389#	8397#	8402#	8406#	8408#	8412#	8415#	8421#	8426#	8430#	8432#
8436#	8442#	8447	8452#	8454#	8458#	8466#	8471	8476#	8478#	8508#	8513#	8515#
8520#	8522#	8527#	8529#	8534#	8536#	8541#	8543#	8548#	8550#	8555#	8565#	8571#
8576#	8578#	8592#	8597#	8617#	8622#	8624#	8634#	8636#	8646#	8648#	8653#	8655#
8660#	8662#	8667#	8669#	8674#	8830#	8840#	8855#	8858#	8863#	8868	8873#	8875#
8877#	8882#	8885#	8892#	8894#	8899#	8902#	8909#	8914#	8918#	8920#	8925#	8928#
8933#	8938#	8942#	8944#	8949#	8952#	8957	8962#	8964#	8969#	8972#	8977	8982#
8984#	9016#	9021	9024#	9032	9037#	9039#	9041#	9046#	9054#	9059#	9066#	9068#
9076#	9084#	9092#	9097#	9106#	9111	9116#	9142#	9153#	9163#	9177#	9190#	9192#
9237#	9242#	9244#	9249#	9251#	9256#	9258#	9263#	9265#	9270#	9272#	9277#	9279#
9284#	9294#	9300#	9305#	9307#	9315#	9320#	9341#	9346#	9348#	9358#	9360#	9370#
9372#	9377#	9379#	9384#	9386#	9391#	9534#	9553#	9557#	9562#	9567#	9573	9603#
9605#	9613#	9620#	9625#	9627#	9636#	9648#	9651#	9662#	9671#	9683#	9686#	9697#
9706#	9718#	9721#	9732#	9741#	9753#	9756#	9767#	9776#	9788#	9791#	9802#	9811#
9821#	9824#	9832#	9842#	9850#	9860#	9868#	9878#	9886#	9900#	9906#	9918#	9936#
9956#	9968#	9978#	9990#	10000#	10012#	10022#	10034#	10044#	10056#	10066#	10076#	10086#

10096#	10106#	10116#	10126#	10136#	10152#	10159#	10164#	10166#	10170#	10210#	10213#	10218#
10220#	10252#	10262#	10296#	10313	10318#	10325#	10342	10347#	10354#	10373	10378#	10418#
10423#	10442#	10455	10459#	10464	10469#	10474	10482#	10487#	10492	10500#	10505#	10511
10520#	10525	10530#	10533	10544#	10549	10554#	10556#	10565#	10570#	10589#	10602	10606#
10611	10616#	10621	10629#	10634#	10639	10647#	10652#	10657	10666#	10671	10676#	10679
10690#	10695	10700#	10702#	10711#	10716#	10735#	10748	10752#	10755#	10760	10765#	10769#
10772	10778#	10785#	10790#	10794#	10797	10803#	10810#	10815#	10819#	10822	10831#	10836
10841#	10844	10855#	10860	10865#	10867#	10910#	10915#	10917#	10922#	10924#	10929#	10931#
10936#	10938#	10943#	10945#	10950#	10952#	10957#	10970#	10976#	10981#	10983#	11006#	11011#
11086#	11091#	11098#	11108#	11110#	11120#	11133#	11139#	11144#	11146#	11173#	11183#	11221#
11238	11243#	11250#	11267	11272#	11279#	11298	11303#	11343#	11348#	11367#	11380	11384#
11389	11394#	11399	11407#	11412#	11417	11425#	11430#	11436	11445#	11450	11455#	11458
11469#	11474	11479#	11481#	11490#	11495#	11516#	11529	11533#	11538	11543#	11548	11556#
11561#	11566	11574#	11579#	11584	11593#	11598	11603#	11606	11617#	11622	11627#	11629#
11638#	11643#	11662#	11675	11679#	11682#	11687	11692#	11696#	11699	11705#	11712#	11717#
11721#	11724	11730#	11737#	11742#	11746#	11749	11758#	11763	11768#	11771	11782#	11787
11792#	11794#	11837#	11842#	11844#	11849#	11851#	11856#	11858#	11863#	11865#	11870#	11872#
11877#	11879#	11884#	11898#	11904#	11909#	11911#	11934#	11939#	12014#	12019#	12026#	12036#
12038#	12048#	12061#	12067#	12072#	12074#	12134#	12144#	12147#	12152#	12161#	12171#	12182#
12189	12192#	12199	12202#	12209#	12211#	12213#	12215#	12220#	12223#	12228#	12230	12235#
12242	12245#	12252#	12254#	12256#	12270#	12278#	12281#	12284#	12291	12294#	12301#	12303#
12305	12310#	12312#	12314#	12316#	12332#	12337#	12348#	12354#	12359	12364#	12370#	12372#
12375#	12381#	12386	12391#	12397#	12399#	12401#	12404#	12409#	12414	12419#	12425#	12427#
12429#	12435#	12440	12445#	12447#	12449#	12454#	12459	12464#	12471#	12478#	12483#	12494#
12500#	12505	12510#	12516#	12518#	12524#	12529	12534#	12540#	12542#	12547#	12552	12557#
12563#	12565#	12571#	12576	12581#	12583#	12585#	12590#	12595	12600#	12607#	12639#	12644#
12646#	12651#	12653#	12658#	12660#	12665#	12667#	12672#	12674#	12679#	12681#	12686#	12691#
12696#	12701#	12706#	12714#	12719#	12749#	12754#	12756#	12766#	12768#	12778#	12847#	12855#
12867#	12872#	12879#	12886#	12891#	12896#	12906#	12909#	12911#	12919	12925#	12938#	12941#
12946	12951#	12953#	12955#	12960#	12962#	12970#	12977#	12983#	12986#	12993#	12995#	13000#
13006#	13011#	13013#	13020#	13030#	13033	13039#	13049#	13054#	13064#	13067#	13069#	13071#
13073#	13078#	13109#	13114#	13118#	13128#	13131#	13141#	13144#	13149#	13155#	13160	13165#
13167#	13171#	13177#	13187	13192#	13194#	13203#	13205#	13207#	13209#	13214#	13222#	13224#
13232#	13234#	13236#	13239#	13242#	13249#	13254#	13259	13264#	13266#	13268#	13274#	13284
13289#	13291#	13300#	13308#	13310#	13312#	13315#	13318#	13323	13328#	13330#	13332#	13335#
13345	13350#	13352#	13361#	13369#	13371#	13373#	13382#	13387	13392#	13400#	13408	13413#
13415#	13424#	13426#	13431#	13433#	13435#	13451#	13456	13461#	13463#	13471	13476#	13478#
13487#	13489#	13494#	13496#	13528#	13533#	13535#	13545#	13550#	13555#	13557#	13562#	13573#
13579#	13584#	13588#	13591#	13596#	13604#	13614#	13620#	13627#	13630#	13635#	13682#	13687#
13689#	13699#	13701#	13711#	13716#	13721#	13723#	13728#	13732#	13738#	13745#	13747#	13800#
13803#	13808#	13810#	13842#	13852#	13959#	13964#	13966#	13971#	13973#	13978#	13980#	13985#
13987#	13992#	13994#	13999#	14001#	14006#	14019#	14025#	14030#	14032#	14040#	14045#	14127#
14132#	14139#	14149#	14151#	14161#	14183#	14189#	14194#	14196#	14223#	14233#	14345#	14350#
14352#	14357#	14359#	14364#	14366#	14371#	14373#	14378#	14380#	14385#	14387#	14392#	14406#
14412#	14417#	14419#	14427#	14432#	14515#	14520#	14527#	14537#	14539#	14549#	14571#	14577#
14582#	14584#	14645#	14655#	14667#	14677#	14688#	14695	14698#	14705	14708#	14715#	14717#
14719#	14721#	14726#	14729#	14734#	14736	14741#	14748	14751#	14758#	14760#	14762#	14776#
14784#	14787#	14790#	14797	14800#	14807#	14809#	14811	14816#	14818#	14820#	14822#	14842#
14845#	14850	14855#	14857	14860#	14867#	14869#	14883#	14890#	14895	14900#	14904#	14914#
14917#	14922	14927#	14929	14932#	14939#	14941#	14955#	14962#	14967	14972#	14976#	15008#
15013#	15015#	15025#	15027#	15037#	15044#	15049#	15080#	15085#	15087#	15097#	15099#	15109#
15178#	15181#	15186	15191#	15193	15196#	15203#	15205#	15213#	15216#	15221	15226#	15228#
15239#	15244#	15251#	15258#	15269#	15277	15282#	15287#	15295#	15297#	15320#	15323#	15328
15343#	15345#	15347#	15362#	15364#	15370#	15375#	15382#	15386	15391#	15397#	15400#	15407#
15409#	15411#	15422#	15428#	15433#	15435#	15442	15464#	15470#	15477	15482#	15490#	15493#
15495#	15497#	15527#	15532#	15534#	15542#	15544#	15552#	15557#	15562#	15564#	15569#	15581#

15587#	15592#	15596#	15599#	15604#	15609#	15616#	15619#	15624#	15655#	15660#	15662#	15672#
15674#	15684#	15689#	15694#	15696#	15701#	15705#	15711#	15718#	15720#	15772#	15775#	15780#
15782#	15815#	15825#	15932#	15937#	15939#	15944#	15946#	15951#	15953#	15958#	15960#	15965#
15967#	15972#	15974#	15979#	15993#	15999#	16004#	16006#	16014#	16019#	16099#	16104#	16111#
16121#	16123#	16133#	16155#	16161#	16166#	16168#	16205#	16208#	16213#	16215#	16248#	16258#
16370#	16375#	16377#	16382#	16384#	16389#	16391#	16396#	16398#	16403#	16405#	16410#	16412#
16417#	16431#	16437#	16442#	16444#	16452#	16457#	16541#	16546#	16553#	16563#	16565#	16575#
16597#	16603#	16608#	16610#	16859#	16870#	16978#	16991#	17007#	17016#	17022#	17029#	17044#
17053#	17059#	17066#	17081#	17090#	17096#	17103#	17116#	17125#	17131#	17138#	17154#	17163#
17169#	17176#	17190#	17203#	17217#	17230#	17246#	17262#	17267	17270#	17275#	17277#	17284#
17320#	17327#	17335	17340#	17342#	17389#	17402#	17407	17412#	17422#	17427	17430#	17437#
17439#	17453#	17464#	17466#									
1267#	1277	1302#	1424	1448#	1453	1455#	1460	1462#	1467	1469#	1474	1476#
1481	1483#	1488	1522#	1629	1651#	1656	1658#	1663	1665#	1670	1672#	1677
1679#	1684	1686#	1691	1693#	1698	1760#	1765	1770	1778#	1786	1792	1806#
1811	1826	1830#	1835	1850	1852#	1857	1862	1880#	1885	1887#	1892	1894#
1899	1901#	1906	1908#	1913	1915#	1920	1922#	1927	1940#	1963	2009	2013#
2049	2056#	2061	2099#	2107	2109#	2114	2116#	2121	2123#	2128	2130#	2135
2137#	2142	2144#	2149	2151#	2156	2158#	2163	2170#	2179	2376#	2381	2386
2394#	2402	2408	2422#	2427	2442	2446#	2451	2466	2486#	2491	2493#	2498
2500#	2505	2507#	2512	2514#	2519	2521#	2526	2528#	2533	2544#	2571	2618
2620#	2644	2651#	2656	2694#	2702	2704#	2709	2711#	2716	2718#	2723	2725#
2730	2732#	2737	2739#	2744	2746#	2751	2753#	2758	2764#	2773	2889#	2897
2903	2917#	2922	2937	2941#	2946	2961	2970#	2980	3007#	3012	3014#	3019
3021#	3026	3028#	3033	3035#	3040	3042#	3047	3049#	3054	3071#	3094	3141
3143#	3168	3175#	3180	3220#	3228	3230#	3235	3237#	3242	3244#	3249	3251#
3256	3258#	3263	3265#	3270	3272#	3277	3279#	3284	3291#	3300	3396#	3404
3413	3415#	3423	3432	3473#	3478	3480#	3485	3487#	3492	3494#	3499	3501#
3506	3508#	3513	3515#	3520	3527#	3532	3561#	3565	3569#	3574	3576#	3581
3583#	3588	3590#	3595	3597#	3602	3604#	3600	3682#	3690	3699	3701#	3709
3718	3759#	3764	3766#	3771	3773#	3778	3780#	3785	3787#	3792	3794#	3799
3801#	3806	3813#	3818	3847#	3851	3856#	3861	3863#	3868	3870#	3875	3877#
3882	3884#	3889	3891#	3896	3966#	3972	3979	3981#	3989	3998	4023#	4029
4055#	4060	4062#	4067	4069#	4074	4076#	4081	4083#	4088	4090#	4095	4097#
4102	4109#	4114	4133#	4139	4159#	4163	4167#	4172	4174#	4179	4181#	4186
4188#	4193	4195#	4200	4202#	4207	4279#	4285	4292	4294#	4302	4311	4336#
4342	4365#	4370	4372#	4377	4379#	4384	4386#	4391	4393#	4398	4400#	4405
4407#	4412	4419#	4424	4443#	4449	4466#	4470	4474#	4479	4481#	4486	4488#
4493	4495#	4500	4502#	4507	4509#	4514	4590#	4594	4599	4614#	4622	4630
4655#	4663	4665#	4673	4675#	4682	4700#	4705	4707#	4712	4714#	4719	4721#
4726	4728#	4733	4735#	4740	4742#	4747	4754#	4759	4794#	4799	4801#	4806
4808#	4813	4815#	4820	4822#	4827	4829#	4834	4836#	4841	4934#	4940	4947
4962#	4979	4998#	5023	5047#	5052	5054#	5059	5061#	5066	5068#	5073	5075#
5080	5082#	5087	5089#	5094	5101#	5106	5124#	5128	5143#	5147	5149#	5154
5156#	5161	5163#	5168	5170#	5175	5177#	5182	5184#	5189	5191#	5196	5238#
5248	5281#	5386	5392#	5397	5541	5570#	5575	5580	5625#	5631	5678#	5688
5748#	5762	5764#	5899	5914#	5921	5928	5959#	5964	5966#	5971	5973#	5978
5980#	5985	5987#	5992	5994#	5999	6001#	6006	6020#	6033	6041#	6046	6067#
6072	6074#	6079	6081#	6086	6088#	6093	6095#	6100	6102#	6107	6109#	6114
6118#	6123	6125#	6134	6137#	6142	6146#	6155	6280#	6330	6332#	6354	6356#
6483	6498#	6505	6512	6543#	6548	6550#	6555	6557#	6562	6564#	6569	6571#
6576	6578#	6583	6585#	6590	6604#	6617	6625#	6630	6651#	6656	6658#	6663
6665#	6670	6672#	6677	6679#	6684	6686#	6691	6693#	6698	6702#	6707	6711#
6720	6830#	6837	6844	6866#	6875	6981	7013#	7018	7020#	7025	7027#	7032
7034#	7039	7041#	7046	7048#	7053	7055#	7060	7068#	7073	7093#	7098	7100#
7105	7107#	7112	7114#	7119	7121#	7126	7128#	7133	7135#	7140	7142#	7147

NSKO = 000300

7149#	7154	7255#	7262	7269	7291#	7300	7318	7324#	7423	7426#	7436	7442#
7457	7489#	7494	7496#	7501	7503#	7508	7510#	7515	7517#	7522	7524#	7529
7531#	7536	7544#	7549	7569#	7574	7576#	7586	7588#	7598	7600#	7605	7607#
7612	8006#	8013	8020	8063#	8071	8086	8095#	8100	8105	8151#	8182	8190#
8196	8205#	8212	8221#	8229	8237#	8242	8279#	8285	8295#	8300	8307#	8315
8319#	8344	8348#	8363	8367#	8408	8412#	8432	8436#	8454	8458#	8478	8508#
8513	8515#	8520	8522#	8527	8529#	8534	8536#	8541	8543#	8548	8550#	8555
8565#	8578	8592#	8597	8617#	8622	8624#	8634	8636#	8646	8648#	8653	8655#
8660	8662#	8667	8669#	8674	8830#	8840	8855#	8877	8882#	8894	8899#	8920
8925#	8944	8949#	8964	8969#	8984	9016#	9021	9039	9041#	9046	9054#	9068
9076#	9084	9092#	9097	9106#	9111	9116	9142#	9153	9163#	9192	9237#	9242
9244#	9249	9251#	9256	9258#	9263	9265#	9270	9272#	9277	9279#	9284	9294#
9307	9315#	9320	9341#	9346	9348#	9358	9360#	9370	9372#	9377	9379#	9384
9386#	9391	9534#	9918	9936#	10170	10210#	10220	10252#	10262	10296#	10313	10318
10325#	10342	10347	10354#	10373	10378	10418#	10423	10442#	10455	10474	10492	10511
10533	10556	10565#	10570	10589#	10602	10621	10639	10657	10679	10702	10711#	10716
10735#	10748	10772	10797	10822	10844	10867	10910#	10915	10917#	10922	10924#	10929
10931#	10936	10938#	10943	10945#	10950	10952#	10957	10970#	10983	11006#	11011	11086#
11091	11098#	11108	11110#	11120	11133#	11146	11173#	11183	11221#	11238	11243	11250#
11267	11272	11279#	11298	11303	11343#	11348	11367#	11380	11399	11417	11436	11458
11481	11490#	11495	11516#	11529	11548	11566	11584	11606	11629	11638#	11643	11662#
11675	11699	11724	11749	11771	11794	11837#	11842	11844#	11849	11851#	11856	11858#
11863	11865#	11870	11872#	11877	11879#	11884	11898#	11911	11934#	11939	12014#	12019
12026#	12036	12038#	12048	12061#	12074	12134#	12144	12147#	12152	12161#	12171	12182#
12189	12213	12215#	12316	12332#	12337	12348#	12370	12372#	12399	12401#	12427	12429#
12447	12449#	12471	12478#	12483	12494#	12516	12518#	12540	12542#	12563	12565#	12583
12585#	12607	12639#	12644	12646#	12651	12653#	12658	12660#	12665	12667#	12672	12674#
12679	12681#	12686	12691#	12696	12701#	12706	12714#	12719	12749#	12754	12756#	12766
12768#	12778	12847#	12855	12867#	12872	12879#	12886	12891#	12909	12911#	12919	13071
13073#	13078	13109#	13114	13118#	13128	13131#	13141	13144#	13207	13209#	13234	13236#
13310	13312#	13371	13373#	13433	13435#	13496	13528#	13533	13535#	13545	13550#	13555
13557#	13562	13573#	13588	13591#	13596	13604#	13614	13620#	13627	13630#	13635	13682#
13687	13689#	13699	13701#	13711	13716#	13721	13723#	13728	13732#	13747	13800#	13810
13842#	13852	13959#	13964	13966#	13971	13973#	13978	13980#	13985	13987#	13992	13994#
13999	14001#	14006	14019#	14032	14040#	14045	14127#	14132	14139#	14149	14151#	14161
14183#	14196	14223#	14233	14345#	14350	14352#	14357	14359#	14364	14366#	14371	14373#
14378	14380#	14385	14387#	14392	14406#	14419	14427#	14432	14515#	14520	14527#	14537
14539#	14549	14571#	14584	14645#	14655	14667#	14677	14688#	14695	14719	14721#	14822
14842#	14857	14904	14914#	14929	14976	15008#	15013	15015#	15025	15027#	15037	15044#
15049	15080#	15085	15087#	15097	15099#	15109	15178#	15193	15228	15239#	15244	15251#
15258	15269#	15277	15282	15287#	15295	15297#	15409	15411#	15497	15527#	15532	15534#
15542	15544#	15552	15557#	15562	15564#	15569	15581#	15596	15599#	15604	15609#	15616
15619#	15624	15655#	15660	15662#	15672	15674#	15684	15689#	15694	15696#	15701	15705#
15720	15772#	15782	15815#	15825	15932#	15937	15939#	15944	15946#	15951	15953#	15958
15960#	15965	15967#	15972	15974#	15979	15993#	16006	16014#	16019	16099#	16104	16111#
16121	16123#	16133	16155#	16168	16205#	16215	16248#	16258	16370#	16375	16377#	16382
16384#	16389	16391#	16396	16398#	16403	16405#	16410	16412#	16417	16431#	16444	16452#
16457	16541#	16546	16553#	16563	16565#	16575	16597#	16610	16859#	16870	16978#	16991
17007#	17029	17044#	17066	17081#	17103	17116#	17138	17154#	17176	17190#	17203	17217#
17230	17246#	17342	17389#	17466								
1270#	1275	1312#	1321	1330	1343#	1353	1363	1365#	1370	1372#	1377	1379#
1384	1386#	1391	1393#	1398	1400#	1410	1420	1532#	1541	1550	1563#	1573
1583	1585#	1590	1592#	1597	1599#	1604	1606#	1611	1613#	1618	1620#	1625
1814#	1819	1824	1838#	1843	1848	1945#	1960	1968#	1983	1992#	2003	2016#
2021	2026	2034#	2043	2101#	2105	2172#	2177	2430#	2435	2440	2454#	2459
2464	2553#	2568	2574#	2587	2602#	2615	2626#	2638	2696#	2700	2766#	2771

SNSK1 = 000110

2925#	2930	2935	2949#	2954	2959	3076#	3091	3099#	3115	3124#	3135	3153#
3162	3222#	3226	3293#	3298	4967#	4974	5004#	5012	5021	5241#	5246	5335#
5345	5357#	5361	5408#	5413	5415#	5420	5422#	5427	5429#	5434	5436#	5441
5443#	5448	5450#	5455	5474#	5489	5498#	5509	5515#	5539	5681#	5686	5755#
5760	5767#	5781	5897	6026#	6031	6127#	6132	6148#	6153	6285#	6290	6292#
6309	6311#	6328	6341#	6349	6367#	6384	6386#	6394	6396#	6403	6406#	6417
6419#	6451	6453#	6481	6610#	6615	6713#	6718	6885#	6894	6896#	6921	6942#
6979	7309#	7315	7327#	7347	7349#	7396	7398#	7404	7406#	7421	7447#	7455
8074#	8079	8084	8165#	8180	8324#	8342	8351#	8359	8370#	8377	8379#	8406
8415#	8430	8442#	8447	8452	8466#	8471	8476	8571#	8576	8858#	8875	8885#
8892	8902#	8918	8928#	8942	8952#	8957	8962	8972#	8977	8982	9024#	9032
9037	9059#	9066	9177#	9190	9300#	9305	9553#	9557	9562#	9605	9613#	9627
9636#	9648	9651#	9662	9671#	9683	9686#	9697	9706#	9718	9721#	9732	9741#
9753	9756#	9767	9776#	9788	9791#	9802	9811#	9821	9824#	9832	9842#	9850
9860#	9868	9878#	9886	9900#	9906	9956#	9968	9978#	9990	10000#	10012	10022#
10034	10044#	10056	10066#	10076	10086#	10096	10106#	10116	10126#	10136	10152#	10166
10213#	10218	10459#	10464	10469	10482#	10487	10500#	10505	10520#	10525	10530	10544#
10549	10554	10606#	10611	10616	10629#	10634	10647#	10652	10666#	10671	10676	10690#
10695	10700	10752#	10769	10778#	10794	10803#	10819	10831#	10836	10841	10855#	10860
10865	10976#	10981	11139#	11144	11384#	11389	11394	11407#	11412	11425#	11430	11445#
11450	11455	11469#	11474	11479	11533#	11538	11543	11556#	11561	11574#	11579	11593#
11598	11603	11617#	11622	11627	11679#	11696	11705#	11721	11730#	11746	11758#	11763
11768	11782#	11787	11792	11904#	11909	12067#	12072	12192#	12199	12211	12220#	12230
12314	12354#	12359	12364	12375#	12397	12404#	12425	12435#	12440	12445	12454#	12459
12464	12500#	12505	12510	12524#	12529	12534	12547#	12552	12557	12571#	12576	12581
12590#	12595	12600	12896#	12906	12925#	12995	13000#	13069	13149#	13167	13171#	13205
13214#	13222	13224#	13232	13239#	13266	13268#	13308	13315#	13330	13332#	13369	13382#
13387	13392	13400#	13408	13413	13415#	13424	13426#	13431	13451#	13456	13461	13463#
13471	13476	13478#	13487	13489#	13494	13579#	13584	13738#	13745	13803#	13808	14025#
14030	14189#	14194	14412#	14417	14577#	14582	14698#	14705	14717	14726#	14736	14820
14845#	14850	14855	14860#	14867	14869#	14883	14890#	14895	14900	14917#	14922	14927
14932#	14939	14941#	14955	14962#	14967	14972	15181#	15186	15191	15196#	15203	15205#
15213	15216#	15221	15226	15320#	15345	15347#	15362	15364#	15397	15407#	15407	15422#
15495	15587#	15592	15711#	15718	15775#	15780	15999#	16004	16161#	16161	16208#	16213
16437#	16442	16603#	16608	17016#	17022	17053#	17059	17090#	17096	17124#	17131	17163#
17169	17262#	17267	17277	17284#	17320	17327#	17335	17340	17402#	17407	17412	17422#
17427	17464											
1346#	1351	1356#	1361	1403#	1408	1413#	1418	1566#	1571	1576#	1581	1951#
1956	1974#	1979	1994#	1999	2036#	2041	2559#	2564	2576#	2581	2608#	2613
2628#	2633	3082#	3087	3105#	3111	3126#	3131	3155#	3160	5480#	5485	5500#
5505	5524#	5533	5784#	5793	5795#	5808	5810#	5829	5832#	5864	5870#	5876
5895	6295#	6300	6307	6314#	6319	6326	6375#	6381	6425#	6430	6435	6440#
6447	6459#	6465	6467#	6474	6479	6948#	6977	7338#	7345	7355#	7360	7367
7373#	7379	7384#	7391	7412#	7419	8330#	8335	8340	8382#	8389	8397#	8402
8421#	8426	8863#	8868	8877	8909#	8914	8933#	8938	9567#	9573	9603	9620#
9625	10159#	10164	10755#	10760	10765	10785#	10790	10810#	10815	11682#	11687	11692
11712#	11717	11737#	11742	12202#	12209	12223#	12228	12235#	12242	12254	12256#	12312
12381#	12386	12391	12409#	12414	12419	12938#	12953	12955#	12960	12962#	12983	12986#
12993	13006#	13011	13013#	13033	13067	13155#	13160	13165	13177#	13187	13192	13194#
13203	13242#	13249	13254#	13259	13264	13274#	13284	13289	13291#	13300	13318#	13323
13328	13335#	13345	13350	13352#	13361	14708#	14715	14729#	14734	14741#	14748	14760
14762#	14818	15323#	15328	15343	15370#	15386	15391	15428#	15433	15435#	15442	15493
17270#	17275	17430#	17437	17439#	17453							
5526#	5531	5835#	5842	5846#	5861	5879#	5885	5893	6960#	6965	6967#	6975
12245#	12252	12250#	12278	12281#	12305	12310	12941#	12946	12951	12970#	12977	13020#
13030	13039#	13049	13054#	13064	14751#	14758	14776#	14784	14787#	14811	14816	15375#

SNSK2 = 000120

SNSK3 = 000120

5757#	5760#	5762#	5766#	5769#	5782#	5783#	5786#	5793#	5797#	5808#	5812#	5829#
5834#	5837#	5842#	5847#	5849#	5861#	5864#	5872#	5877#	5878#	5881#	5886#	5887#
5893#	5895#	5897#	5899#	5916#	5922#	5923#	5928#	5961#	5964#	5968#	5971#	5975#
5978#	5982#	5985#	5989#	5992#	5996#	5999#	6003#	6006#	6021#	6025#	6028#	6031#
6033#	6043#	6046#	6069#	6072#	6076#	6079#	6083#	6086#	6090#	6093#	6097#	6100#
6104#	6107#	6111#	6114#	6120#	6123#	6126#	6129#	6132#	6134#	6139#	6142#	6147#
6150#	6153#	6155#	6282#	6287#	6290#	6294#	6297#	6301#	6302#	6307#	6309#	6313#
6316#	6320#	6321#	6326#	6328#	6330#	6336#	6342#	6344#	6349#	6354#	6360#	6369#
6376#	6378#	6381#	6384#	6388#	6394#	6397#	6403#	6410#	6417#	6424#	6427#	6431#
6432#	6435#	6442#	6447#	6451#	6458#	6461#	6465#	6469#	6475#	6476#	6479#	6481#
6483#	6500#	6506#	6507#	6512#	6545#	6548#	6552#	6555#	6559#	6562#	6566#	6569#
6573#	6576#	6580#	6583#	6587#	6590#	6605#	6609#	6612#	6615#	6617#	6627#	6630#
6653#	6656#	6660#	6663#	6667#	6670#	6674#	6677#	6681#	6684#	6688#	6691#	6695#
6698#	6704#	6707#	6712#	6715#	6718#	6720#	6832#	6838#	6839#	6844#	6868#	6876#
6877#	6887#	6894#	6897#	6899#	6921#	6944#	6953#	6962#	6965#	6972#	6975#	6977#
6979#	6981#	7015#	7018#	7022#	7025#	7029#	7032#	7036#	7039#	7043#	7046#	7050#
7053#	7057#	7060#	7070#	7073#	7095#	7098#	7102#	7105#	7109#	7112#	7116#	7119#
7123#	7126#	7130#	7133#	7137#	7140#	7144#	7147#	7151#	7154#	7257#	7263#	7264#
7269#	7293#	7301#	7302#	7310#	7315#	7318#	7326#	7332#	7340#	7345#	7347#	7354#
7357#	7361#	7362#	7367#	7374#	7379#	7386#	7391#	7396#	7400#	7404#	7411#	7414#
7419#	7421#	7423#	7428#	7430#	7433#	7436#	7437#	7444#	7452#	7455#	7457#	7491#
7494#	7498#	7501#	7505#	7508#	7512#	7515#	7519#	7522#	7526#	7529#	7533#	7536#
7546#	7549#	7571#	7574#	7583#	7586#	7595#	7598#	7602#	7605#	7609#	7612#	8008#
8014#	8015#	8020#	8068#	8072#	8073#	8076#	8080#	8081#	8084#	8086#	8097#	8101#
8102#	8105#	8152#	8154#	8167#	8180#	8182#	8191#	8193#	8196#	8209#	8212#	8226#
8229#	8239#	8242#	8280#	8282#	8285#	8297#	8300#	8309#	8315#	8321#	8326#	8332#
8336#	8337#	8340#	8342#	8344#	8350#	8353#	8359#	8363#	8369#	8374#	8377#	8381#
8386#	8389#	8399#	8402#	8406#	8408#	8414#	8417#	8423#	8426#	8430#	8432#	8438#
8444#	8448#	8449#	8452#	8454#	8460#	8468#	8472#	8473#	8476#	8478#	8510#	8513#
8517#	8520#	8524#	8527#	8531#	8534#	8538#	8541#	8545#	8548#	8552#	8555#	8566#
8570#	8573#	8576#	8578#	8594#	8597#	8619#	8622#	8631#	8634#	8643#	8646#	8650#
8653#	8657#	8660#	8664#	8667#	8671#	8674#	8832#	8834#	8837#	8840#	8841#	8857#
8860#	8865#	8869#	8870#	8873#	8875#	8877#	8884#	8887#	8892#	8894#	8901#	8904#
8911#	8914#	8918#	8920#	8927#	8930#	8935#	8938#	8942#	8944#	8951#	8954#	8958#
8959#	8962#	8964#	8971#	8974#	8978#	8979#	8982#	8984#	9018#	9022#	9023#	9029#
9033#	9034#	9037#	9039#	9043#	9046#	9058#	9063#	9066#	9068#	9081#	9084#	9094#
9097#	9108#	9112#	9113#	9116#	9144#	9153#	9164#	9166#	9179#	9190#	9192#	9239#
9242#	9246#	9249#	9253#	9256#	9260#	9263#	9267#	9270#	9274#	9277#	9281#	9284#
9295#	9299#	9302#	9305#	9307#	9317#	9320#	9343#	9346#	9355#	9358#	9367#	9370#
9374#	9377#	9381#	9384#	9388#	9391#	9554#	9557#	9566#	9569#	9574#	9575#	9603#
9605#	9615#	9622#	9625#	9627#	9637#	9639#	9648#	9655#	9662#	9672#	9674#	9683#
9690#	9697#	9707#	9709#	9718#	9725#	9732#	9742#	9744#	9753#	9760#	9767#	9777#
9779#	9788#	9795#	9802#	9812#	9814#	9821#	9825#	9827#	9832#	9843#	9845#	9850#
9861#	9863#	9868#	9879#	9881#	9886#	9901#	9903#	9906#	9957#	9959#	9968#	9979#
9981#	9990#	10001#	10003#	10012#	10023#	10025#	10034#	10045#	10047#	10056#	10067#	10069#
10076#	10087#	10089#	10096#	10107#	10109#	10116#	10127#	10129#	10136#	10154#	10161#	10164#
10166#	10212#	10215#	10218#	10220#	10254#	10256#	10259#	10262#	10263#	10298#	10314#	10315#
10318#	10327#	10343#	10344#	10347#	10356#	10374#	10375#	10378#	10420#	10423#	10443#	10448#
10455	10461#	10465#	10466#	10469#	10474	10475	10484#	10487#	10492	10493	10502#	10505#
10511	10512	10522#	10526#	10527#	10530#	10533	10534	10546#	10550#	10551#	10554#	10556#
10557	10558#	10567#	10570#	10590#	10595#	10602	10608#	10612#	10613#	10616#	10621	10622
10631#	10634#	10639	10640	10649#	10652#	10657	10658	10668#	10672#	10673#	10676#	10679
10680	10692#	10696#	10697#	10700#	10702#	10703	10704#	10713#	10716#	10736#	10741#	10748
10754#	10757#	10761#	10762#	10765#	10769#	10772	10773	10780#	10787#	10790#	10794#	10797
10798	10805#	10812#	10815#	10819#	10822	10823	10833#	10837#	10838#	10841#	10844	10845
10857#	10861#	10862#	10865#	10867#	10868	10869#	10912#	10915#	10919#	10922#	10926#	10929#

10933#	10936#	10940#	10943#	10947#	10950#	10954#	10957#	10971#	10975#	10978#	10981#	10983#
11008#	11011#	11088#	11091#	11105#	11108#	11117#	11120#	11134#	11138#	11141#	11144#	11146#
11175#	11177#	11180#	11183#	11184#	11223#	11239#	11240#	11243#	11252#	11268#	11269#	11272#
11281#	11299#	11300#	11303#	11345#	11348#	11368#	11373#	11380#	11386#	11390#	11391#	11394#
11399#	11400#	11409#	11412#	11417#	11418#	11427#	11430#	11436#	11437#	11447#	11451#	11452#
11455#	11458#	11459#	11471#	11475#	11476#	11479#	11481#	11482#	11483#	11492#	11495#	11517#
11522#	11529#	11535#	11539#	11540#	11543#	11548#	11549#	11558#	11561#	11566#	11567#	11576#
11579#	11584#	11585#	11595#	11599#	11600#	11603#	11606#	11607#	11619#	11623#	11624#	11627#
11629#	11630#	11631#	11640#	11643#	11663#	11668#	11675#	11681#	11684#	11688#	11689#	11692#
11696#	11699#	11700#	11707#	11714#	11717#	11721#	11724#	11725#	11732#	11739#	11742#	11746#
11749#	11750#	11760#	11764#	11765#	11768#	11771#	11772#	11784#	11788#	11789#	11792#	11794#
11795#	11796#	11839#	11842#	11846#	11849#	11853#	11856#	11860#	11863#	11867#	11870#	11874#
11877#	11881#	11884#	11899#	11903#	11906#	11909#	11911#	11936#	11939#	12016#	12019#	12033#
12036#	12045#	12048#	12062#	12066#	12069#	12072#	12074#	12136#	12138#	12141#	12144#	12145#
12149#	12152#	12163#	12165#	12168#	12171#	12172#	12186#	12190#	12191#	12196#	12200#	12201#
12206#	12209#	12211#	12213#	12219#	12222#	12225#	12228#	12231#	12232#	12239#	12243#	12244#
12249#	12252#	12254#	12258#	12271#	12275#	12278#	12283#	12288#	12292#	12293#	12298#	12301#
12303#	12306#	12307#	12310#	12312#	12314#	12316#	12334#	12337#	12350#	12356#	12360#	12361#
12364#	12370#	12374#	12377#	12383#	12387#	12388#	12391#	12397#	12399#	12403#	12406#	12411#
12415#	12416#	12419#	12425#	12427#	12431#	12437#	12441#	12442#	12445#	12447#	12451#	12456#
12460#	12461#	12464#	12471#	12480#	12483#	12496#	12502#	12506#	12507#	12510#	12516#	12520#
12526#	12530#	12531#	12534#	12540#	12544#	12549#	12553#	12554#	12557#	12563#	12567#	12573#
12577#	12578#	12581#	12583#	12587#	12592#	12596#	12597#	12600#	12607#	12641#	12644#	12648#
12651#	12655#	12658#	12662#	12665#	12669#	12672#	12676#	12679#	12683#	12686#	12693#	12696#
12703#	12706#	12716#	12719#	12751#	12754#	12763#	12766#	12775#	12778#	12848#	12850#	12855#
12869#	12872#	12883#	12886#	12893#	12898#	12900#	12903#	12906#	12907#	12909#	12916#	12920#
12921#	12927#	12940#	12943#	12947#	12948#	12951#	12953#	12957#	12960#	12963#	12967#	12974#
12977#	12983#	12990#	12993#	12995#	13002#	13008#	13011#	13015#	13022#	13024#	13027#	13030#
13031#	13034#	13035#	13041#	13043#	13046#	13049#	13050#	13055#	13059#	13064#	13067#	13069#
13071#	13075#	13078#	13111#	13114#	13120#	13128#	13133#	13141#	13146#	13151#	13157#	13161#
13162#	13165#	13167#	13173#	13184#	13188#	13189#	13192#	13200#	13203#	13205#	13207#	13211#
13216#	13222#	13226#	13232#	13234#	13238#	13241#	13246#	13249#	13256#	13260#	13261#	13264#
13266#	13270#	13281#	13285#	13286#	13289#	13297#	13300#	13308#	13310#	13314#	13317#	13320#
13324#	13325#	13328#	13330#	13334#	13342#	13346#	13347#	13350#	13358#	13361#	13369#	13371#
13375#	13384#	13388#	13389#	13392#	13405#	13409#	13410#	13413#	13421#	13424#	13428#	13431#
13433#	13437#	13453#	13457#	13458#	13461#	13468#	13472#	13473#	13476#	13484#	13487#	13491#
13494#	13496#	13530#	13533#	13542#	13545#	13552#	13555#	13559#	13562#	13574#	13578#	13581#
13584#	13588#	13593#	13596#	13606#	13608#	13611#	13614#	13615#	13622#	13627#	13632#	13635#
13684#	13687#	13696#	13699#	13708#	13711#	13718#	13721#	13725#	13728#	13733#	13737#	13740#
13745#	13747#	13802#	13805#	13808#	13810#	13844#	13846#	13849#	13852#	13853#	13961#	13964#
13968#	13971#	13975#	13978#	13982#	13985#	13989#	13992#	13996#	13999#	14003#	14006#	14020#
14024#	14027#	14030#	14032#	14042#	14045#	14129#	14132#	14146#	14149#	14158#	14161#	14184#
14188#	14191#	14194#	14196#	14225#	14227#	1430#	14233#	14234#	14347#	14350#	14354#	14357#
14361#	14364#	14368#	14371#	14375#	14378#	14382#	14385#	14389#	14392#	14407#	14411#	14414#
14417#	14419#	14429#	14432#	14517#	14520#	14534#	14537#	14546#	14549#	14572#	14576#	14579#
14582#	14584#	14647#	14649#	14652#	14655#	14656#	14669#	14671#	14674#	14677#	14678#	14692#
14696#	14697#	14702#	14706#	14707#	14712#	14715#	14717#	14719#	14725#	14728#	14731#	14734#
14737#	14738#	14745#	14749#	14750#	14755#	14758#	14760#	14764#	14777#	14781#	14784#	14789#
14794#	14798#	14799#	14804#	14807#	14809#	14812#	14813#	14816#	14818#	14820#	14822#	14844#
14847#	14851#	14852#	14855#	14858#	14859#	14862#	14867#	14870#	14872#	14883#	14892#	14896#
14897#	14900#	14904#	14916#	14919#	14923#	14924#	14927#	14930#	14931#	14934#	14939#	14942#
14944#	14955#	14964#	14968#	14969#	14972#	14976#	15010#	15013#	15022#	15025#	15034#	15037#
15046#	15049#	15082#	15085#	15094#	15097#	15106#	15109#	15180#	15183#	15187#	15188#	15191#
15194#	15195#	15198#	15203#	15206#	15208#	15213#	15218#	15222#	15223#	15226#	15228#	15241#
15244#	15255#	15258#	15274#	15278#	15279#	15282#	15292#	15295#	15303#	15322#	15325#	15329#
15330#	15343#	15345#	15349#	15362#	15365#	15369#	15372#	15379#	15382#	15387#	15388#	15391#

15397#	15404#	15407#	15409#	15417#	15424#	15430#	15433#	15437#	15443#	15444#	15465#	15469#
15472#	15478#	15479#	15482#	15490#	15493#	15495#	15497#	15529#	15532#	15539#	15542#	15549#
15552#	15559#	15562#	15566#	15569#	15582#	15586#	15589#	15592#	15596#	15601#	15604#	15611#
15616#	15621#	15624#	15657#	15660#	15669#	15672#	15681#	15684#	15691#	15694#	15698#	15701#
15706#	15710#	15713#	15718#	15720#	15774#	15777#	15780#	15782#	15817#	15819#	15822#	15825#
15826#	15934#	15937#	15941#	15944#	15948#	15951#	15955#	15958#	15962#	15965#	15969#	15972#
15976#	15979#	15994#	15998#	16001#	16004#	16006#	16016#	16019#	16101#	16104#	16118#	16121#
16130#	16133#	16156#	16160#	16163#	16166#	16168#	16207#	16210#	16213#	16215#	16250#	16252#
16255#	16258#	16259#	16372#	16375#	16379#	16382#	16386#	16389#	16393#	16396#	16400#	16403#
16407#	16410#	16414#	16417#	16432#	16436#	16439#	16442#	16444#	16454#	16457#	16543#	16546#
16560#	16563#	16572#	16575#	16598#	16602#	16605#	16608#	16610#	16863#	16870#	17017#	17022#
17054#	17059#	17091#	17096#	17126#	17131#	17164#	17169#	17264#	17268#	17269#	17272#	17275#
17277#	17286#	17320#	17329#	17336#	17337#	17340#	17404#	17408#	17409#	17412#	17424#	17428#
17429#	17432#	17437#	17440#	17442#	17453#	17464#						
720#	1268	1269#	1271	1272#	1302	1303#	1313	1314#	1321	1323#	1344	1345#
1347	1348#	1353	1355#	1357	1358#	1366	1367#	1373	1374#	1380	1381#	1387
1388#	1394	1395#	1401	1402#	1404	1405#	1410	1412#	1414	1415#	1449	1450#
1456	1457#	1463	1464#	1470	1471#	1477	1478#	1484	1485#	1522	1523#	1533
1534#	1541	1543#	1564	1565#	1567	1568#	1573	1575#	1577	1578#	1586	1587#
1593	1594#	1600	1601#	1607	1608#	1614	1615#	1621	1622#	1652	1653#	1659
1660#	1666	1667#	1673	1674#	1680	1681#	1687	1688#	1694	1695#	1761	1762#
1765	1767#	1779	1781	1782	1783#	1786	1788#	1807	1808#	1811	1813#	1815
1816#	1819	1821#	1831	1832#	1835	1837#	1839	1840#	1843	1845#	1853	1854#
1857	1859#	1881	1882#	1888	1889#	1895	1896#	1902	1903#	1909	1910#	1916
1917#	1923	1924#	1941	1942#	1945	1946#	1947	1949	1950#	1952	1953#	1963
1965#	1968	1969#	1970	1972	1973#	1975	1976#	1992	1993#	1995	1996#	2004
2007	2008#	2014	2015#	2017	2018#	2021	2023#	2034	2035#	2037	2038#	2044
2047	2048#	2057	2058#	2099	2100#	2101	2102#	2110	2111#	2117	2118#	2124
2125#	2131	2132#	2138	2139#	2145	2146#	2152	2153#	2159	2160#	2170	2171#
2173	2174#	2180	2183	2184#	2377	2378#	2381	2383#	2395	2397	2398	2399#
2402	2404#	2423	2424#	2427	2429#	2431	2432#	2435	2437#	2447	2448#	2451
2453#	2455	2456#	2459	2461#	2487	2488#	2494	2495#	2501	2502#	2508	2509#
2515	2516#	2522	2523#	2529	2530#	2545	2546#	2553	2554#	2555	2557	2558#
2560	2561#	2571	2573#	2574	2575#	2577	2578#	2588	2591	2592#	2602	2603#
2604	2606	2607#	2609	2610#	2621	2622#	2626	2627#	2629	2630#	2639	2642
2643#	2652	2653#	2694	2695#	2696	2697#	2705	2706#	2712	2713#	2719	2720#
2726	2727#	2733	2734#	2740	2741#	2747	2748#	2754	2755#	2764	2765#	2767
2768#	2774	2777	2778#	2890	2892	2893	2894#	2897	2899#	2918	2919#	2922
2924#	2926	2927#	2930	2932#	2942	2943#	2946	2948#	2950	2951#	2954	2956#
2971	2972#	2973#	2976	2977#	3008	3009#	3015	3016#	3022	3023#	3029	3030#
3036	3037#	3043	3044#	3050	3051#	3072	3073#	3076	3077#	3078	3080	3081#
3083	3084#	3094	3096#	3099	3100#	3101	3103	3104#	3106	3107#	3124	3125#
3127	3128#	3136	3139	3140#	3144	3145#	3153	3154#	3156	3157#	3163	3166
3167#	3176	3177#	3220	3221#	3222	3223#	3231	3232#	3238	3239#	3245	3246#
3252	3253#	3259	3260#	3266	3267#	3273	3274#	3280	3281#	3291	3292#	3294
3295#	3301	3304	3305#	3396	3397#	3404	3406#	3416	3417#	3423	3425#	3474
3475#	3481	3482#	3488	3489#	3495	3496#	3502	3503#	3509	3510#	3516	3517#
3528	3529#	3561	3562#	3570	3571#	3577	3578#	3584	3585#	3591	3592#	3598
3599#	3605	3606#	3682	3683#	3690	3692#	3702	3703#	3709	3711#	3760	3761#
3767	3768#	3774	3775#	3781	3782#	3788	3789#	3795	3796#	3802	3803#	3814
3815#	3847	3848#	3857	3858#	3864	3865#	3871	3872#	3878	3879#	3885	3886#
3892	3893#	3966	3967#	3972	3974#	3982	3983#	3989	3991#	4023	4024#	4025
4026#	4056	4057#	4063	4064#	4070	4071#	4077	4078#	4084	4085#	4091	4092#
4098	4099#	4110	4111#	4133	4134#	4135	4136#	4159	4160#	4168	4169#	4175
4176#	4182	4183#	4189	4190#	4196	4197#	4203	4204#	4279	4280#	4285	4287#
4295	4296#	4302	4304#	4336	4337#	4338	4339#	4366	4367#	4373	4374#	4380

STAGNU= 052420

4381#	4387	4388#	4394	4395#	4401	4402#	4408	4409#	4420	4421#	4443	4444#
4445	4446#	4466	4467#	4475	4476#	4482	4483#	4489	4490#	4496	4497#	4503
4504#	4510	4511#	4590	4591#	4594	4596#	4615	4616#	4622	4624#	4656	4657#
4666	4667#	4676	4678	4679#	4701	4702#	4708	4709#	4715	4716#	4722	4723#
4729	4730#	4736	4737#	4743	4744#	4755	4756#	4795	4796#	4802	4803#	4809
4810#	4816	4817#	4823	4824#	4830	4831#	4837	4838#	4934	4935#	4940	4942#
4963	4964#	4967	4968#	4999	5000#	5005	5006#	5012	5014#	5048	5049#	5055
5056#	5062	5063#	5069	5070#	5076	5077#	5083	5084#	5090	5091#	5102	5103#
5124	5125#	5143	5144#	5150	5151#	5157	5158#	5164	5165#	5171	5172#	5178
5179#	5185	5186#	5192	5193#	5239	5240#	5242	5243#	5281	5282#	5336	5337#
5338#	5341	5342#	5357	5358#	5359	5360#	5387	5390	5391#	5393	5394#	5397
5399#	5409	5410#	5416	5417#	5423	5424#	5430	5431#	5437	5438#	5444	5445#
5451	5452#	5474	5475#	5476	5478	5479#	5481	5482#	5498	5499#	5501	5502#
5510	5513	5514#	5516	5517#	5524	5525#	5527	5528#	5534	5537	5538#	5571
5572#	5575	5577#	5626	5627#	5679	5680#	5682	5683#	5749	5750#	5756	5757#
5765	5766#	5768	5769#	5781	5783#	5785	5786#	5796	5797#	5811	5812#	5833
5834#	5836	5837#	5846	5847#	5848	5849#	5871	5872#	5876	5878#	5880	5881#
5885	5887#	5915	5916#	5921	5923#	5960	5961#	5967	5968#	5974	5975#	5981
5982#	5988	5989#	5995	5996#	6002	6003#	6020	6021#	6022	6024	6025#	6027
6028#	6042	6043#	6068	6069#	6075	6076#	6082	6083#	6089	6090#	6096	6097#
6103	6104#	6110	6111#	6119	6120#	6125	6126#	6128	6129#	6138	6139#	6146
6147#	6149	6150#	6156	6159	6160#	6281	6282#	6286	6287#	6293	6294#	6296
6297#	6300	6302#	6312	6313#	6315	6316#	6319	6321#	6333	6335	6336#	6341
6342#	6343	6344#	6357	6359	6360#	6368	6369#	6375	6376#	6377	6378#	6387
6388#	6396	6397#	6407	6409	6410#	6420	6422	6423	6424#	6426	6427#	6430
6432#	6441	6442#	6454	6456	6457	6458#	6460	6461#	6468	6469#	6474	6476#
6499	6500#	6505	6507#	6544	6545#	6551	6552#	6558	6559#	6565	6566#	6572
6573#	6579	6580#	6586	6587#	6604	6605#	6606	6608	6609#	6611	6612#	6626
6627#	6652	6653#	6659	6660#	6666	6667#	6673	6674#	6680	6681#	6687	6688#
6694	6695#	6703	6704#	6711	6712#	6714	6715#	6721	6724	6725#	6831	6832#
6837	6839#	6867	6868#	6875	6877#	6886	6887#	6896	6897#	6898	6899#	6943
6944#	6949	6951	6952	6953#	6961	6962#	6968	6970	6971	6972#	7014	7015#
7021	7022#	7028	7029#	7035	7036#	7042	7043#	7049	7050#	7056	7057#	7069
7070#	7094	7095#	7101	7102#	7108	7109#	7115	7116#	7122	7123#	7129	7130#
7136	7137#	7143	7144#	7150	7151#	7256	7257#	7262	7264#	7292	7293#	7300
7302#	7309	7310#	7325	7326#	7328	7330	7331	7332#	7339	7340#	7350	7352
7353	7354#	7356	7357#	7360	7362#	7373	7374#	7385	7386#	7399	7400#	7407
7409	7410	7411#	7413	7414#	7427	7428#	7429#	7432	7433#	7443	7444#	7448
7450	7451	7452#	7490	7491#	7497	7498#	7504	7505#	7511	7512#	7518	7519#
7525	7526#	7532	7533#	7545	7546#	7570	7571#	7577	7579	7581	7582	7583#
7589	7591	7593	7594#	7595#	7601	7602#	7608	7609#	8007	8008#	8013	8015#
8064	8066	8067	8068#	8071	8073#	8075	8076#	8079	8081#	8096	8097#	8100
8102#	8151	8152#	8153	8154#	8166	8167#	8190	8191#	8192	8193#	8206	8208
8209#	8222	8224	8225	8226#	8238	8239#	8279	8280#	8281	8282#	8296	8297#
8308	8309#	8320	8321#	8325	8326#	8331	8332#	8335	8337#	8349	8350#	8352
8353#	8368	8369#	8371	8373	8374#	8380	8381#	8383	8385	8386#	8398	8399#
8413	8414#	8416	8417#	8422	8423#	8437	8438#	8443	8444#	8447	8449#	8459
8460#	8467	8468#	8471	8473#	8509	8510#	8516	8517#	8523	8524#	8530	8531#
8537	8538#	8544	8545#	8551	8552#	8565	8566#	8567	8569	8570#	8572	8573#
8593	8594#	8618	8619#	8625	8627	8629	8630	8631#	8637	8639	8641	8642
8643#	8649	8650#	8656	8657#	8663	8664#	8670	8671#	8831	8832#	8833#	8836
8837#	8856	8857#	8859	8860#	8864	8865#	8868	8870#	8883	8884#	8886	8887#
8900	8901#	8903	8904#	8910	8911#	8926	8927#	8929	8930#	8934	8935#	8950
8951#	8953	8954#	8957	8959#	8970	8971#	8973	8974#	8977	8979#	9017	9018#
9021	9023#	9025	9027	9028	9029#	9032	9034#	9042	9043#	9055	9057	9058#
9060	9062	9063#	9077	9079	9080	9081#	9093	9094#	9107	9108#	9111	9113#

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 463
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0462

9143	9144#	9163	9164#	9165	9166#	9178	9179#	9238	9239#	9245	9246#	9252
9253#	9259	9260#	9266	9267#	9273	9274#	9280	9281#	9294	9295#	9296	9298
9299#	9301	9302#	9316	9317#	9342	9343#	9349	9351	9353	9354	9355#	9361
9363	9365	9366#	9367#	9373	9374#	9380	9381#	9387	9388#	9534#	9553	9554#
9563	9565	9566#	9568	9569#	9573	9575#	9614	9615#	9621	9622#	9636	9637#
9638	9639#	9652	9654	9655#	9671	9672#	9673	9674#	9687	9689	9690#	9706
9707#	9708	9709#	9722	9724	9725#	9741	9742#	9743	9744#	9757	9759	9760#
9776	9777#	9778	9779#	9792	9794	9795#	9811	9812#	9813	9814#	9824	9825#
9826	9827#	9842	9843#	9844	9845#	9860	9861#	9862	9863#	9878	9879#	9880
9881#	9900	9901#	9902	9903#	9936#	9956	9957#	9958	9959#	9978	9979#	9980
9981#	10000	10001#	10002	10003#	10022	10023#	10024	10025#	10044	10045#	10046	10047#
10066	10067#	10068	10069#	10086	10087#	10088	10089#	10106	10107#	10108	10109#	10126
10127#	10128	10129#	10153	10154#	10160	10161#	10211	10212#	10214	10215#	10253	10254#
10255#	10258	10259#	10297	10298#	10313	10315#	10326	10327#	10342	10344#	10355	10356#
10373	10375#	10419	10420#	10443	10445	10447	10448#	10460	10461#	10464	10466#	10483
10484#	10501	10502#	10521	10522#	10525	10527#	10545	10546#	10549	10551#	10566	10567#
10590	10592	10594	10595#	10607	10608#	10611	10613#	10630	10631#	10648	10649#	10667
10668#	10671	10673#	10691	10692#	10695	10697#	10712	10713#	10736	10738	10740	10741#
10753	10754#	10756	10757#	10760	10762#	10779	10780#	10786	10787#	10804	10805#	10811
10812#	10832	10833#	10836	10838#	10856	10857#	10860	10862#	10911	10912#	10918	10919#
10925	10926#	10932	10933#	10939	10940#	10946	10947#	10953	10954#	10970	10971#	10972
10974	10975#	10977	10978#	11007	11008#	11087	11088#	11099	11101	11103	11104	11105#
11111	11113	11115	11116	11117#	11133	11134#	11135	11137	11138#	11140	11141#	11174
11175#	11176#	11179	11180#	11222	11223#	11238	11240#	11251	11252#	11267	11269#	11280
11281#	11298	11300#	11344	11345#	11368	11370	11372	11373#	11385	11386#	11389	11391#
11408	11409#	11426	11427#	11446	11447#	11450	11452#	11470	11471#	11474	11476#	11491
11492#	11517	11519	11521	11522#	11534	11535#	11538	11540#	11557	11558#	11575	11576#
11594	11595#	11598	11600#	11618	11619#	11622	11624#	11639	11640#	11663	11665	11667
11668#	11680	11681#	11683	11684#	11687	11689#	11706	11707#	11713	11714#	11731	11732#
11738	11739#	11759	11760#	11763	11765#	11783	11784#	11787	11789#	11838	11839#	11845
11846#	11852	11853#	11859	11860#	11866	11867#	11873	11874#	11880	11881#	11898	11899#
11900	11902	11903#	11905	11906#	11935	11936#	12015	12016#	12027	12029	12031	12032
12033#	12039	12041	12043	12044	12045#	12061	12062#	12063	12065	12066#	12068	12069#
12135	12136#	12137#	12140	12141#	12148	12149#	12162	12163#	12164#	12167	12168#	12183
12185	12186#	12189	12191#	12193	12195	12196#	12199	12201#	12203	12205	12206#	12216
12218	12219#	12221	12222#	12224	12225#	12230	12232#	12236	12238	12239#	12242	12244#
12246	12248	12249#	12257	12258#	12270	12271#	12272	12274	12275#	12282	12283#	12285
12287	12288#	12291	12293#	12295	12297	12298#	12305	12307#	12333	12334#	12349	12350#
12355	12356#	12359	12361#	12373	12374#	12376	12377#	12382	12383#	12386	12388#	12402
12403#	12405	12406#	12410	12411#	12414	12416#	12430	12431#	12436	12437#	12440	12442#
12450	12451#	12455	12456#	12459	12461#	12479	12480#	12495	12496#	12501	12502#	12505
12507#	12519	12520#	12525	12526#	12529	12531#	12543	12544#	12548	12549#	12552	12554#
12566	12567#	12572	12573#	12576	12578#	12586	12587#	12591	12592#	12595	12597#	12640
12641#	12647	12648#	12654	12655#	12661	12662#	12668	12669#	12675	12676#	12682	12683#
12692	12693#	12702	12703#	12715	12716#	12750	12751#	12757	12759	12761	12762	12763#
12769	12771	12773	12774	12775#	12847	12848#	12849	12850#	12868	12869#	12880	12882
12883#	12892	12893#	12897	12898#	12899#	12902	12903#	12912	12914	12915	12916#	12919
12921#	12926	12927#	12939	12940#	12942	12943#	12946	12948#	12956	12957#	12962	12963#
12964	12966	12967#	12971	12973	12974#	12987	12989	12990#	13001	13002#	13007	13008#
13014	13015#	13021	13022#	13023#	13026	13027#	13033	13035#	13040	13041#	13042#	13045
13046#	13054	13055#	13056	13058	13059#	13074	13075#	13110	13111#	13119	13120#	13132
13133#	13145	13146#	13150	13151#	13156	13157#	13160	13162#	13172	13173#	13178	13180
13182	13183	13184#	13187	13189#	13195	13197	13199	13200#	13210	13211#	13215	13216#
13225	13226#	13237	13238#	13240	13241#	13243	13245	13246#	13255	13256#	13259	13261#
13269	13270#	13275	13277	13279	13280	13281#	13284	13286#	13292	13294	13296	13297#
13313	13314#	13316	13317#	13319	13320#	13323	13325#	13333	13334#	13336	13338	13340

13341	13342#	13345	13347#	13353	13355	13357	13358#	13374	13375#	13383	13384#	13387	
13389#	13401	13403	13404	13405#	13408	13410#	13416	13418	13420	13421#	13427	13428#	
13436	13437#	13452	13453#	13456	13458#	13464	13466	13467	13468#	13471	13473#	13479	
13481	13483	13484#	13490	13491#	13529	13530#	13536	13538	13540	13541	13542#	13551	
13552#	13558	13559#	13573	13574#	13575	13577	13578#	13580	13581#	13592	13593#	13605	
13606#	13607#	13610	13611#	13621	13622#	13631	13632#	13683	13684#	13690	13692	13694	
13695	13696#	13702	13704	13706	13707	13708#	13717	13718#	13724	13725#	13732	13733#	
13734	13736	13737#	13739	13740#	13801	13802#	13804	13805#	13843	13844#	13845#	13848	
13849#	13960	13961#	13967	13968#	13974	13975#	13981	13982#	13988	13989#	13995	13996#	
14002	14003#	14019	14020#	14021	14023	14024#	14026	14027#	14041	14042#	14128	14129#	
14140	14142	14144	14145	14146#	14152	14154	14156	14157	14158#	14183	14184#	14185	
14187	14188#	14190	14191#	14224	14225#	14226#	14229	14230#	14346	14347#	14353	14354#	
14360	14361#	14367	14368#	14374	14375#	14381	14382#	14388	14389#	14406	14407#	14408	
14410	14411#	14413	14414#	14428	14429#	14516	14517#	14528	14530	14532	14533	14534#	
14540	14542	14544	14545	14546#	14571	14572#	14573	14575	14576#	14578	14579#	14646	
14647#	14648#	14651	14652#	14668	14669#	14670#	14673	14674#	14689	14691	14692#	14695	
14697#	14699	14701	14702#	14705	14707#	14709	14711	14712#	14722	14724	14725#	14727	
14728#	14730	14731#	14736	14738#	14742	14744	14745#	14748	14750#	14752	14754	14755#	
14763	14764#	14776	14777#	14778	14780	14781#	14788	14789#	14791	14793	14794#	14797	
14799#	14801	14803	14804#	14811	14813#	14843	14844#	14846	14847#	14850	14852#	14857	
14859#	14861	14862#	14869	14870#	14871	14872#	14891	14892#	14895	14897#	14915	14916#	
14918	14919#	14922	14924#	14929	14931#	14933	14934#	14941	14942#	14943	14944#	14963	
14964#	14967	14969#	15009	15010#	15016	15018	15020	15021	15022#	15028	15030	15032	
15033	15034#	15045	15046#	15081	15082#	15088	15090	15092	15093	15094#	15100	15102	
15104	15105	15106#	15179	15180#	15182	15183#	15186	15188#	15193	15195#	15197	15198#	
15205	15206#	15207	15208#	15217	15218#	15221	15223#	15240	15241#	15252	15254	15255#	
15270	15272	15273	15274#	15277	15279#	15288	15290	15291	15292#	15298	15300	15302	
15303#	15321	15322#	15324	15325#	15328	15330#	15348	15349#	15364	15365#	15366	15368	
15369#	15371	15372#	15376	15378	15379#	15386	15388#	15401	15403	15404#	15412	15414	
15416	15417#	15423	15424#	15429	15430#	15436	15437#	15442	15444#	15464	15465#	15466	
15468	15469#	15471	15472#	15477	15479#	15528	15529#	15535	15537	15538	15539#	15545	
15547	15548	15549#	15558	15559#	15565	15566#	15581	15582#	15583	15585	15586#	15588	
15589#	15600	15601#	15610	15611#	15620	15621#	15656	15657#	15663	15665	15667	15668	
15669#	15675	15677	15679	15680	15681#	15690	15691#	15697	15698#	15705	15706#	15707	
15709	15710#	15712	15713#	15773	15774#	15776	15777#	15816	15817#	15818#	15821	15822#	
15933	15934#	15940	15941#	15947	15948#	15954	15955#	15961	15962#	15968	15969#	15975	
15976#	15993	15994#	15995	15997	15998#	16000	16001#	16015	16016#	16100	16101#	16112	
16114	16116	16117	16118#	16124	16126	16128	16129	16130#	16155	16156#	16157	16159	
16160#	16162	16163#	16206	16207#	16209	16210#	16249	16250#	16251#	16254	16255#	16371	
16372#	16378	16379#	16385	16386#	16392	16393#	16399	16400#	16406	16407#	16413	16414#	
16431	16432#	16433	16435	16436#	16438	16439#	16453	16454#	16542	16543#	16554	16556	
16558	16559	16560#	16566	16568	16570	16571	16572#	16597	16598#	16599	16601	16602#	
16604	16605#	16860	16862	16863#	16978#	17007#	17016	17017#	17044#	17053	17054#	17081#	
17090	17091#	17116#	17125	17126#	17154#	17163	17164#	17190#	17217#	17246#	17263	17264#	
17267	17269#	17271	17272#	17285	17286#	17328	17329#	17335	17337#	17389#	17403	17404#	
17407	17409#	17423	17424#	17427	17429#	17431	17432#	17439	17440#	17441	17442#		
	720#	1264#	1266#	1275#	1277#	1280#	1281#	1282#	1283#	1284#	1285#	1298#	1299#
	1300#	1301#	1315#	1316#	1321#	1322#	1324#	1325#	1330#	1332#	1333#	1334#	1335#
	1336#	1337#	1351#	1353#	1354#	1361#	1363#	1370#	1377#	1384#	1391#	1398#	1408#
	1410#	1411#	1418#	1420#	1422#	1423#	1424#	1425	1432#	1433#	1434#	1435#	1453#
	1460#	1467#	1474#	1481#	1488#	1518#	1519#	1520#	1521#	1535#	1536#	1541#	1542#
	1544#	1545#	1550#	1552#	1553#	1554#	1555#	1556#	1557#	1571#	1573#	1574#	1581#
	1583#	1590#	1597#	1604#	1611#	1618#	1625#	1627#	1628#	1629#	1630	1656#	1663#
	1670#	1677#	1684#	1691#	1698#	1713#	1714#	1715#	1716#	1717#	1718#	1727#	1728#
	1729#	1730#	1731#	1732#	1733#	1734#	1735#	1736#	1737#	1738#	1739#	1740#	1741#
	1742#	1750#	1751#	1752#	1753#	1754#	1755#	1756#	1757#	1758#	1759#	1763#	1764#

\$TEMP - 000300

1765#	1766#	1768#	1769#	1770#	1774#	1775#	1776#	1777#	1784#	1785#	1786#	1787#
1789#	1791#	1792#	1794#	1795#	1796#	1797#	1798#	1799#	1800#	1801#	1802#	1803#
1804#	1805#	1811#	1812#	1819#	1820#	1824#	1826#	1828#	1829#	1835#	1836#	1843#
1844#	1848#	1850#	1855#	1856#	1857#	1858#	1860#	1861#	1862#	1876#	1877#	1878#
1879#	1885#	1892#	1899#	1906#	1913#	1920#	1927#	1934#	1935#	1936#	1937#	1938#
1939#	1943#	1944#	1956#	1958#	1959#	1960#	1961#	1963#	1964#	1966#	1967#	1979#
1981#	1982#	1983#	1984#	1986#	1988#	1989#	1991#	1999#	2001#	2002#	2003#	2006
2009#	2011#	2012#	2019#	2020#	2021#	2022#	2024#	2025#	2026#	2028#	2030#	2031#
2033#	2041#	2043#	2046	2049#	2051#	2052#	2053#	2054#	2061#	2069#	2070#	2073#
2074#	2077#	2078#	2079#	2080#	2081#	2082#	2083#	2084#	2085#	2086#	2087#	2088#
2089#	2090#	2105#	2107#	2114#	2121#	2128#	2135#	2142#	2149#	2156#	2163#	2165#
2166#	2167#	2169#	2177#	2179#	2182	2330#	2331#	2332#	2333#	2334#	2335#	2344#
2345#	2346#	2347#	2348#	2349#	2350#	2351#	2352#	2353#	2354#	2355#	2356#	2357#
2358#	2359#	2366#	2367#	2368#	2369#	2370#	2371#	2372#	2373#	2374#	2375#	2379#
2380#	2381#	2382#	2384#	2385#	2386#	2390#	2391#	2392#	2393#	2400#	2401#	2402#
2403#	2405#	2407#	2408#	2410#	2411#	2412#	2413#	2414#	2415#	2416#	2417#	2418#
2419#	2420#	2421#	2427#	2428#	2435#	2436#	2440#	2442#	2444#	2445#	2451#	2452#
2459#	2460#	2464#	2466#	2468#	2469#	2482#	2483#	2484#	2485#	2491#	2498#	2505#
2512#	2519#	2526#	2533#	2540#	2541#	2542#	2543#	2547#	2548#	2549#	2551#	2552#
2564#	2566#	2567#	2568#	2569#	2571#	2572#	2581#	2583#	2585#	2586#	2587#	2590
2593#	2594#	2595#	2597#	2598#	2599#	2601#	2613#	2615#	2616#	2618#	2623#	2625#
2633#	2635#	2637#	2638#	2641	2644#	2646#	2647#	2648#	2649#	2656#	2664#	2665#
2668#	2669#	2672#	2673#	2674#	2675#	2676#	2677#	2678#	2679#	2680#	2681#	2682#
2683#	2684#	2685#	2700#	2702#	2709#	2716#	2723#	2730#	2737#	2744#	2751#	2758#
2760#	2761#	2762#	2763#	2771#	2773#	2776	2834#	2835#	2836#	2837#	2838#	2839#
2848#	2849#	2850#	2851#	2852#	2853#	2854#	2855#	2856#	2857#	2858#	2859#	2860#
2861#	2862#	2863#	2871#	2872#	2873#	2874#	2875#	2876#	2877#	2878#	2879#	2880#
2881#	2882#	2885#	2886#	2887#	2888#	2895#	2896#	2897#	2898#	2900#	2902#	2903#
2905#	2906#	2907#	2908#	2909#	2910#	2911#	2912#	2913#	2914#	2915#	2916#	2922#
2923#	2930#	2931#	2935#	2937#	2939#	2940#	2946#	2947#	2954#	2955#	2959#	2961#
2963#	2964#	2974#	2977#	2978#	2979#	2980#	2981#	2983#	2984#	2985#	2986#	2987#
2988#	2989#	2990#	3003#	3004#	3005#	3006#	3012#	3019#	3026#	3033#	3040#	3047#
3054#	3061#	3062#	3063#	3064#	3065#	3066#	3067#	3068#	3069#	3070#	3074#	3075#
3087#	3089#	3090#	3091#	3092#	3094#	3095#	3097#	3098#	3111#	3113#	3114#	3115#
3116#	3118#	3120#	3121#	3123#	3131#	3133#	3134#	3135#	3138	3141#	3146#	3148#
3149#	3150#	3152#	3160#	3162#	3165	3168#	3170#	3171#	3172#	3173#	3180#	3188#
3189#	3192#	3193#	3196#	3197#	3198#	3199#	3200#	3201#	3202#	3203#	3204#	3205#
3206#	3207#	3208#	3209#	3226#	3228#	3235#	3242#	3249#	3256#	3263#	3270#	3277#
3284#	3286#	3288#	3289#	3290#	3298#	3300#	3303	3361#	3362#	3363#	3364#	3365#
3366#	3374#	3375#	3376#	3377#	3378#	3379#	3380#	3381#	3388#	3389#	3390#	3391#
3392#	3393#	3398#	3399#	3400#	3401#	3402#	3403#	3404#	3405#	3407#	3408#	3409#
3410#	3411#	3412#	3413#	3418#	3419#	3420#	3422#	3423#	3424#	3426#	3428#	3429#
3431#	3432#	3434#	3435#	3436#	3437#	3438#	3439#	3440#	3441#	3444#	3445#	3446#
3447#	3448#	3449#	3452#	3454#	3455#	3456#	3469#	3470#	3471#	3472#	3478#	3485#
3492#	3499#	3506#	3513#	3520#	3522#	3523#	3524#	3525#	3532#	3540#	3541#	3546#
3548#	3549#	3550#	3557#	3558#	3565#	3567#	3568#	3574#	3581#	3588#	3595#	3602#
3609#	3645#	3646#	3647#	3648#	3649#	3650#	3659#	3660#	3661#	3662#	3663#	3664#
3665#	3666#	3674#	3675#	3676#	3677#	3678#	3679#	3684#	3685#	3686#	3687#	3688#
3689#	3690#	3691#	3693#	3694#	3695#	3696#	3697#	3698#	3699#	3704#	3705#	3706#
3708#	3709#	3710#	3712#	3714#	3715#	3717#	3718#	3720#	3721#	3722#	3723#	3724#
3725#	3726#	3727#	3730#	3731#	3732#	3733#	3734#	3735#	3738#	3740#	3741#	3742#
3755#	3756#	3757#	3758#	3764#	3771#	3778#	3785#	3792#	3799#	3806#	3808#	3809#
3810#	3811#	3818#	3826#	3827#	3832#	3834#	3835#	3836#	3843#	3844#	3851#	3853#
3855#	3861#	3868#	3875#	3882#	3889#	3896#	3929#	3930#	3931#	3932#	3933#	3934#
3943#	3944#	3945#	3946#	3947#	3948#	3956#	3957#	3958#	3959#	3960#	3961#	3962#
3963#	3968#	3969#	3970#	3971#	3972#	3973#	3975#	3976#	3977#	3978#	3979#	3984#

3985#	3986#	3988#	3989#	3990#	3992#	3994#	3995#	3997#	3998#	4000#	4001#	4002#
4003#	4004#	4005#	4006#	4007#	4010#	4011#	4012#	4013#	4014#	4015#	4018#	4020#
4021#	4022#	4029#	4030#	4032#	4033#	4034#	4036#	4037#	4038#	4051#	4052#	4053#
4054#	4060#	4067#	4074#	4081#	4088#	4095#	4102#	4104#	4105#	4106#	4107#	4114#
4122#	4123#	4126#	4128#	4129#	4130#	4131#	4132#	4139#	4140#	4142#	4144#	4145#
4146#	4155#	4156#	4163#	4165#	4166#	4172#	4179#	4186#	4193#	4200#	4207#	4242#
4243#	4244#	4245#	4246#	4247#	4256#	4257#	4258#	4259#	4260#	4261#	4269#	4270#
4271#	4272#	4273#	4274#	4275#	4276#	4281#	4282#	4283#	4284#	4285#	4286#	4288#
4289#	4290#	4291#	4292#	4297#	4298#	4299#	4301#	4302#	4303#	4305#	4307#	4308#
4310#	4311#	4313#	4314#	4315#	4316#	4317#	4318#	4319#	4320#	4323#	4324#	4325#
4326#	4327#	4328#	4331#	4333#	4334#	4335#	4342#	4343#	4345#	4346#	4347#	4348#
4361#	4362#	4363#	4364#	4370#	4377#	4384#	4391#	4398#	4405#	4412#	4414#	4415#
4416#	4417#	4424#	4432#	4433#	4436#	4438#	4439#	4440#	4441#	4442#	4447#	4448#
4449#	4450#	4454#	4455#	4462#	4463#	4470#	4472#	4473#	4479#	4486#	4493#	4500#
4507#	4514#	4549#	4550#	4551#	4552#	4553#	4554#	4563#	4564#	4565#	4566#	4567#
4568#	4569#	4570#	4576#	4577#	4578#	4579#	4580#	4581#	4582#	4583#	4584#	4585#
4586#	4587#	4592#	4593#	4594#	4595#	4597#	4598#	4599#	4606#	4607#	4610#	4611#
4677#	4619#	4620#	4621#	4622#	4623#	4625#	4627#	4628#	4629#	4630#	4632#	4633#
4639#	4641#	4642#	4644#	4645#	4647#	4648#	4650#	4651#	4652#	4653#	4654#	4658#
4659#	4660#	4662#	4663#	4668#	4669#	4670#	4672#	4673#	4680#	4681#	4682#	4696#
4697#	4698#	4699#	4705#	4712#	4719#	4726#	4733#	4740#	4747#	4749#	4750#	4751#
4752#	4759#	4767#	4768#	4771#	4772#	4777#	4779#	4780#	4781#	4790#	4791#	4792#
4793#	4799#	4806#	4813#	4820#	4827#	4834#	4841#	4899#	4900#	4901#	4902#	4903#
4904#	4913#	4914#	4915#	4916#	4917#	4918#	4924#	4925#	4926#	4927#	4928#	4929#
4930#	4931#	4936#	4937#	4938#	4939#	4940#	4941#	4943#	4944#	4945#	4946#	4947#
4954#	4955#	4958#	4959#	4965#	4966#	4971#	4973#	4974#	4975	4977#	4978#	4979#
4986#	4987#	4988#	4989#	4990#	4991#	4992#	4993#	4994#	4995#	4996#	4997#	5001#
5003#	5007#	5008#	5009#	5011#	5012#	5013#	5015#	5017#	5018#	5020#	5021#	5023#
5027#	5028#	5029#	5030#	5043#	5044#	5045#	5046#	5052#	5059#	5066#	5073#	5080#
5087#	5094#	5096#	5097#	5098#	5099#	5106#	5114#	5115#	5118#	5119#	5122#	5123#
5128#	5129	5139#	5140#	5147#	5154#	5161#	5168#	5175#	5182#	5189#	5196#	5246#
5248#	5251#	5252#	5253#	5254#	5255#	5256#	5257#	5258#	5259#	5260#	5261#	5262#
5268#	5269#	5270#	5271#	5272#	5273#	5283#	5284#	5285#	5286#	5287#	5288#	5289#
5290#	5291#	5292#	5293#	5294#	5296#	5297#	5298#	5299#	5300#	5301#	5302#	5303#
5304#	5305#	5306#	5307#	5309#	5310#	5311#	5312#	5313#	5314#	5315#	5316#	5317#
5318#	5319#	5320#	5321#	5322#	5326#	5327#	5339#	5342#	5343#	5344#	5345#	5346#
5348#	5349#	5350#	5351#	5352#	5353#	5354#	5355#	5361#	5362#	5364#	5365#	5372#
5373#	5374#	5375#	5376#	5377#	5386#	5389	5397#	5398#	5404#	5405#	5406#	5407#
5413#	5420#	5427#	5434#	5441#	5448#	5455#	5462#	5463#	5464#	5465#	5466#	5467#
5468#	5469#	5470#	5471#	5472#	5473#	5485#	5487#	5488#	5489#	5490#	5492#	5494#
5495#	5497#	5505#	5507#	5508#	5509#	5512	5518#	5520#	5521#	5523#	5531#	5533#
5536	5539#	5541#	5543#	5544#	5545#	5546#	5547#	5548#	5549#	5550#	5553#	5554#
5566#	5567#	5568#	5569#	5573#	5574#	5575#	5576#	5578#	5579#	5580#	5598#	5599#
5600#	5601#	5602#	5603#	5604#	5605#	5606#	5607#	5608#	5609#	5614#	5615#	5631#
5633#	5634#	5635#	5636#	5637#	5638#	5675#	5677#	5686#	5688#	5690#	5691#	5692#
5693#	5694#	5695#	5696#	5697#	5708#	5709#	5710#	5711#	5712#	5713#	5722#	5723#
5724#	5725#	5726#	5727#	5728#	5729#	5737#	5738#	5739#	5740#	5743#	5744#	5745#
5747#	5751#	5752#	5753#	5754#	5758#	5759#	5760#	5762#	5770#	5771#	5774#	5776#
5777#	5778#	5779#	5780#	5781#	5782#	5787#	5789#	5790#	5791#	5792#	5793#	5798#
5800#	5801#	5802#	5803#	5804#	5805#	5806#	5807#	5808#	5813#	5814#	5817#	5818#
5821#	5823#	5824#	5825#	5827#	5828#	5829#	5838#	5839#	5840#	5841#	5842#	5844#
5845#	5850#	5851#	5854#	5856#	5857#	5858#	5859#	5860#	5861#	5862#	5864#	5866#
5867#	5873#	5875#	5876#	5877#	5882#	5884#	5885#	5886#	5888#	5890#	5891#	5892#
5893#	5895#	5897#	5899#	5906#	5907#	5908#	5909#	5910#	5911#	5912#	5913#	5917#
5918#	5919#	5920#	5921#	5922#	5924#	5925#	5926#	5927#	5928#	5930#	5931#	5932#
5933#	5934#	5935#	5936#	5937#	5938#	5939#	5940#	5941#	5955#	5956#	5957#	5958#

5964#	5971#	5978#	5985#	5992#	5999#	6006#	6014#	6015#	6016#	6017#	6018#	6019#
6031#	6033#	6034#	6036#	6037#	6038#	6039#	6046#	6063#	6064#	6065#	6066#	6072#
6079#	6086#	6093#	6100#	6107#	6114#	6116#	6117#	6123#	6132#	6134#	6135#	6142#
6144#	6145#	6153#	6155#	6158#	6237#	6238#	6239#	6240#	6241#	6242#	6251#	6252#
6253#	6254#	6255#	6256#	6257#	6258#	6266#	6267#	6268#	6269#	6272#	6273#	6274#
6276#	6277#	6279#	6283#	6284#	6288#	6289#	6290#	6298#	6299#	6300#	6301#	6303#
6304#	6305#	6306#	6307#	6309#	6317#	6318#	6319#	6320#	6322#	6323#	6324#	6325#
6326#	6328#	6330#	6337#	6338#	6339#	6340#	6345#	6346#	6347#	6348#	6349#	6350#
6352#	6353#	6354#	6361#	6363#	6364#	6365#	6366#	6370#	6371#	6372#	6373#	6374#
6379#	6380#	6381#	6382#	6384#	6389#	6390#	6391#	6392#	6393#	6394#	6398#	6400#
6401#	6402#	6403#	6404#	6411#	6412#	6413#	6414#	6415#	6416#	6417#	6428#	6429#
6430#	6431#	6433#	6434#	6435#	6437#	6439#	6443#	6444#	6445#	6446#	6447#	6449#
6450#	6451#	6462#	6464#	6465#	6470#	6471#	6472#	6473#	6474#	6475#	6477#	6478#
6479#	6481#	6483#	6490#	6491#	6492#	6493#	6494#	6495#	6496#	6497#	6501#	6502#
6503#	6504#	6505#	6506#	6508#	6509#	6510#	6511#	6512#	6514#	6515#	6516#	6517#
6518#	6519#	6520#	6521#	6522#	6523#	6524#	6525#	6539#	6540#	6541#	6542#	6548#
6555#	6562#	6569#	6576#	6583#	6590#	6598#	6599#	6600#	6601#	6602#	6603#	6615#
6617#	6618#	6620#	6621#	6622#	6623#	6630#	6647#	6648#	6649#	6650#	6656#	6663#
6670#	6677#	6684#	6691#	6698#	6700#	6701#	6707#	6709#	6710#	6718#	6720#	6723#
6801#	6802#	6803#	6804#	6805#	6806#	6815#	6816#	6817#	6818#	6819#	6820#	6826#
6827#	6828#	6829#	6833#	6834#	6835#	6836#	6837#	6838#	6840#	6841#	6842#	6843#
6844#	6846#	6847#	6848#	6849#	6857#	6858#	6859#	6860#	6861#	6862#	6863#	6865#
6869#	6870#	6871#	6872#	6873#	6874#	6875#	6876#	6878#	6880#	6881#	6888#	6889#
6890#	6891#	6892#	6893#	6894#	6900#	6901#	6910#	6911#	6912#	6913#	6914#	6916#
6917#	6918#	6919#	6920#	6921#	6922#	6924#	6925#	6934#	6935#	6939#	6941#	6945#
6947#	6954#	6955#	6956#	6957#	6958#	6959#	6963#	6964#	6965#	6973#	6974#	6975#
6977#	6979#	6981#	6988#	6989#	6990#	6991#	6992#	6993#	6994#	6995#	7009#	7010#
7011#	7012#	7018#	7025#	7032#	7039#	7046#	7053#	7060#	7063#	7064#	7065#	7066#
7073#	7089#	7090#	7091#	7092#	7098#	7105#	7112#	7119#	7126#	7133#	7140#	7147#
7154#	7226#	7227#	7228#	7229#	7230#	7231#	7240#	7241#	7242#	7243#	7244#	7245#
7251#	7252#	7253#	7254#	7258#	7259#	7260#	7261#	7262#	7263#	7265#	7266#	7267#
7268#	7269#	7271#	7272#	7273#	7274#	7280#	7281#	7282#	7283#	7284#	7285#	7286#
7287#	7288#	7290#	7294#	7295#	7296#	7297#	7298#	7299#	7300#	7301#	7303#	7305#
7306#	7307#	7308#	7311#	7312#	7313#	7314#	7315#	7316	7318#	7320#	7322#	7333#
7334#	7335#	7337#	7341#	7342#	7343#	7344#	7345#	7347#	7358#	7359#	7360#	7361#
7363#	7365#	7366#	7367#	7369#	7370#	7371#	7372#	7375#	7376#	7377#	7378#	7379#
7380#	7382#	7383#	7387#	7388#	7389#	7390#	7391#	7393#	7395#	7396#	7401#	7403#
7404#	7415#	7416#	7417#	7418#	7419#	7421#	7423#	7430#	7433#	7434#	7435#	7436#
7437#	7445#	7446#	7453#	7454#	7455#	7457#	7464#	7465#	7466#	7467#	7468#	7469#
7470#	7471#	7485#	7486#	7487#	7488#	7494#	7501#	7508#	7515#	7522#	7529#	7536#
7539#	7540#	7541#	7542#	7549#	7565#	7566#	7567#	7568#	7574#	7586#	7598#	7605#
7612#	7973#	7974#	7975#	7976#	7977#	7978#	7987#	7988#	7989#	7990#	7991#	7992#
7993#	7994#	8000#	8001#	8002#	8003#	8004#	8005#	8009#	8010#	8011#	8012#	8013#
8014#	8016#	8017#	8018#	8019#	8020#	8022#	8023#	8024#	8025#	8034#	8035#	8043#
8045#	8046#	8047#	8054#	8055#	8058#	8059#	8060#	8062#	8069#	8070#	8071#	8072#
8077#	8078#	8079#	8080#	8082#	8083#	8084#	8086#	8098#	8099#	8100#	8101#	8103#
8104#	8105#	8115#	8117#	8118#	8119#	8120#	8123#	8124#	8132#	8133#	8136#	8138#
8139#	8140#	8141#	8155#	8156#	8157#	8158#	8159#	8161#	8162#	8163#	8164#	8168#
8169#	8170#	8171#	8174#	8176#	8177#	8178#	8179#	8180#	8182#	8183#	8194#	8195#
8196#	8197#	8210#	8211#	8212#	8227#	8228#	8229#	8240#	8241#	8242#	8251#	8252#
8253#	8255#	8256#	8257#	8258#	8277#	8278#	8283#	8284#	8285#	8286#	8291#	8292#
8300#	8305#	8306#	8310#	8312#	8313#	8314#	8315#	8322#	8323#	8327#	8329#	8333#
8334#	8335#	8336#	8338#	8339#	8340#	8342#	8344#	8354#	8356#	8357#	8358#	8359#
8361#	8362#	8363#	8375#	8376#	8377#	8387#	8388#	8389#	8391#	8393#	8394#	8396#
8400#	8401#	8402#	8404#	8405#	8406#	8408#	8418#	8420#	8424#	8425#	8426#	8428#
8429#	8430#	8432#	8439#	8441#	8445#	8446#	8447#	8448#	8450#	8451#	8452#	8454#

8461#	8463#	8464#	8465#	8469#	8470#	8471#	8472#	8474#	8475#	8476#	8478#	8485#
8486#	8487#	8488#	8489#	8491#	8492#	8493#	8494#	8495#	8504#	8505#	8506#	8507#
8513#	8520#	8527#	8534#	8541#	8548#	8555#	8559#	8560#	8561#	8562#	8563#	8564#
8576#	8578#	8579#	8587#	8588#	8589#	8590#	8597#	8613#	8614#	8615#	8616#	8622#
8634#	8646#	8653#	8660#	8667#	8674#	8772#	8773#	8774#	8775#	8776#	8777#	8786#
8787#	8788#	8789#	8790#	8791#	8792#	8793#	8802#	8803#	8810#	8811#	8821#	8822#
8823#	8824#	8825#	8826#	8827#	8829#	8834#	8837#	8838#	8839#	8840#	8841#	8849#
8851#	8861#	8862#	8866#	8867#	8868#	8869#	8871#	8872#	8873#	8875#	8877#	8888#
8889#	8890#	8891#	8892#	8894#	8905#	8906#	8907#	8908#	8912#	8913#	8914#	8916#
8917#	8918#	8920#	8931#	8932#	8936#	8937#	8938#	8940#	8941#	8942#	8944#	8955#
8956#	8957#	8958#	8960#	8961#	8962#	8964#	8975#	8976#	8977#	8978#	8980#	8981#
8982#	8984#	8993#	8995#	8998#	9000#	9001#	9003#	9010#	9012#	9013#	9015#	9019#
9020#	9021#	9022#	9030#	9031#	9032#	9033#	9035#	9036#	9037#	9039#	9044#	9045#
9046#	9064#	9065#	9066#	9068#	9082#	9083#	9084#	9095#	9096#	9097#	9109#	9110#
9111#	9112#	9114#	9115#	9116#	9125#	9126#	9129#	9130#	9131#	9133#	9134#	9145#
9146#	9149#	9150#	9151#	9152#	9153#	9168#	9169#	9170#	9171#	9171#	9172#	9175#
9176#	9180#	9181#	9182#	9183#	9186#	9187#	9188#	9189#	9190#	9190#	9193#	9202#
9203#	9204#	9205#	9206#	9207#	9208#	9209#	9210#	9211#	9212#	9213#	9214#	9215#
9216#	9217#	9218#	9219#	9220#	9221#	9222#	9223#	9224#	9225#	9233#	9234#	9235#
9236#	9242#	9249#	9256#	9263#	9270#	9277#	9284#	9288#	9289#	9290#	9291#	9292#
9293#	9305#	9307#	9308#	9310#	9311#	9312#	9313#	9320#	9337#	9338#	9339#	9340#
9346#	9358#	9370#	9377#	9384#	9391#	9542#	9543#	9544#	9545#	9555#	9556#	9557#
9559#	9561#	9573#	9574#	9576#	9577#	9578#	9579#	9580#	9581#	9582#	9583#	9584#
9585#	9586#	9587#	9588#	9589#	9590#	9591#	9592#	9593#	9594#	9595#	9596#	9597#
9598#	9599#	9603#	9605#	9616#	9617#	9618#	9619#	9623#	9624#	9625#	9627#	9629#
9630#	9640#	9641#	9644#	9645#	9646#	9647#	9648#	9649#	9656#	9657#	9658#	9659#
9660#	9661#	9662#	9669#	9670#	9675#	9676#	9679#	9680#	9681#	9682#	9683#	9684#
9691#	9692#	9693#	9694#	9695#	9696#	9697#	9704#	9705#	9710#	9711#	9714#	9715#
9716#	9717#	9718#	9719#	9726#	9727#	9728#	9729#	9730#	9731#	9732#	9739#	9740#
9745#	9746#	9749#	9750#	9751#	9752#	9753#	9754#	9761#	9762#	9763#	9764#	9765#
9766#	9767#	9774#	9775#	9780#	9781#	9784#	9785#	9786#	9787#	9788#	9789#	9796#
9797#	9798#	9799#	9800#	9801#	9802#	9809#	9810#	9815#	9816#	9819#	9820#	9821#
9822#	9828#	9829#	9830#	9831#	9832#	9833#	9840#	9841#	9846#	9847#	9848#	9849#
9850#	9851#	9858#	9859#	9864#	9865#	9866#	9867#	9868#	9869#	9876#	9877#	9882#
9883#	9884#	9885#	9886#	9887#	9889#	9890#	9891#	9892#	9898#	9899#	9904#	9905#
9906#	9907#	9909#	9910#	9911#	9913#	9918#	9944#	9945#	9946#	9947#	9948#	9949#
9960#	9961#	9964#	9965#	9966#	9967#	9968#	9969#	9976#	9977#	9982#	9983#	9986#
9987#	9988#	9989#	9990#	9991#	9998#	9999#	10004#	10005#	10008#	10009#	10010#	10011#
10012#	10013#	10020#	10021#	10026#	10027#	10030#	10031#	10032#	10033#	10034#	10035#	10042#
10043#	10048#	10049#	10052#	10053#	10054#	10055#	10056#	10057#	10064#	10065#	10070#	10071#
10074#	10075#	10076#	10077#	10084#	10085#	10090#	10091#	10094#	10095#	10096#	10097#	10104#
10105#	10110#	10111#	10114#	10115#	10116#	10117#	10124#	10125#	10130#	10131#	10134#	10135#
10136#	10137#	10144#	10145#	10148#	10149#	10155#	10156#	10157#	10158#	10162#	10163#	10164#
10166#	10170#	10207#	10209#	10218#	10220#	10222#	10223#	10224#	10225#	10226#	10227#	10228#
10229#	10241#	10242#	10243#	10244#	10245#	10246#	10256#	10259#	10262#	10263#	10275#	10276#
10277#	10278#	10279#	10280#	10281#	10282#	10283#	10284#	10285#	10286#	10287#	10288#	10289#
10290#	10299#	10300#	10301#	10302#	10303#	10304#	10305#	10306#	10313#	10314#	10316#	10317#
10318#	10328#	10329#	10330#	10331#	10332#	10333#	10334#	10335#	10342#	10343#	10345#	10346#
10347#	10357#	10358#	10359#	10360#	10361#	10362#	10363#	10364#	10373#	10374#	10376#	10377#
10378#	10385#	10386#	10387#	10389#	10390#	10391#	10392#	10393#	10394#	10396#	10397#	10398#
10399#	10400#	10401#	10403#	10404#	10405#	10406#	10407#	10408#	10409#	10410#	10411#	10423#
10430#	10431#	10432#	10434#	10435#	10436#	10443#	10448#	10449#	10450#	10451#	10452#	10453#
10455#	10457#	10458#	10462#	10463#	10464#	10465#	10467#	10468#	10469#	10471#	10472#	10474#
10475#	10477#	10479#	10480#	10481#	10485#	10486#	10487#	10489#	10490#	10492#	10493#	10495#
10497#	10498#	10499#	10503#	10504#	10505#	10507#	10508#	10511#	10512#	10514#	10516#	10517#
10519#	10523#	10524#	10525#	10526#	10528#	10529#	10530#	10533#	10534#	10536#	10538#	10539#

10541#	10542#	10543#	10547#	10548#	10549#	10550#	10552#	10553#	10554#	10556#	10558#	10570#
10577#	10578#	10579#	10581#	10582#	10583#	10590#	10595#	10596#	10597#	10598#	10599#	10600#
10602#	10604#	10605#	10609#	10610#	10611#	10612#	10614#	10615#	10616#	10618#	10619#	10621#
10622#	10624#	10626#	10627#	10628#	10632#	10633#	10634#	10636#	10637#	10639#	10640#	10642#
10644#	10645#	10646#	10650#	10651#	10652#	10654#	10655#	10657#	10658#	10660#	10662#	10663#
10665#	10669#	10670#	10671#	10672#	10674#	10675#	10676#	10679#	10680#	10682#	10684#	10685#
10687#	10688#	10689#	10693#	10694#	10695#	10696#	10698#	10699#	10700#	10702#	10704#	10716#
10723#	10724#	10725#	10727#	10728#	10729#	10736#	10741#	10742#	10743#	10744#	10745#	10746#
10748#	10750#	10751#	10758#	10759#	10760#	10761#	10763#	10764#	10765#	10767#	10768#	10769#
10772#	10773#	10775#	10777#	10781#	10782#	10783#	10784#	10788#	10789#	10790#	10792#	10793#
10794#	10797#	10798#	10800#	10802#	10806#	10807#	10808#	10809#	10813#	10814#	10815#	10817#
10818#	10819#	10822#	10823#	10825#	10827#	10828#	10830#	10834#	10835#	10836#	10837#	10839#
10840#	10841#	10844#	10845#	10847#	10849#	10850#	10852#	10853#	10854#	10858#	10859#	10860#
10861#	10863#	10864#	10865#	10867#	10869#	10875#	10876#	10877#	10878#	10883#	10884#	10885#
10886#	10887#	10888#	10889#	10890#	10891#	10892#	10893#	10894#	10895#	10896#	10897#	10898#
10906#	10907#	10908#	10909#	10915#	10922#	10929#	10936#	10943#	10950#	10957#	10964#	10965#
10966#	10967#	10968#	10969#	10981#	10983#	10984#	11001#	11002#	11003#	11004#	11011#	11025#
11026#	11027#	11029#	11030#	11031#	11041#	11042#	11043#	11045#	11046#	11047#	11057#	11058#
11059#	11061#	11062#	11063#	11082#	11083#	11084#	11085#	11091#	11108#	11120#	11127#	11128#
11129#	11130#	11131#	11132#	11144#	11146#	11147#	11162#	11163#	11164#	11165#	11166#	11167#
11177#	11180#	11183#	11184#	11196#	11197#	11198#	11199#	11200#	11201#	11202#	11203#	11204#
11205#	11206#	11207#	11208#	11209#	11210#	11211#	11212#	11213#	11214#	11215#	11224#	11225#
11226#	11227#	11228#	11229#	11230#	11231#	11238#	11239#	11241#	11242#	11243#	11253#	11254#
11255#	11256#	11257#	11258#	11259#	11260#	11267#	11268#	11270#	11271#	11272#	11282#	11283#
11284#	11285#	11286#	11287#	11288#	11289#	11298#	11299#	11301#	11302#	11303#	11310#	11311#
11312#	11314#	11315#	11316#	11317#	11318#	11319#	11321#	11322#	11323#	11324#	11325#	11326#
11328#	11329#	11330#	11331#	11332#	11333#	11334#	11335#	11336#	11348#	11355#	11356#	11357#
11359#	11360#	11361#	11368#	11373#	11374#	11375#	11376#	11377#	11378#	11380#	11382#	11383#
11387#	11388#	11389#	11390#	11392#	11393#	11394#	11396#	11397#	11399#	11400#	11402#	11404#
11405#	11406#	11410#	11411#	11412#	11414#	11415#	11417#	11418#	11420#	11422#	11423#	11424#
11428#	11429#	11430#	11432#	11433#	11436#	11437#	11439#	11441#	11442#	11444#	11448#	11449#
11450#	11451#	11453#	11454#	11455#	11458#	11459#	11461#	11463#	11464#	11466#	11467#	11468#
11472#	11473#	11474#	11475#	11477#	11478#	11479#	11481#	11483#	11495#	11497#	11498#	11504#
11505#	11506#	11508#	11509#	11510#	11517#	11522#	11523#	11524#	11525#	11526#	11527#	11529#
11531#	11532#	11536#	11537#	11538#	11539#	11541#	11542#	11543#	11545#	11546#	11548#	11549#
11551#	11553#	11554#	11555#	11559#	11560#	11561#	11563#	11564#	11566#	11567#	11569#	11571#
11572#	11573#	11577#	11578#	11579#	11581#	11582#	11584#	11585#	11587#	11589#	11590#	11592#
11596#	11597#	11598#	11599#	11601#	11602#	11603#	11606#	11607#	11609#	11611#	11612#	11614#
11615#	11616#	11620#	11621#	11622#	11623#	11625#	11626#	11627#	11629#	11631#	11643#	11650#
11651#	11652#	11654#	11655#	11656#	11663#	11668#	11669#	11670#	11671#	11672#	11673#	11675#
11677#	11678#	11685#	11686#	11687#	11688#	11690#	11691#	11692#	11694#	11695#	11696#	11699#
11700#	11702#	11704#	11708#	11709#	11710#	11711#	11715#	11716#	11717#	11719#	11720#	11721#
11724#	11725#	11727#	11729#	11733#	11734#	11735#	11736#	11740#	11741#	11742#	11744#	11745#
11746#	11749#	11750#	11752#	11754#	11755#	11757#	11761#	11762#	11763#	11764#	11766#	11767#
11768#	11771#	11772#	11774#	11776#	11777#	11779#	11780#	11781#	11785#	11786#	11787#	11788#
11790#	11791#	11792#	11794#	11796#	11802#	11803#	11804#	11805#	11810#	11811#	11812#	11813#
11814#	11815#	11816#	11817#	11818#	11819#	11820#	11821#	11822#	11823#	11824#	11825#	11833#
11834#	11835#	11836#	11842#	11849#	11856#	11863#	11870#	11877#	11884#	11892#	11893#	11894#
11895#	11896#	11897#	11909#	11911#	11912#	11929#	11930#	11931#	11932#	11939#	11953#	11954#
11955#	11957#	11958#	11959#	11969#	11970#	11971#	11973#	11974#	11975#	11985#	11986#	11987#
11989#	11990#	11991#	12010#	12011#	12012#	12013#	12019#	12036#	12048#	12055#	12056#	12057#
12058#	12059#	12060#	12072#	12074#	12075#	12090#	12091#	12092#	12093#	12094#	12095#	12101#
12102#	12103#	12104#	12105#	12106#	12112#	12113#	12114#	12115#	12116#	12117#	12118#	12119#
12120#	12121#	12127#	12129#	12130#	12131#	12132#	12133#	12138#	12141#	12142#	12143#	12144#
12145#	12150#	12151#	12152#	12154#	12156#	12157#	12158#	12159#	12160#	12165#	12168#	12169#
12170#	12171#	12172#	12174#	12175#	12187#	12188#	12189#	12190#	12197#	12198#	12199#	12200#

12207#	12208#	12209#	12211#	12213#	12226#	12227#	12228#	12230#	12231#	12233#	12234#	12240#
12241#	12242#	12243#	12250#	12251#	12252#	12254#	12264#	12265#	12266#	12267#	12268#	12269#
12276#	12277#	12278#	12279#	12289#	12290#	12291#	12292#	12299#	12300#	12301#	12303#	12305#
12306#	12308#	12309#	12310#	12312#	12314#	12316#	12335#	12336#	12337#	12340#	12341#	12342#
12344#	12345#	12346#	12351#	12353#	12357#	12358#	12359#	12360#	12362#	12363#	12364#	12366#
12367#	12368#	12369#	12370#	12378#	12380#	12384#	12385#	12386#	12387#	12389#	12390#	12391#
12393#	12394#	12395#	12396#	12397#	12399#	12407#	12408#	12412#	12413#	12414#	12415#	12417#
12418#	12419#	12421#	12422#	12423#	12424#	12425#	12427#	12432#	12434#	12438#	12439#	12440#
12441#	12443#	12444#	12445#	12447#	12452#	12453#	12457#	12458#	12459#	12460#	12462#	12463#
12464#	12466#	12468#	12469#	12470#	12471#	12473#	12474#	12481#	12482#	12483#	12486#	12487#
12488#	12490#	12491#	12492#	12497#	12499#	12503#	12504#	12505#	12506#	12508#	12509#	12510#
12512#	12513#	12514#	12515#	12516#	1251#	12523#	12527#	12528#	12529#	12530#	12532#	12533#
12534#	12536#	12537#	12538#	12539#	12540#	12545#	12546#	12550#	12551#	12552#	12553#	12555#
12556#	12557#	12559#	12560#	12561#	12562#	12563#	12568#	12570#	12574#	12575#	12576#	12577#
12579#	12580#	12581#	12583#	12588#	12589#	12593#	12594#	12595#	12596#	12598#	12599#	12600#
12602#	12604#	12605#	12606#	12607#	12614#	12616#	12617#	12618#	12619#	12621#	12622#	12623#
12624#	12625#	12626#	12627#	12635#	12636#	12637#	12638#	12644#	12651#	12658#	12665#	12672#
12679#	12686#	12688#	12689#	12696#	12698#	12699#	12706#	12709#	12710#	12711#	12712#	12719#
12727#	12728#	12729#	12730#	12731#	12732#	12733#	12734#	12735#	12736#	12737#	12738#	12745#
12746#	12747#	12748#	12754#	12766#	12778#	12808#	12809#	12810#	12811#	12812#	12813#	12814#
12815#	12819#	12820#	12821#	12822#	12828#	12829#	12830#	12831#	12832#	12833#	12834#	12835#
12843#	12844#	12845#	12846#	12851#	12852#	12853#	12854#	12855#	12856#	12865#	12866#	12870#
12871#	12872#	12874#	12875#	12884#	12885#	12886#	12894#	12895#	12900#	12903#	12904#	12905#
12906#	12907#	12909#	12917#	12918#	12919#	12920#	12928#	12930#	12931#	12933#	12934#	12935#
12936#	12937#	12944#	12945#	12946#	12947#	12949#	12950#	12951#	12953#	12958#	12959#	12960#
12968#	12969#	12975#	12976#	12977#	12979#	12980#	12981#	12982#	12983#	12984#	12991#	12992#
12993#	12995#	13003#	13005#	13009#	13010#	13011#	13016#	13017#	13018#	13019#	13024#	13027#
13028#	13029#	13030#	13031#	13033#	13034#	13036#	13038#	13043#	13046#	13047#	13048#	13049#
13050#	13052#	13053#	13060#	13061#	13062#	13063#	13064#	13065#	13067#	13069#	13071#	13076#
13077#	13078#	13083#	13084#	13085#	13087#	13088#	13089#	13090#	13105#	13106#	13114#	13121#
13122#	13123#	13125#	13126#	13127#	13128#	13134#	13135#	13136#	13138#	13139#	13140#	13141#
13147#	13148#	13152#	13154#	13158#	13159#	13160#	13161#	13163#	13164#	13165#	13167#	13169#
13170#	13174#	13176#	13185#	13186#	13187#	13188#	13190#	13191#	13192#	13201#	13202#	13203#
13205#	13207#	13212#	13213#	13217#	13219#	13220#	13221#	13222#	13227#	13229#	13230#	13231#
13232#	13234#	13247#	13248#	13249#	13251#	13253#	13257#	13258#	13259#	13260#	13262#	13263#
13264#	13266#	13271#	13273#	13282#	13283#	13284#	13285#	13287#	13288#	13289#	13298#	13299#
13300#	13302#	13303#	13304#	13305#	13306#	13307#	13308#	13310#	13321#	13322#	13323#	13324#
13326#	13327#	13328#	13330#	13343#	13344#	13345#	13346#	13348#	13349#	13350#	13359#	13360#
13361#	13363#	13364#	13365#	13366#	13367#	13368#	13369#	13371#	13379#	13381#	13385#	13386#
13387#	13388#	13390#	13391#	13392#	13397#	13399#	13406#	13407#	13408#	13409#	13411#	13412#
13413#	13422#	13423#	13424#	13429#	13430#	13431#	13433#	13438#	13440#	13441#	13442#	13443#
13445#	13446#	13447#	13454#	13455#	13456#	13457#	13459#	13460#	13461#	13469#	13470#	13471#
13472#	13474#	13475#	13476#	13485#	13486#	13487#	13492#	13493#	13494#	13496#	13503#	13505#
13506#	13507#	13508#	13510#	13511#	13512#	13513#	13514#	13515#	13516#	13524#	13525#	13526#
13527#	13533#	13545#	13547#	13549#	13555#	13562#	13565#	13566#	13567#	13568#	13569#	13570#
13571#	13572#	13582#	13583#	13584#	13586#	13587#	13588#	13589#	13596#	13608#	13611#	13612#
13613#	13614#	13615#	13617#	13618#	13623#	13624#	13625#	13626#	13627#	13635#		
13648#	13649#	13650#	13651#	13652#	13653#	13654#	13655#	13656#	13657#	13658#		13660#
13661#	13662#	13663#	13664#	13665#	13666#	13667#	13668#	13669#	13678#	13679#	13680#	13681#
13687#	13699#	13711#	13714#	13715#	13721#	13728#	13730#	13731#	13743#	13744#	13745#	13747#
13748#	13797#	13799#	13808#	13810#	13812#	13813#	13814#	13815#	13816#	13817#	13818#	13819#
13831#	13832#	13833#	13834#	13835#	13836#	13846#	13849#	13852#	13853#	13865#	13866#	13867#
13868#	13869#	13870#	13871#	13872#	13873#	13874#	13875#	13876#	13877#	13878#	13879#	13880#
13886#	13887#	13923#	13925#	13926#	13927#	13928#	13930#	13931#	13932#	13933#	13935#	13936#
13937#	13944#	13945#	13946#	13947#	13955#	13956#	13957#	13958#	13964#	13971#	13978#	13985#
13992#	13999#	14006#	14013#	14014#	14015#	14016#	14017#	14018#	14030#	14032#	14033#	14035#

14036#	14037#	14038#	14045#	14059#	14060#	14061#	14062#	14063#	14064#	14065#	14066#	14067#
14068#	14069#	14070#	14071#	14072#	14078#	14079#	14123#	14124#	14125#	14126#	14132#	14149#
14161#	14168#	14169#	14170#	14171#	14172#	14174#	14177#	14178#	14179#	14180#	14181#	14182#
14194#	14196#	14197#	14212#	14213#	14214#	14215#	14216#	14217#	14227#	14230#	14233#	14234#
14246#	14247#	14248#	14249#	14250#	14251#	14252#	14253#	14254#	14255#	14256#	14257#	14258#
14259#	14260#	14261#	14262#	14263#	14264#	14265#	14271#	14272#	14309#	14311#	14312#	14313#
14314#	14316#	14317#	14318#	14319#	14321#	14322#	14323#	14330#	14331#	14332#	14333#	14341#
14342#	14343#	14344#	14350#	14357#	14364#	14371#	14378#	14385#	14392#	14400#	14401#	14402#
14403#	14404#	14405#	14417#	14419#	14420#	14422#	14423#	14424#	14425#	14432#	14446#	14447#
14448#	14449#	14450#	14451#	14452#	14453#	14454#	14455#	14456#	14457#	14458#	14459#	14460#
14461#	14462#	14463#	14469#	14470#	14511#	14512#	14513#	14514#	14520#	14537#	14549#	14556#
14557#	14558#	14559#	14560#	14562#	14565#	14566#	14567#	14568#	14569#	14570#	14582#	14584#
14585#	14600#	14601#	14602#	14603#	14604#	14605#	14611#	14612#	14613#	14614#	14615#	14616#
14622#	14623#	14624#	14625#	14626#	14627#	14628#	14629#	14630#	14631#	14638#	14640#	14641#
14642#	14643#	14644#	14649#	14652#	14653#	14654#	14655#	14656#	14658#	14659#	14660#	14662#
14663#	14664#	14665#	14666#	14671#	14674#	14675#	14676#	14677#	14678#	14680#	14681#	14693#
14694#	14695#	14696#	14703#	14704#	14705#	14706#	14713#	14714#	14715#	14717#	14719#	14732#
14733#	14734#	14736#	14737#	14739#	14740#	14746#	14747#	14748#	14749#	14756#	14757#	14758#
14760#	14770#	14771#	14772#	14773#	14774#	14775#	14782#	14783#	14784#	14785#	14795#	14796#
14797#	14798#	14805#	14806#	14807#	14809#	14811#	14812#	14814#	14815#	14816#	14818#	14820#
14822#	14835#	14836#	14837#	14839#	14840#	14841#	14848#	14849#	14850#	14851#	14853#	14854#
14855#	14857#	14858#	14863#	14864#	14865#	14866#	14867#	14873#	14874#	14877#	14878#	14879#
14880#	14881#	14882#	14883#	14884#	14886#	14887#	14893#	14894#	14895#	14896#	14898#	14899#
14900#	14902#	14903#	14904#	14907#	14908#	14909#	14911#	14912#	14913#	14920#	14921#	14922#
14923#	14925#	14926#	14927#	14929#	14930#	14935#	14936#	14937#	14938#	14939#	14945#	14946#
14949#	14950#	14951#	14952#	14953#	14954#	14955#	14956#	14958#	14959#	14965#	14966#	14967#
14968#	14970#	14971#	14972#	14974#	14975#	14976#	14983#	14985#	14986#	14987#	14988#	14990#
14991#	14992#	14993#	14994#	14995#	14996#	15004#	15005#	15006#	15007#	15013#	15025#	15037#
15039#	15040#	15041#	15042#	15049#	15058#	15059#	15060#	15061#	15062#	15063#	15064#	15065#
15066#	15067#	15068#	15069#	15076#	15077#	15078#	15079#	15085#	15097#	15109#	15139#	15140#
15141#	15142#	15143#	15144#	15145#	15146#	15152#	15153#	15159#	15160#	15161#	15162#	15163#
15164#	15165#	15166#	15174#	15175#	15176#	15177#	15184#	15185#	15186#	15187#	15189#	15190#
15191#	15193#	15194#	15199#	15200#	15201#	15202#	15203#	15209#	15210#	15211#	15212#	15213#
15214#	15219#	15220#	15221#	15222#	15224#	15225#	15226#	15228#	15237#	15238#	15242#	15243#
15244#	15246#	15247#	15256#	15257#	15258#	15263#	15264#	15267#	15268#	15275#	15276#	15277#
15278#	15280#	15281#	15282#	15293#	15294#	15295#	15307#	15309#	15310#	15311#	15314#	15315#
15316#	15317#	15318#	15319#	15326#	15327#	15328#	15329#	15331#	15332#	15333#	15334#	15335#
15336#	15339#	15340#	15341#	15342#	15343#	15345#	15350#	15351#	15352#	15353#	15354#	15355#
15358#	15359#	15360#	15361#	15362#	15373#	15374#	15380#	15381#	15382#	15384#	15385#	15386#
15387#	15389#	15390#	15391#	15393#	15394#	15395#	15396#	15397#	15398#	15405#	15406#	15407#
15409#	15425#	15427#	15431#	15432#	15433#	15438#	15439#	15440#	15441#	15442#	15443#	15445#
15447#	15450#	15451#	15454#	15455#	15456#	15457#	15458#	15459#	15460#	15461#	15463#	15473#
15474#	15475#	15476#	15477#	15478#	15480#	15481#	15482#	15484#	15485#	15486#	15487#	15488#
15489#	15490#	15491#	15493#	15495#	15497#	15506#	15508#	15509#	15510#	15511#	15513#	15514#
15515#	15516#	15517#	15518#	15519#	15523#	15524#	15525#	15526#	15532#	15542#	15552#	15554#
15556#	15562#	15569#	15573#	15574#	15575#	15576#	15577#	15578#	15579#	15580#	15590#	15591#
15592#	15594#	15595#	15596#	15597#	15604#	15606#	15607#	15612#	15613#	15614#	15615#	15616#
15624#	15635#	15636#	15637#	15638#	15639#	15640#	15651#	15652#	15653#	15654#	15660#	15672#
15684#	15687#	15688#	15694#	15701#	15703#	15704#	15714#	15715#	15718#	15720#	15721#	15769#
15771#	15780#	15782#	15784#	15785#	15786#	15787#	15788#	15789#	15790#	15791#	15804#	15805#
15806#	15807#	15808#	15809#	15819#	15822#	15825#	15826#	15838#	15839#	15840#	15841#	15842#
15843#	15844#	15845#	15846#	15847#	15848#	15849#	15850#	15851#	15852#	15853#	15859#	15860#
15896#	15898#	15899#	15900#	15901#	15903#	15904#	15905#	15906#	15908#	15909#	15910#	15917#
15918#	15919#	15920#	15928#	15929#	15930#	15931#	15937#	15944#	15951#	15958#	15965#	15972#
15979#	15987#	15988#	15989#	15990#	15991#	15992#	16004#	16006#	16007#	16009#	16010#	16011#
16012#	16019#	16033#	16034#	16035#	16036#	16037#	16038#	16039#	16040#	16041#	16042#	16043#

16044#	16045#	16046#	16052#	16053#	16095#	16096#	16097#	16098#	16104#	16121#	16133#	16140#	
16141#	16142#	16143#	16144#	16146#	16149#	16150#	16151#	16152#	16153#	16154#	16166#	16168#	
16169#	16202#	16204#	16213#	16215#	16217#	16218#	16219#	16220#	16221#	16222#	16223#	16224#	
16237#	16238#	16239#	16240#	16241#	16242#	16252#	16255#	16258#	16259#	16271#	16272#	16273#	
16274#	16275#	16276#	16277#	16278#	16279#	16280#	16281#	16282#	16283#	16284#	16285#	16286#	
16287#	16288#	16289#	16290#	16296#	16297#	16334#	16336#	16337#	16338#	16339#	16341#	16342#	
16343#	16344#	16346#	16347#	16348#	16355#	16356#	16357#	16358#	16366#	16367#	16368#	16369#	
16375#	16382#	16389#	16396#	16403#	16410#	16417#	16425#	16426#	16427#	16428#	16429#	16430#	
16442#	16444#	16445#	16447#	16448#	16449#	16450#	16457#	16471#	16472#	16473#	16474#	16475#	
16476#	16477#	16478#	16479#	16480#	16481#	16482#	16483#	16484#	16485#	16486#	16487#	16488#	
16494#	16495#	16537#	16538#	16539#	16540#	16546#	16563#	16575#	16582#	16583#	16584#	16585#	
16586#	16588#	16591#	16592#	16593#	16594#	16595#	16596#	16608#	16610#	16611#	16864#	16865#	
16866#	16867#	16870#	16979#	16980#	16981#	16982#	16983#	16984#	16985#	16986#	16987#	16988#	
16989#	16990#	16991#	17008#	17009#	17010#	17011#	17012#	17013#	17014#	17015#	17018#	17019#	
17020#	17021#	17022#	17023	17025#	17026#	17027#	17028#	17029#	17045#	17046#	17047#	17048#	
17049#	17050#	17051#	17052#	17055#	17056#	17057#	17058#	17059#	17060	17062#	17063#	17064#	
17065#	17066#	17082#	17083#	17084#	17085#	17086#	17087#	17088#	17089#	17092#	17093#	17094#	
17095#	17096#	17097	17099#	17100#	17101#	17102#	17103#	17117#	17118#	17119#	17120#	17121#	
17122#	17123#	17124#	17127#	17128#	17129#	17130#	17131#	17132	17134#	17135#	17136#	17137#	
17138#	17155#	17156#	17157#	17158#	17159#	17160#	17161#	17162#	17165#	17166#	17167#	17168#	
17169#	17170	17172#	17173#	17174#	17175#	17176#	17191#	17192#	17193#	17194#	17195#	17196#	
17197#	17198#	17199#	17200#	17201#	17202#	17203#	17218#	17219#	17220#	17221#	17222#	17223#	
17224#	17225#	17226#	17227#	17228#	17229#	17230#	17254#	17255#	17265#	17266#	17267#	17268#	
17273#	17274#	17275#	17277#	17287#	17288#	17289#	17290#	17291#	17292#	17293#	17294#	17295#	
17296#	17297#	17298#	17299#	17300#	17301#	17302#	17304#	17305#	17306#	17307#	17308#	17309#	
17310#	17311#	17312#	17313#	17314#	17315#	17316#	17317#	17318#	17319#	17320#	17330#	17331#	
17335#	17336#	17338#	17339#	17340#	17342#	17390#	17391#	17392#	17393#	17394#	17396#	17405#	
17406#	17407#	17408#	17410#	17411#	17412#	17425#	17426#	17427#	17428#	17433#	17434#	17435#	
17436#	17437#	17443#	17444#	17447#	17448#	17449#	17450#	17451#	17452#	17453#	17454#	17456#	
17457#	17460#	17461#	17462#	17463#	17464#	17466#							
\$TESTN 000636	1138#	17497*											
\$TIMES 000404	1040#	1298*	1518*	1713*	2330*	2834*	3361*	3645*	3929*	4242*	4549*	4899*	5251*
	5598*	5708*	6237*	6801*	7226*	7973*	8772*	10241*	11162*	12090*	12808*	13831*	14212*
	14600*	15139*	15804*	16237*	16866*	16884*	17492	17495*					
\$TPFLG 000435	1056#	17652											
\$TRAP 062154	1187	18003#											
\$TRAP2 062176	18014#	18025											
\$TRPAD 062210	18008	18025#											
\$TSKO = 052414	1269#	1277	1303#	1424	1450#	1453	1457#	1460	1464#	1467	1471#	1474	1478#
	1481	1485#	1488	1523#	1629	1653#	1656	1660#	1663	1667#	1670	1674#	1677
	1681#	1684	1688#	1691	1695#	1698	1762#	1766	1767#	1770	1783#	1787	1788#
	1792	1808#	1812	1813#	1826	1832#	1836	1837#	1850	1854#	1858	1859#	1862
	1882#	1885	1889#	1892	1896#	1899	1903#	1906	1910#	1913	1917#	1920	1924#
	1927	1942#	1964	1965#	2009	2015#	2049	2058#	2061	2100#	2107	2111#	2114
	2118#	2121	2125#	2128	2132#	2135	2139#	2142	2146#	2149	2153#	2156	2160#
	2163	2171#	2179	2378#	2382	2383#	2386	2399#	2403	2404#	2408	2424#	2428
	2429#	2442	2448#	2452	2453#	2466	2488#	2491	2495#	2498	2502#	2505	2509#
	2512	2516#	2519	2523#	2526	2530#	2533	2546#	2572	2573#	2618	2622#	2644
	2653#	2656	2695#	2702	2706#	2709	2713#	2716	2720#	2723	2727#	2730	2734#
	2737	2741#	2744	2748#	2751	2755#	2758	2765#	2773	2894#	2898	2899#	2903
	2919#	2923	2924#	2937	2943#	2947	2948#	2961	2972#	2974	2977#	2981	3009#
	3012	3016#	3019	3023#	3026	3030#	3033	3037#	3040	3044#	3047	3051#	3054
	3073#	3095	3096#	3141	3145#	3168	3177#	3180	3221#	3228	3232#	3235	3239#
	3242	3246#	3249	3253#	3256	3260#	3263	3267#	3270	3274#	3277	3281#	3284
	3292#	3300	3397#	3405	3406#	3413	3417#	3424	3425#	3432	3475#	3478	3482#
	3485	3489#	3492	3496#	3499	3503#	3506	3510#	3513	3517#	3520	3529#	3532

3562#	3565	3571#	3574	3578#	3581	3585#	3588	3592#	3595	3599#	3602	3606#
3609	3683#	3691	3692#	3699	3703#	3710	3711#	3718	3761#	3764	3768#	3771
3775#	3778	3782#	3785	3789#	3792	3796#	3799	3803#	3806	3815#	3818	3848#
3851	3858#	3861	3865#	3868	3872#	3875	3879#	3882	3886#	3889	3893#	3896
3967#	3973	3974#	3979	3983#	3990	3991#	3998	4024#	4029	4057#	4060	4064#
4067	4071#	4074	4078#	4081	4085#	4088	4092#	4095	4099#	4102	4111#	4114
4134#	4139	4160#	4163	4169#	4172	4176#	4179	4183#	4186	4190#	4193	4197#
4200	4204#	4207	4280#	4286	4287#	4292	4296#	4303	4304#	4311	4337#	4342
4367#	4370	4374#	4377	4381#	4384	4388#	4391	4395#	4398	4402#	4405	4409#
4412	4421#	4424	4444#	4449	4467#	4470	4476#	4479	4483#	4486	4490#	4493
4497#	4500	4504#	4507	4511#	4514	4591#	4595	4596#	4599	4616#	4623	4624#
4630	4657#	4663	4667#	4673	4679#	4682	4702#	4705	4709#	4712	4716#	4719
4723#	4726	4730#	4733	4737#	4740	4744#	4747	4756#	4759	4796#	4799	4803#
4806	4810#	4813	4817#	4820	4824#	4827	4831#	4834	4838#	4841	4935#	4941
4942#	4947	4964#	4979	5000#	5023	5049#	5052	5056#	5059	5063#	5066	5070#
5073	5077#	5080	5084#	5087	5091#	5094	5103#	5106	5125#	5128	5144#	5147
5151#	5154	5158#	5161	5165#	5168	5172#	5175	5179#	5182	5186#	5189	5193#
5196	5240#	5248	5282#	5386	5394#	5398	5399#	5541	5572#	5576	5577#	5580
5627#	5631	5680#	5688	5750#	5762	5766#	5899	5916#	5922	5923#	5928	5961#
5964	5968#	5971	5975#	5978	5982#	5985	5989#	5992	5996#	5999	6003#	6006
6021#	6033	6043#	6046	6069#	6072	6076#	6079	6083#	6086	6090#	6093	6097#
6100	6104#	6107	6111#	6114	6120#	6123	6126#	6134	6139#	6142	6147#	6155
6282#	6330	6336#	6354	6360#	6483	6500#	6506	6507#	6512	6545#	6548	6552#
6555	6559#	6562	6566#	6569	6573#	6576	6580#	6583	6587#	6590	6605#	6617
6627#	6630	6653#	6656	6660#	6663	6667#	6670	6674#	6677	6681#	6684	6688#
6691	6695#	6698	6704#	6707	6712#	6720	6832#	6838	6839#	6844	6868#	6876
6877#	6981	7015#	7018	7022#	7025	7029#	7032	7036#	7039	7043#	7046	7050#
7053	7057#	7060	7070#	7073	7095#	7098	7102#	7105	7109#	7112	7116#	7119
7123#	7126	7130#	7133	7137#	7140	7144#	7147	7151#	7154	7257#	7263	7264#
7269	7293#	7301	7302#	7318	7326#	7423	7428#	7430	7433#	7437	7444#	7457
7491#	7494	7498#	7501	7505#	7508	7512#	7515	7519#	7522	7526#	7529	7533#
7536	7546#	7549	7571#	7574	7583#	7586	7595#	7598	7602#	7605	7609#	7612
8008#	8014	8015#	8020	8068#	8072	8073#	8086	8097#	8101	8102#	8105	8152#
8182	8191#	8196	8209#	8212	8226#	8229	8239#	8242	8280#	8285	8297#	8300
8309#	8315	8321#	8344	8350#	8363	8369#	8408	8414#	8432	8438#	8454	8460#
8478	8510#	8513	8517#	8520	8524#	8527	8531#	8534	8538#	8541	8545#	8548
8552#	8555	8566#	8578	8594#	8597	8619#	8622	8631#	8634	8643#	8646	8650#
8653	8657#	8660	8664#	8667	8671#	8674	8832#	8834	8837#	8841	8857#	8877
8884#	8894	8901#	8920	8927#	8944	8951#	8964	8971#	8984	9018#	9022	9023#
9039	9043#	9046	9058#	9068	9081#	9084	9094#	9097	9108#	9112	9113#	9116
9144#	9153	9164#	9192	9239#	9242	9246#	9249	9253#	9256	9260#	9263	9267#
9270	9274#	9277	9281#	9284	9295#	9307	9317#	9320	9343#	9346	9355#	9358
9367#	9370	9374#	9377	9381#	9384	9388#	9391	9554#	9557	9566#	9605	9615#
9627	9637#	9648	9655#	9662	9672#	9683	9690#	9697	9707#	9718	9725#	9732
9742#	9753	9760#	9767	9777#	9788	9795#	9802	9812#	9821	9825#	9832	9843#
9850	9861#	9868	9879#	9886	9901#	9906	9957#	9968	9979#	9990	10001#	10012
10023#	10034	10045#	10056	10067#	10076	10087#	10096	10107#	10116	10127#	10136	10154#
10166	10212#	10220	10254#	10256	10259#	10263	10298#	10314	10315#	10318	10327#	10343
10344#	10347	10356#	10374	10375#	10378	10420#	10423	10443#	10558	10567#	10570	10590#
10704	10713#	10716	10736#	10869	10912#	10915	10919#	10922	10926#	10929	10933#	10936
10940#	10943	10947#	10950	10954#	10957	10971#	10983	11008#	11011	11088#	11091	11105#
11108	11117#	11120	11134#	11146	11175#	11177	11180#	11184	11223#	11239	11240#	11243
11252#	11268	11269#	11272	11281#	11299	11300#	11303	11345#	11348	11368#	11483	11492#
11495	11517#	11631	11640#	11643	11663#	11796	11839#	11842	11846#	11849	11853#	11856
11860#	11863	11867#	11870	11874#	11877	11881#	11884	11899#	11911	11936#	11939	12016#
12019	12033#	12036	12045#	12048	12062#	12074	12136#	12138	12141#	12145	12149#	12152

12163#	12165	12168#	12172	12186#	12190	12191#	12213	12219#	12316	12334#	12337	12350#
12370	12374#	12399	12403#	12427	12431#	12447	12451#	12471	12480#	12483	12496#	12516
12520#	12540	12544#	12563	12567#	12583	12587#	12607	12641#	12644	12648#	12651	12655#
12658	12662#	12665	12669#	12672	12676#	12679	12683#	12686	12693#	12696	12703#	12706
12716#	12719	12751#	12754	12763#	12766	12775#	12778	12848#	12855	12869#	12872	12883#
12886	12893#	12909	12916#	12920	12921#	13071	13075#	13078	13111#	13114	13120#	13128
13133#	13141	13146#	13207	13211#	13234	13238#	13310	13314#	13371	13375#	13433	13437#
13496	13530#	13533	13542#	13545	13552#	13555	13559#	13562	13574#	13588	13593#	13596
13606#	13608	13611#	13615	13622#	13627	13632#	13635	13684#	13687	13696#	13699	13708#
13711	13718#	13721	13725#	13728	13733#	13747	13802#	13810	13844#	13846	13849#	13853
13961#	13964	13968#	13971	13975#	13978	13982#	13985	13989#	13992	13996#	13999	14003#
14006	14020#	14032	14042#	14045	14129#	14132	14146#	14149	14158#	14161	14184#	14196
14225#	14227	14230#	14234	14347#	14350	14354#	14357	14361#	14364	14368#	14371	14375#
14378	14382#	14385	14389#	14392	14407#	14419	14429#	14432	14517#	14520	14534#	14537
14546#	14549	14572#	14584	14647#	14649	14652#	14656	14669#	14671	14674#	14678	14692#
14696	14697#	14719	14725#	14822	14844#	14858	14859#	14904	14916#	14930	14931#	14976
15010#	15013	15022#	15025	15034#	15037	15046#	15049	15082#	15085	15094#	15097	15106#
15109	15180#	15194	15195#	15228	15241#	15244	15255#	15258	15274#	15278	15279#	15282
15292#	15295	15303#	15409	15417#	15497	15529#	15532	15539#	15542	15549#	15552	15559#
15562	15566#	15569	15582#	15596	15601#	15604	15611#	15616	15621#	15624	15657#	15660
15669#	15672	15681#	15684	15691#	15694	15698#	15701	15706#	15720	15774#	15782	15817#
15819	15822#	15826	15934#	15937	15941#	15944	15948#	15951	15955#	15958	15962#	15965
15969#	15972	15976#	15979	15994#	16006	16016#	16019	16101#	16104	16118#	16121	16130#
16133	16156#	16168	16207#	16215	16250#	16252	16255#	16259	16372#	16375	16379#	16382
16386#	16389	16393#	16396	16400#	16403	16407#	16410	16414#	16417	16432#	16444	16454#
16457	16543#	1656	16560#	16563	16572#	16575	16598#	16610	16863#	16870	17017#	17022
17054#	17059	17091#	17096	17126#	17131	17164#	17169	17264#	17268	17269#	17277	17286#
17320	17329#	17336	17337#	17340	17404#	17408	17409#	17412	17424#	17428	17429#	17464
1722#	1275	1314#	1322	1323#	1330	1345#	1354	1355#	1363	1367#	1370	1374#
1377	1381#	1384	1388#	1391	1395#	1398	1402#	1411	1412#	1420	1534#	1542
1543#	1550	1565#	1574	1575#	1583	1587#	1590	1594#	1597	1601#	1604	1608#
1611	1615#	1618	1622#	1625	1816#	1820	1821#	1824	1840#	1844	1845#	1848
1946#	1960	1969#	1983	1993#	2003	2018#	2022	2023#	2026	2035#	2043	2102#
2105	2174#	2177	2432#	2436	2437#	2440	2456#	2460	2461#	2464	2554#	2568
2575#	2587	2603#	2615	2627#	2638	2697#	2700	2768#	2771	2927#	2931	2932#
2935	2951#	2955	2956#	2959	2977#	2980	3077#	3091	3100#	3115	3125#	3135
3154#	3162	3223#	3226	3295#	3298	4026#	4029	4136#	4139	4339#	4342	4446#
4449	4968#	4974	5006#	5013	5014#	5021	5243#	5246	5337#	5339	5342#	5346
5358#	5361	5410#	5413	5417#	5420	5424#	5427	5431#	5434	5438#	5441	5445#
5448	5452#	5455	5475#	5489	5499#	5509	5517#	5539	5683#	5686	5757#	5760
5769#	5782	5783#	5897	6025#	6033	6129#	6132	6150#	6153	6287#	6290	6294#
6309	6313#	6328	6342#	6349	6369#	6384	6388#	6394	6397#	6403	6410#	6417
6424#	6451	6458#	6481	6609#	6617	6715#	6718	6887#	6894	6897#	6921	6944#
6979	7310#	7315	7332#	7347	7354#	7396	7400#	7404	7411#	7421	7433#	7436
7452#	7455	8076#	8080	8081#	8084	8154#	8182	8193#	8196	8282#	8285	8326#
8342	8353#	8359	8374#	8377	8381#	8406	8417#	8430	8444#	8448	8449#	8452
8468#	8472	8473#	8476	8570#	8578	8837#	8840	8860#	8875	8887#	8892	8904#
8918	8930#	8942	8954#	8958	8959#	8962	8974#	8978	8979#	8982	9029#	9033
9034#	9037	9063#	9066	9166#	9192	9299#	9307	9569#	9574	9575#	9603	9622#
9625	9639#	9648	9674#	9683	9709#	9718	9744#	9753	9779#	9788	9814#	9821
9827#	9832	9845#	9850	9863#	9868	9881#	9886	9903#	9906	9959#	9968	9981#
9990	10003#	10012	10025#	10034	10047#	10056	10069#	10076	10089#	10096	10109#	10116
10129#	10136	10161#	10164	10215#	10218	10259#	10262	10448#	10534	10535#	10558	10595#
10680	10681#	10704	10741#	10845	10846#	10869	10975#	10983	11138#	11146	11180#	11183
11373#	11459	11460#	11483	11522#	11607	11608#	11631	11668#	11772	11773#	11796	11903#
11911	12066#	12074	12141#	12144	12168#	12171	12196#	12200	12201#	12211	12222#	12231

STSK1 - 052416

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54MACY11 30A(1052) 14-JUL-81 14:19 PAGE 475
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0474

	12232#	12314	12356#	12360	12361#	12364	12377#	12397	12406#	12425	12437#	12441	12442#
	12445	12456#	12460	12461#	12464	12502#	12506	12507#	12510	12526#	12530	12531#	12534
	12549#	12553	12554#	12557	12573#	12577	12578#	12581	12592#	12596	12597#	12600	12850#
	12855	12898#	12900	12903#	12907	12927#	12995	13002#	13069	13151#	13167	13173#	13205
	13216#	13222	13226#	13232	13241#	13266	13270#	13308	13317#	13330	13334#	13369	13384#
	13388	13389#	13392	13405#	13409	13410#	13413	13421#	13424	13428#	13431	13453#	13457
	13458#	13461	13468#	13472	13473#	13476	13484#	13487	13491#	13494	13578#	13588	13611#
	13614	13737#	13747	13805#	13808	13849#	13852	14024#	14032	14188#	14196	14230#	14233
	14411#	14419	14576#	14584	14652#	14655	14674#	14677	14702#	14706	14707#	14717	14728#
	14737	14738#	14820	14847#	14851	14852#	14855	14862#	14867	14870#	14883	14892#	14896
	14897#	14900	14919#	14923	14924#	14927	14934#	14939	14942#	14955	14964#	14968	14969#
	14972	15183#	15187	15188#	15191	15198#	15203	15206#	15213	15218#	15222	15223#	15226
	15322#	15345	15349#	15362	15365#	15397	15404#	15407	15424#	15495	15586#	15596	15710#
	15720	15777#	15780	15822#	15825	15998#	16006	16160#	16168	16210#	16213	16255#	16258
	16436#	16444	16602#	16610	17272#	17275	17432#	17437	17440#	17453			
\$TSK10= 051516	10461#	10465	10466#	10469	10484#	10487	10502#	10505	10522#	10526	10527#	10530	10546#
	10550	10551#	10554	10608#	10612	10613#	10616	10631#	10634	10649#	10652	10668#	10672
	10673#	10676	10692#	10696	10697#	10700	10754#	10769	10780#	10794	10805#	10819	10833#
	10837	10838#	10841	10857#	10861	10862#	10865	11386#	11390	11391#	11394	11409#	11412
	11427#	11430	11447#	11451	11452#	11455	11471#	11475	11476#	11479	11535#	11539	11540#
	11543	11558#	11561	11576#	11579	11595#	11599	11600#	11603	11619#	11623	11624#	11627
	11581#	11696	11707#	11721	11732#	11746	11760#	11764	11765#	11768	11784#	11788	11789#
	11792												
\$TSK11= 051512	10757#	10761	10762#	10765	10787#	10790	10812#	10815	11684#	11688	11689#	11692	11714#
	11717	11739#	11742										
\$TSK2 = 052417	1348#	1351	1358#	1361	1405#	1408	1415#	1418	1568#	1571	1578#	1581	1950#
	1960	1973#	1983	1996#	1999	2038#	2041	2558#	2568	2578#	2581	2607#	2615
	2630#	2633	3081#	3091	3104#	3115	3128#	3131	3157#	3160	5342#	5345	5360#
	5361	5479#	5489	5502#	5505	5525#	5533	5786#	5793	5797#	5808	5812#	5829
	5834#	5864	5872#	5877	5878#	5895	6028#	6031	6297#	6301	6302#	6307	6316#
	6320	6321#	6326	6344#	6349	6376#	6381	6427#	6431	6432#	6435	6442#	6447
	6461#	6465	6469#	6475	6476#	6479	6612#	6615	6899#	6921	6953#	6977	7340#
	7345	7357#	7361	7362#	7367	7374#	7379	7386#	7391	7414#	7419	8167#	8180
	8332#	8336	8337#	8340	8386#	8389	8399#	8402	8423#	8426	8573#	8576	8865#
	8869	8870#	8873	8911#	8914	8935#	8938	9179#	9190	9302#	9305	10448#	10512
	10513#	10558	10595#	10658	10659#	10704	10741#	10823	10824#	10869	10978#	10981	11141#
	11144	11373#	11437	11438#	11483	11522#	11585	11586#	11631	11668#	11750	11751#	11796
	11906#	11909	12069#	12072	12206#	12209	12225#	12228	12239#	12243	12244#	12254	12258#
	12312	12383#	12387	12388#	12391	12411#	12415	12416#	12419	12903#	12906	12940#	12953
	12957#	12960	12963#	12983	12990#	12993	13008#	13011	13015#	13034	13035#	13067	13157#
	13161	13162#	13165	13184#	13188	13189#	13192	13200#	13203	13246#	13249	13256#	13260
	13261#	13264	13281#	13285	13286#	13289	13297#	13300	13320#	13324	13325#	13328	13342#
	13346	13347#	13350	13358#	13361	13581#	13584	13740#	13745	14027#	14030	14191#	14194
	14414#	14417	14579#	14582	14712#	14715	14731#	14734	14745#	14749	14750#	14760	14764#
	14818	14872#	14883	14944#	14955	15208#	15213	15325#	15329	15330#	15343	15369#	15397
	15430#	15433	15437#	15443	15444#	15493	15589#	15592	15713#	15718	16001#	16004	16163#
	16166	16439#	16442	16605#	16608	17442#	17453						
\$TSK3 = 052236	1953#	1956	1976#	1979	2561#	2564	2610#	2613	3084#	3087	3107#	3111	5482#
	5485	5528#	5531	5837#	5842	5847#	5861	5881#	5886	5887#	5893	6378#	6381
	6962#	6965	6972#	6975	10448#	10493	10494#	10558	10595#	10640	10641#	10704	10741#
	10798	10799#	10869	11373#	11418	11419#	11483	11522#	11567	11568#	11631	11668#	11725
	11726#	11796	12249#	12252	12271#	12278	12283#	12306	12307#	12310	12943#	12947	12948#
	12951	12967#	12983	13022#	13024	13027#	13031	13041#	13043	13046#	13050	13055#	13064
	14755#	14758	14777#	14784	14789#	14812	14813#	14816	15372#	15387	15388#	15391	15465#
	15490												
\$TSK4 052237	5849#	5861	10448#	10475	10476#	10558	10595#	10622	10623#	10704	10741#	10773	10774#

\$TSK5 = 052241	10869	11373#	11400	11401#	11483	11522#	11549	11550#	11631	11668#	11700	11701#	11796
	12275#	12278	12288#	12292	12293#	12303	12974#	12977	13027#	13030	13046#	13049	13059#
	13064	14781#	14784	14794#	14798	14799#	14809	15379#	15382	15469#	15490	10869	11373#
	10448#	10455	10456#	10558	10595#	10602	10603#	10704	10741#	10748	10749#	12298#	12301
	11380	11381#	11483	11522#	11529	11530#	11631	11668#	11675	11676#	11796	10869	11373#
	14804#	14807	15472#	15478	15479#	15482						10869	11373#
\$TSK6 = 177777	10448#	10556	10557#	10558	10595#	10702	10703#	10704	10741#	10867	10868#	10869	11373#
	11481	11482#	11483	11522#	11629	11630#	11631	11668#	11794	11795#	11796		
\$TSK7 = 051503	10448#	10474	10492	10511	10533	10556	10595#	10621	10639	10657	10679	10702	10741#
	10772	10797	10822	10844	10867	11373#	11399	11417	11436	11458	11481	11522#	11548
	11566	11584	11606	11629	11668#	11699	11724	11749	11771	11794			
\$TSTM 000304	1026#												
\$TSTM 000402	1039#	16864*	16883*	17496*	17497	17500	17566						
\$TTYIN 061664	17907	17908	17928	17932#									
\$TYPE 060402	17652#	17969	18026										
\$TYPEC 060606	17679	17686	17693	17698#	17699	17850							
\$TYPEX 060654	17704	17706	17709#										
\$TYPOC 060702	17741#	18027											
\$TYPON 060716	17740	17743#	18029										
\$TYPOS 060656	17736#	18028											
\$U = 000403	2970#	2973	5335#	5338	7426#	7429	8830#	8833	10252#	10255	11173#	11176	12134#
	12137	12161#	12164	12896#	12899	13020#	13023	13039#	13042	13604#	13607	13842#	13845
	14223#	14226	14645#	14648	14667#	14670	15815#	15818	16248#	16251			
\$UNIT 000644	1141#												
\$UNITM 000310	1028#												
\$USWR 000656	1148#												
\$SARGC= 000006	720#	9534#	9936#	16978#	17007#	17044#	17081#	17116#	17154#	17190#	17217#	17246#	17389#
\$SBYTE= 000403	720#	1267#	1270#	1312#	1343#	1346#	1356#	13 5#	1372#	1379#	1386#	1393#	1400#
	1403#	1413#	1448#	1455#	1462#	1469#	1476#	1483#	1532#	1563#	1566#	1576#	1585#
	1592#	1599#	1606#	1613#	1620#	1651#	1658#	1665#	1672#	1679#	1686#	1693#	1760#
	1778#	1780#	1806#	1814#	1830#	1838#	1852#	1880#	1887#	1894#	1901#	1908#	1915#
	1922#	1940#	1946#	1948#	1951#	1969#	1971#	1974#	1994#	2013#	2016#	2036#	2056#
	2109#	2116#	2123#	2130#	2137#	2144#	2151#	2158#	2172#	2376#	2394#	2396#	2422#
	2430#	2446#	2454#	2486#	2493#	2500#	2507#	2514#	2521#	2528#	2544#	2554#	2556#
	2559#	2576#	2603#	2605#	2608#	2620#	2628#	2651#	2704#	2711#	2718#	2725#	2732#
	2739#	2746#	2753#	2766#	2889#	2891#	2917#	2925#	2949#	2949#	3007#	3014#	3021#
	3028#	3035#	3042#	3049#	3071#	3077#	3079#	3082#	3100#	3102#	3105#	3126#	3143#
	3155#	3175#	3230#	3237#	3244#	3251#	3258#	3265#	3272#	3279#	3293#	3415#	3473#
	3480#	3487#	3494#	3501#	3508#	3515#	3527#	3569#	3576#	3583#	3590#	3597#	3604#
	3701#	3759#	3766#	3773#	3780#	3787#	3794#	3801#	3813#	3856#	3863#	3870#	3877#
	3884#	3891#	3981#	4024#	4055#	4062#	4069#	4076#	4083#	4090#	4097#	4109#	4134#
	4167#	4174#	4181#	4188#	4195#	4202#	4294#	4337#	4365#	4372#	4379#	4386#	4393#
	4400#	4407#	4419#	4444#	4474#	4 81#	4488#	4495#	4502#	4509#	4614#	4655#	4665#
	4675#	4677#	4700#	4707#	47 14#	4721#	4728#	4735#	4742#	4754#	4794#	4801#	4808#
	4815#	4822#	4829#	4836#	4962#	4998#	5004#	5047#	5054#	5061#	5068#	5075#	5082#
	5089#	5101#	5149#	5156#	5163#	5170#	5177#	5184#	5191#	5238#	5241#	5358#	5392#
	5408#	5415#	5422#	5429#	5436#	5443#	5450#	5475#	5477#	5480#	5500#	5515#	5526#
	5570#	5625#	5678#	5681#	5748#	5755#	5764#	5767#	5784#	5795#	5810#	5832#	5835#
	5847#	5870#	5879#	5914#	5959#	5966#	5973#	5980#	5987#	5994#	6001#	6021#	6023#
	6026#	6041#	6067#	6074#	6081#	6088#	6095#	6102#	6109#	6118#	6127#	6137#	6148#
	6280#	6285#	6292#	6295#	6311#	6314#	6332#	6334#	6342#	6356#	6358#	6367#	6376#
	6386#	6406#	6408#	6419#	6421#	6425#	6440#	6453#	6455#	6459#	6467#	6498#	6543#
	6550#	6557#	6564#	6571#	6578#	6585#	6605#	6607#	6610#	6625#	6651#	6658#	6665#
	6672#	6679#	6686#	6693#	6702#	6713#	6830#	6866#	6885#	6897#	6942#	6948#	6950#
	6960#	6967#	6969#	7013#	7020#	7027#	7034#	7041#	7048#	7055#	7068#	7093#	7100#
	7107#	7114#	7121#	7128#	7 35#	7142#	7149#	7255#	7291#	7324#	7327#	7329#	7338#

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 477
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0476

7349#	7351#	7355#	7384#	7398#	7406#	7408#	7412#	7442#	7447#	7449#	7489#	7496#	
7503#	7510#	7517#	7524#	7531#	7544#	7569#	7576#	7578#	7580#	7588#	7590#	7592#	
7600#	7607#	8006#	8063#	8065#	8074#	8095#	8152#	8165#	8191#	8205#	8207#	8221#	
8223#	8237#	8280#	8295#	8307#	8319#	8324#	8330#	8348#	8351#	8367#	8370#	8372#	
8379#	8382#	8384#	8397#	8412#	8415#	8421#	8436#	8442#	8458#	8466#	8508#	8515#	
8522#	8529#	8536#	8543#	8550#	8566#	8568#	8571#	8592#	8617#	8624#	8626#	8628#	
8636#	8638#	8640#	8648#	8655#	8662#	8669#	8855#	8858#	8863#	8882#	8885#	8899#	
8902#	8909#	8925#	8928#	8933#	8949#	8952#	8969#	8972#	9016#	9024#	9026#	9041#	
9054#	9056#	9059#	9061#	9076#	9078#	9092#	9106#	9142#	9164#	9177#	9237#	9244#	
9251#	9258#	9265#	9272#	9279#	9295#	9297#	9300#	9315#	9341#	9348#	9350#	9352#	
9360#	9362#	9364#	9372#	9379#	9386#	9562#	9564#	9567#	9613#	9620#	9637#	9651#	
9653#	9672#	9686#	9688#	9707#	9721#	9723#	9742#	9756#	9758#	9777#	9791#	9793#	
9812#	9825#	9843#	9861#	9879#	9901#	9957#	9979#	10001#	10023#	10045#	10067#	10087#	
10107#	10127#	10152#	10159#	10210#	10213#	10296#	10325#	10354#	10418#	10459#	10482#	10500#	
10520#	10544#	10565#	10606#	10629#	10647#	10666#	10690#	10711#	10752#	10755#	10778#	10785#	
10803#	10810#	10831#	10855#	10910#	10917#	10924#	10931#	10938#	10945#	10952#	10971#	10973#	
10976#	11006#	11086#	11098#	11100#	11102#	11110#	11112#	11114#	11134#	11136#	11139#	11221#	
11250#	11279#	11343#	11384#	11407#	11425#	11445#	11469#	11490#	11533#	11556#	11574#	11593#	
11617#	11638#	11679#	11682#	11705#	11712#	11730#	11737#	11758#	11782#	11837#	11844#	11851#	
11858#	11865#	11872#	11879#	11899#	11901#	11904#	11934#	12014#	12026#	12028#	12030#	12038#	
12040#	12042#	12062#	12064#	12067#	12147#	12182#	12184#	12192#	12194#	12202#	12204#	12215#	
12217#	12220#	12223#	12235#	12237#	12245#	12247#	12256#	12271#	12273#	12281#	12284#	12286#	
12294#	12296#	12332#	12348#	12354#	12372#	12375#	12381#	12401#	12404#	12409#	12429#	12435#	
12449#	12454#	12478#	12494#	12500#	12518#	12524#	12542#	12547#	12565#	12571#	12585#	12590#	
12639#	12646#	12653#	12660#	12667#	12674#	12681#	12691#	12701#	12714#	12749#	12756#	12758#	
12760#	12768#	12770#	12772#	12848#	12867#	12879#	12881#	12891#	12911#	12913#	12925#	12938#	
12941#	12955#	12963#	12965#	12970#	12972#	12986#	12988#	13000#	13006#	13013#	13055#	13057#	
13073#	13109#	13118#	13131#	13144#	13149#	13155#	13171#	13177#	13179#	13181#	13194#	13196#	
13198#	13209#	13214#	13224#	13236#	13239#	13242#	13244#	13254#	13268#	13274#	13276#	13278#	
13291#	13293#	13295#	13312#	13315#	13318#	13332#	13335#	13337#	13339#	13352#	13354#	13356#	
13373#	13382#	13400#	13402#	13415#	13417#	13419#	13426#	13435#	13451#	13463#	13465#	13478#	
13480#	13482#	13489#	13528#	13535#	13537#	13539#	13550#	13557#	13574#	13576#	13579#	13591#	
13620#	13630#	13682#	13689#	13691#	13693#	13701#	13703#	13705#	13716#	13723#	13733#	13735#	
13738#	13800#	13803#	13959#	13966#	13973#	13980#	13987#	13994#	14001#	14020#	14022#	14025#	
14040#	14127#	14139#	14141#	14143#	14151#	14153#	14155#	14184#	14186#	14189#	14345#	14352#	
14359#	14366#	14373#	14380#	14387#	14407#	14409#	14412#	14427#	14515#	14527#	14529#	14531#	
14539#	14541#	14543#	14572#	14574#	14577#	14688#	14690#	14698#	14700#	14708#	14710#	14721#	
14723#	14726#	14729#	14741#	14743#	14751#	14753#	14762#	14777#	14779#	14787#	14790#	14792#	
14800#	14802#	14842#	14845#	14860#	14870#	14890#	14914#	14917#	14932#	14942#	14962#	15008#	
15015#	15017#	15019#	15027#	15029#	15031#	15044#	15080#	15087#	15089#	15091#	15099#	15101#	
15103#	15178#	15181#	15196#	15206#	15216#	15239#	15251#	15253#	15269#	15271#	15287#	15289#	
15297#	15299#	15301#	15320#	15323#	15347#	15365#	15367#	15370#	15375#	15377#	15400#	15402#	
15411#	15413#	15415#	15422#	15428#	15435#	15465#	15467#	15470#	15527#	15534#	15536#	15544#	
15546#	15557#	15564#	15582#	15584#	15587#	15599#	15609#	15619#	15655#	15662#	15664#	15666#	
15674#	15676#	15678#	15689#	15696#	15706#	15708#	15711#	15772#	15775#	15932#	15939#	15946#	
15953#	15960#	15967#	15974#	15994#	15996#	15999#	16014#	16099#	16111#	16113#	16115#	16123#	
16125#	16127#	16156#	16158#	16161#	16205#	16208#	16370#	16377#	16384#	16391#	16398#	16405#	
16412#	16432#	16434#	16437#	16452#	16541#	16553#	16555#	16557#	16565#	16567#	16569#	16598#	
16600#	16603#	16859#	16861#	17262#	17270#	17284#	17327#	17402#	17422#	17430#	17440#		
	720#	10454#	10455#	10474#	10475#	10492#	10493#	10511#	10512#	10533#	10534#	10601#	10602#
	10621#	10622#	10639#	10640#	10657#	10658#	10679#	10680#	10747#	10748#	10772#	10773#	10797#
	10798#	10822#	10823#	10844#	10845#	11379#	11380#	11399#	11400#	11417#	11418#	11436#	11437#
	11458#	11459#	11528#	11529#	11548#	11549#	11566#	11567#	11584#	11585#	11606#	11607#	11674#
	11675#	11699#	11700#	11724#	11725#	11749#	11750#	11771#	11772#				
	720#												
	720#	1267#	1270#	1275#	1277#	1312#	1321	1330#	1343#	1346#	1351#	1353	1356#

SSCASE- 000404

SSDST 000000
SSLOC- 000402

1361#	1363#	1365#	1370#	1372#	1377#	1379#	1384#	1386#	1391#	1393#	1398#	1400#
1403#	1408#	1410	1413#	1418#	1420#	1448#	1453#	1455#	1460#	1462#	1467#	1469#
1474#	1476#	1481#	1483#	1488#	1532#	1541	1550#	1563#	1566#	1571#	1573	1576#
1581#	1583#	1585#	1590#	1592#	1597#	1599#	1604#	1606#	1611#	1613#	1618#	1620#
1625#	1651#	1656#	1658#	1663#	1665#	1670#	1672#	1677#	1679#	1684#	1686#	1691#
1693#	1698#	1760#	1765	1770#	1778#	1786	1792#	1806#	1811	1814#	1819	1824#
1826#	1830#	1835	1838#	1843	1848#	1850#	1852#	1857	1862#	1880#	1885#	1887#
1892#	1894#	1899#	1901#	1906#	1908#	1913#	1915#	1920#	1922#	1927#	1940#	1951#
1956#	1963	1974#	1979#	1994#	1999#	2009#	2013#	2016#	2021	2026#	2036#	2041#
2049#	2056#	2061#	2099#	2101#	2105#	2107#	2109#	2114#	2116#	2121#	2123#	2128#
2130#	2135#	2137#	2142#	2144#	2149#	2151#	2156#	2158#	2163#	2172#	2177#	2376#
2381	2386#	2394#	2402	2408#	2422#	2427	2430#	2435	2440#	2442#	2446#	2451
2454#	2459	2464#	2466#	2486#	2491#	2493#	2498#	2500#	2505#	2507#	2512#	2514#
2519#	2521#	2526#	2528#	2533#	2544#	2559#	2564#	2571	2576#	2581#	2608#	2613#
2618#	2620#	2628#	2633#	2644#	2651#	2656#	2694#	2696#	2700#	2702#	2704#	2709#
2711#	2716#	2718#	2723#	2725#	2730#	2732#	2737#	2739#	2744#	2746#	2751#	2753#
2758#	2766#	2771#	2889#	2897	2903#	2917#	2922	2925#	2930	2935#	2937#	2941#
2946	2949#	2954	2959#	2961#	3007#	3012#	3014#	3019#	3021#	3026#	3028#	3033#
3035#	3040#	3042#	3047#	3049#	3054#	3071#	3082#	3087#	3094	3105#	3111#	3126#
3131#	3141#	3143#	3155#	3160#	3168#	3175#	3180#	3220#	3222#	3226#	3228#	3230#
3235#	3237#	3242#	3244#	3249#	3251#	3256#	3258#	3263#	3265#	3270#	3272#	3277#
3279#	3284#	3293#	3298#	3396#	3404	3413#	3415#	3423	3432#	3473#	3478#	3480#
3485#	3487#	3492#	3494#	3499#	3501#	3506#	3508#	3513#	3515#	3520#	3527#	3532#
3561#	3565#	3569#	3574#	3576#	3581#	3583#	3588#	3590#	3595#	3597#	3602#	3604#
3609#	3682#	3690	3699#	3701#	3709	3718#	3759#	3764#	3766#	3771#	3773#	3778#
3780#	3785#	3787#	3792#	3794#	3799#	3801#	3806#	3813#	3818#	3847#	3851#	3856#
3861#	3863#	3868#	3870#	3875#	3877#	3882#	3884#	3889#	3891#	3896#	3966#	3972
3979#	3981#	3989	3998#	4055#	4060#	4062#	4067#	4069#	4074#	4076#	4081#	4083#
4088#	4090#	4095#	4097#	4102#	4109#	4114#	4159#	4163#	4167#	4172#	4174#	4179#
4181#	4186#	4188#	4193#	4195#	4200#	4202#	4207#	4279#	4285	4292#	4294#	4302
4311#	4365#	4370#	4372#	4377#	4379#	4384#	4386#	4391#	4393#	4398#	4400#	4405#
4407#	4412#	4419#	4424#	4466#	4470#	4474#	4479#	4481#	4486#	4488#	4493#	4495#
4500#	4502#	4507#	4509#	4514#	4590#	4594	4599#	4614#	4622	4630#	4655#	4663#
4665#	4673#	4675#	4682#	4700#	4705#	4707#	4712#	4714#	4719#	4721#	4726#	4728#
4733#	4735#	4740#	4742#	4747#	4754#	4759#	4794#	4799#	4801#	4806#	4808#	4813#
4815#	4820#	4822#	4827#	4829#	4834#	4836#	4841#	4934#	4940	4947#	4962#	4979#
4998#	5004#	5012	5021#	5023#	5047#	5052#	5054#	5059#	5061#	5066#	5068#	5073#
5075#	5080#	5082#	5087#	5089#	5094#	5101#	5106#	5143#	5147#	5149#	5154#	5156#
5161#	5163#	5168#	5170#	5175#	5177#	5182#	5184#	5189#	5191#	5196#	5238#	5241#
5246#	5248#	5392#	5397	5408#	5413#	5415#	5420#	5422#	5427#	5429#	5434#	5436#
5441#	5443#	5448#	5450#	5455#	5480#	5485#	5500#	5505#	5515#	5526#	5531#	5539#
5541#	5570#	5575	5580#	5625#	5631#	5678#	5681#	5686#	5688#	5748#	5755#	5760#
5762#	5764#	5767#	5781	5784#	5793#	5795#	5808#	5810#	5829#	5832#	5835#	5842#
5864#	5870#	5876	5879#	5885	5893#	5895#	5897#	5899#	5914#	5921	5928#	5959#
5964#	5966#	5971#	5973#	5978#	5980#	5985#	5987#	5992#	5994#	5999#	6001#	6006#
6026#	6031#	6041#	6046#	6067#	6072#	6074#	6079#	6081#	6086#	6088#	6093#	6095#
6100#	6102#	6107#	6109#	6114#	6118#	6123#	6127#	6132#	6137#	6142#	6148#	6153#
6280#	6285#	6290#	6292#	6295#	6300	6307#	6309#	6311#	6314#	6319	6326#	6328#
6330#	6332#	6354#	6356#	6367#	6384#	6386#	6394#	6406#	6417#	6419#	6425#	6430
6435#	6440#	6447#	6451#	6453#	6459#	6465#	6467#	6474	6479#	6481#	6483#	6498#
6505	6512#	6543#	6548#	6550#	6555#	6557#	6562#	6564#	6569#	6571#	6577#	6578#
6583#	6585#	6590#	6610#	6615#	6625#	6630#	6651#	6656#	6658#	6663#	6669#	6670#
6672#	6677#	6679#	6684#	6686#	6691#	6693#	6698#	6702#	6707#	6713#	6718#	6830#
6837	6844#	6866#	6875	6885#	6894#	6942#	6948#	6960#	6965#	6967#	6975#	6977#
6979#	6981#	7013#	7018#	7020#	7025#	7027#	7032#	7034#	7039#	7041#	7046#	7048#
7053#	7055#	7060#	7068#	7073#	7093#	7098#	7100#	7105#	7107#	7112#	7114#	7119#

7121#	7126#	7128#	7133#	7135#	7140#	7142#	7147#	7149#	7154#	7255#	7262	7269#
7291#	7300	7318#	7324#	7327#	7338#	7345#	7347#	7349#	7355#	7360	7367#	7384#
7391#	7396#	7398#	7404#	7406#	7412#	7419#	7421#	7423#	7442#	7447#	7455#	7457#
7489#	7494#	7496#	7501#	7503#	7508#	7510#	7515#	7517#	7522#	7524#	7529#	7531#
7536#	7544#	7549#	7569#	7574#	7576#	7586#	7588#	7598#	7600#	7605#	7607#	7612#
8006#	8013	8020#	8063#	8071	8074#	8079	8084#	8086#	8095#	8100	8105#	8165#
8180#	8205#	8212#	8221#	8229#	8237#	8242#	8295#	8300#	8307#	8315#	8319#	8324#
8330#	8335	8340#	8342#	8344#	8348#	8351#	8359#	8363#	8367#	8370#	8377#	8379#
8382#	8389#	8397#	8402#	8406#	8408#	8412#	8415#	8421#	8426#	8430#	8432#	8436#
8442#	8447	8452#	8454#	8458#	8466#	8471	8476#	8478#	8508#	8513#	8515#	8520#
8522#	8527#	8529#	8534#	8536#	8541#	8543#	8548#	8550#	8555#	8571#	8576#	8592#
8597#	8617#	8622#	8624#	8634#	8636#	8646#	8648#	8653#	8655#	8660#	8662#	8667#
8669#	8674#	8855#	8858#	8863#	8868	8873#	8875#	8877#	8882#	8885#	8892#	8894#
8899#	8902#	8909#	8914#	8918#	8920#	8925#	8928#	8933#	8938#	8942#	8944#	8949#
8952#	8957	8962#	8964#	8969#	8972#	8977	8982#	8984#	9016#	9021	9024#	9032
9037#	9039#	9041#	9046#	9054#	9059#	9066#	9068#	9076#	9084#	9092#	9097#	9106#
9111	9116#	9142#	9153#	9177#	9190#	9237#	9242#	9244#	9249#	9251#	9256#	9258#
9263#	9265#	9270#	9272#	9277#	9279#	9284#	9300#	9305#	9315#	9320#	9341#	9346#
9348#	9358#	9360#	9370#	9372#	9377#	9379#	9384#	9386#	9391#	9553#	9557#	9562#
9567#	9573	9603#	9605#	9613#	9620#	9625#	9627#	9651#	9662#	9686#	9697#	9721#
9732#	9756#	9767#	9791#	9802#	10152#	10159#	10164#	10166#	10210#	10213#	10218#	10220#
10296#	10313	10318#	10325#	10342	10347#	10354#	10373	10378#	10418#	10423#	10459#	10464
10469#	10482#	10487#	10500#	10505#	10520#	10525	10530#	10544#	10549	10554#	10565#	10570#
10606#	10611	10616#	10629#	10634#	10647#	10652#	10666#	10671	10676#	10690#	10695	10700#
10711#	10716#	10752#	10755#	10760	10765#	10769#	10778#	10785#	10790#	10794#	10803#	10810#
10815#	10819#	10831#	10836	10841#	10855#	10860	10865#	10910#	10915#	10917#	10922#	10924#
10929#	10931#	10936#	10938#	10943#	10945#	10950#	10952#	10957#	10976#	10981#	11006#	11011#
11086#	11091#	11098#	11108#	11110#	11120#	11139#	11144#	11221#	11238	11243#	11250#	11267
11272#	11279#	11298	11303#	11343#	11348#	11384#	11389	11394#	11407#	11412#	11425#	11430#
11445#	11450	11455#	11469#	11474	11479#	11490#	11495#	11533#	11538	11543#	11556#	11561#
11574#	11579#	11593#	11598	11603#	11617#	11622	11627#	11638#	11643#	11679#	11682#	11687
11692#	11696#	11705#	11712#	11717#	11721#	11730#	11737#	11742#	11746#	11758#	11763	11768#
11782#	11787	11792#	11837#	11842#	11844#	11849#	11851#	11856#	11858#	11863#	11865#	11870#
11872#	11877#	11879#	11884#	11904#	11909#	11934#	11939#	12014#	12019#	12026#	12036#	12038#
12048#	12067#	12072#	12147#	12152#	12182#	12189	12192#	12199	12202#	12209#	12211#	12213#
12215#	12220#	12223#	12228#	12230	12235#	12242	12245#	12252#	12254#	12256#	12281#	12284#
12291	12294#	12301#	12303#	12305	12310#	12312#	12314#	12316#	12332#	12337#	12348#	12354#
12359	12364#	12370#	12372#	12375#	12381#	12386	12391#	12397#	12399#	12401#	12404#	12409#
12414	12419#	12425#	12427#	12429#	12435#	12440	12445#	12447#	12449#	12454#	12459	12464#
12471#	12478#	12483#	12494#	12500#	12505	12510#	12516#	12518#	12524#	12529	12534#	12540#
12542#	12547#	12552	12557#	12563#	12565#	12571#	12576	12581#	12583#	12585#	12590#	12595
12600#	12607#	12639#	12644#	12646#	12651#	12653#	12658#	12660#	12665#	12667#	12672#	12674#
12679#	12681#	12686#	12691#	12696#	12701#	12706#	12714#	12719#	12749#	12754#	12756#	12766#
12768#	12778#	12867#	12872#	12879#	12886#	12891#	12909#	12911#	12919	12925#	12938#	12941#
12946	12951#	12953#	12955#	12960#	12970#	12977#	12986#	12993#	12995#	13000#	13006#	13011#
13013#	13033	13067#	13069#	13071#	13073#	13078#	13109#	13114#	13118#	13128#	13131#	13141#
13144#	13149#	13155#	13160	13165#	13167#	13171#	13177#	13187	13192#	13194#	13203#	13205#
13207#	13209#	13214#	13222#	13224#	13232#	13234#	13236#	13239#	13242#	13249#	13254#	13259
13264#	13266#	13268#	13274#	13284	13289#	13291#	13300#	13308#	13310#	13312#	13315#	13318#
13323	13328#	13330#	13332#	13335#	13345	13350#	13352#	13361#	13369#	13371#	13373#	13382#
13387	13392#	13400#	13408	13413#	13415#	13424#	13426#	13431#	13433#	13435#	13451#	13456
13461#	13463#	13471	13476#	13478#	13487#	13489#	13494#	13496#	13528#	13533#	13535#	13545#
13550#	13555#	13557#	13562#	13579#	13584#	13591#	13596#	13620#	13627#	13630#	13635#	13682#
13687#	13689#	13699#	13701#	13711#	13716#	13721#	13723#	13728#	13738#	13745#	13800#	13803#
13808#	13810#	13959#	13964#	13966#	13971#	13973#	13978#	13980#	13985#	13987#	13992#	13994#
13999#	14001#	14006#	14025#	14030#	14040#	14045#	14127#	14132#	14139#	14149#	14151#	14161#

14189#	14194#	14345#	14350#	14352#	14357#	14359#	14364#	14366#	14371#	14373#	14378#	14380#
14385#	14387#	14392#	14412#	14417#	14427#	14432#	14515#	14520#	14527#	14537#	14539#	14549#
14577#	14582#	14688#	14695	14698#	14705	14708#	14715#	14717#	14719#	14721#	14726#	14729#
14734#	14736	14741#	14748	14751#	14758#	14760#	14762#	14787#	14790#	14797	14800#	14807#
14809#	14811	14816#	14818#	14820#	14822#	14842#	14845#	14850	14855#	14857	14860#	14867#
14890#	14895	14900#	14904#	14914#	14917#	14922	14927#	14929	14932#	14939#	14962#	14967
14972#	14976#	15008#	15013#	15015#	15025#	15027#	15037#	15044#	15049#	15080#	15085#	15087#
15097#	15099#	15109#	15178#	15181#	15186	15191#	15193	15196#	15203#	15216#	15221	15226#
15228#	15239#	15244#	15251#	15258#	15269#	15277	15282#	15287#	15295#	15297#	15320#	15323#
15328	15343#	15345#	15347#	15362#	15370#	15375#	15382#	15386	15391#	15400#	15407#	15409#
15411#	15422#	15428#	15433#	15435#	15442	15470#	15477	15482#	15493#	15495#	15497#	15527#
15532#	15534#	15542#	15544#	15552#	15557#	15562#	15564#	15569#	15587#	15592#	15599#	15604#
15609#	15616#	15619#	15624#	15655#	15660#	15662#	15672#	15674#	15684#	15689#	15694#	15696#
15701#	15711#	15718#	15772#	15775#	15780#	15782#	15932#	15937#	15939#	15944#	15946#	15951#
15953#	15958#	15960#	15965#	15967#	15972#	15974#	15979#	15999#	16004#	16014#	16019#	16099#
16104#	16111#	16121#	16123#	16133#	16161#	16166#	16205#	16208#	16213#	16215#	16370#	16375#
16377#	16382#	16384#	16389#	16391#	16396#	16398#	16403#	16405#	16410#	16412#	16417#	16437#
16442#	16452#	16457#	16541#	16546#	16553#	16563#	16565#	16575#	16603#	16608#	16859#	16870#
17262#	17267	17270#	17275#	17277#	17284#	17320#	17327#	17335	17340#	17402#	17407	17412#
17422#	17427	17430#	17437#	17464#								
720#												
720#	1267#	1269#	1270#	1272#	1275#	1277#	1312#	1314#	1330#	1343#	1345#	1346#
1348#	1351#	1356#	1358#	1361#	1363#	1365#	1367#	1370#	1372#	1374#	1377#	1379#
1381#	1384#	1386#	1388#	1391#	1393#	1395#	1398#	1400#	1402#	1403#	1405#	1408#
1413#	1415#	1418#	1420#	1448#	1450#	1453#	1455#	1457#	1460#	1462#	1464#	1467#
1469#	1471#	1474#	1476#	1478#	1481#	1483#	1485#	1488#	1532#	1534#	1550#	1563#
1565#	1566#	1568#	1571#	1576#	1578#	1581#	1583#	1585#	1587#	1590#	1592#	1594#
1597#	1599#	1601#	1604#	1606#	1608#	1611#	1613#	1615#	1618#	1620#	1622#	1625#
1651#	1653#	1656#	1658#	1660#	1663#	1665#	1667#	1670#	1672#	1674#	1677#	1679#
1681#	1684#	1686#	1688#	1691#	1693#	1695#	1698#	1760#	1762#	1764#	1778#	1780#
1783#	1792#	1806#	1808#	1814#	1816#	1824#	1826#	1830#	1832#	1838#	1840#	1848#
1850#	1852#	1854#	1862#	1880#	1882#	1885#	1887#	1889#	1892#	1894#	1896#	1899#
1901#	1903#	1906#	1908#	1910#	1913#	1915#	1917#	1920#	1922#	1924#	1927#	1940#
1942#	1945#	1946#	1948#	1950#	1951#	1953#	1956#	1968#	1969#	1971#	1973#	1974#
1976#	1979#	1994#	1996#	1999#	2009#	2013#	2015#	2016#	2018#	2026#	2036#	2038#
2041#	2049#	2056#	2058#	2061#	2099#	2101#	2105#	2107#	2109#	2111#	2114#	2116#
2118#	2121#	2123#	2125#	2128#	2130#	2132#	2135#	2137#	2139#	2142#	2144#	2146#
2149#	2151#	2153#	2156#	2158#	2160#	2163#	2172#	2174#	2177#	2376#	2378#	2386#
2394#	2396#	2399#	2408#	2422#	2424#	2430#	2432#	2440#	2442#	2446#	2448#	2454#
2456#	2464#	2466#	2486#	2488#	2491#	2493#	2495#	2498#	2500#	2502#	2505#	2507#
2509#	2512#	2514#	2516#	2519#	2521#	2523#	2526#	2528#	2530#	2533#	2544#	2546#
2553#	2554#	2556#	2558#	2559#	2561#	2564#	2576#	2578#	2581#	2602#	2603#	2605#
2607#	2608#	2610#	2613#	2618#	2620#	2622#	2628#	2630#	2633#	2644#	2651#	2653#
2656#	2694#	2696#	2700#	2702#	2704#	2706#	2709#	2711#	2713#	2716#	2718#	2720#
2723#	2725#	2727#	2730#	2732#	2734#	2737#	2739#	2741#	2744#	2746#	2748#	2751#
2753#	2755#	2758#	2766#	2768#	2771#	2889#	2891#	2894#	2903#	2917#	2919#	2925#
2927#	2935#	2937#	2941#	2943#	2949#	2951#	2959#	2961#	3007#	3009#	3012#	3014#
3016#	3019#	3021#	3023#	3026#	3028#	3030#	3033#	3035#	3037#	3040#	3042#	3044#
3047#	3049#	3051#	3054#	3071#	3073#	3076#	3077#	3079#	3081#	3082#	3084#	3087#
3099#	3100#	3102#	3104#	3105#	3107#	3111#	3126#	3128#	3131#	3141#	3143#	3145#
3155#	3157#	3160#	3168#	3175#	3177#	3180#	3220#	3222#	3226#	3228#	3230#	3232#
3235#	3237#	3239#	3242#	3244#	3246#	3249#	3251#	3253#	3256#	3258#	3260#	3263#
3265#	3267#	3270#	3272#	3274#	3277#	3279#	3281#	3284#	3293#	3295#	3298#	3396#
3413#	3415#	3417#	3432#	3473#	3475#	3478#	3480#	3482#	3485#	3487#	3489#	3492#
3494#	3496#	3499#	3501#	3503#	3506#	3508#	3510#	3513#	3515#	3517#	3520#	3527#
3529#	3532#	3561#	3565#	3569#	3571#	3574#	3576#	3578#	3581#	3583#	3585#	3588#

\$\$\$ERFL= 000000
\$\$\$FLAG= 000001

3590#	3592#	3595#	3597#	3599#	3602#	3604#	3606#	3609#	3682#	3699#	3701#	3703#
3718#	3759#	3761#	3764#	3766#	3768#	3771#	3773#	3775#	3778#	3780#	3782#	3785#
3787#	3789#	3792#	3794#	3796#	3799#	3801#	3803#	3806#	3813#	3815#	3818#	3847#
3851#	3856#	3858#	3861#	3863#	3865#	3868#	3870#	3872#	3875#	3877#	3879#	3882#
3884#	3886#	3889#	3891#	3893#	3896#	3966#	3979#	3981#	3983#	3998#	4023#	4024#
4026#	4055#	4057#	4060#	4062#	4064#	4067#	4069#	4071#	4074#	4076#	4078#	4081#
4083#	4085#	4088#	4090#	4092#	4095#	4097#	4099#	4102#	4109#	4111#	4114#	4133#
4134#	4136#	4159#	4163#	4167#	4169#	4172#	4174#	4176#	4179#	4181#	4183#	4186#
4188#	4190#	4193#	4195#	4197#	4200#	4202#	4204#	4207#	4279#	4292#	4294#	4296#
4311#	4336#	4337#	4339#	4365#	4367#	4370#	4372#	4374#	4377#	4379#	4381#	4384#
4386#	4388#	4391#	4393#	4395#	4398#	4400#	4402#	4405#	4407#	4409#	4412#	4419#
4421#	4424#	4443#	4444#	4446#	4466#	4470#	4474#	4476#	4479#	4481#	4483#	4486#
4488#	4490#	4493#	4495#	4497#	4500#	4502#	4504#	4507#	4509#	4511#	4514#	4590#
4599#	4614#	4616#	4630#	4655#	4657#	4663#	4665#	4667#	4673#	4675#	4677#	4679#
4682#	4700#	4702#	4705#	4707#	4709#	4712#	4714#	4716#	4719#	4721#	4723#	4726#
4728#	4730#	4733#	4735#	4737#	4740#	4742#	4744#	4747#	4754#	4756#	4759#	4794#
4796#	4799#	4801#	4803#	4806#	4808#	4810#	4813#	4815#	4817#	4820#	4822#	4824#
4827#	4829#	4831#	4834#	4836#	4838#	4841#	4934#	4947#	4962#	4964#	4979#	4998#
5000#	5004#	5006#	5021#	5023#	5047#	5049#	5052#	5054#	5056#	5059#	5061#	5063#
5066#	5068#	5070#	5073#	5075#	5077#	5080#	5082#	5084#	5087#	5089#	5091#	5094#
5101#	5103#	5106#	5143#	5147#	5149#	5151#	5154#	5156#	5158#	5161#	5163#	5165#
5168#	5170#	5172#	5175#	5177#	5179#	5182#	5184#	5186#	5189#	5191#	5193#	5196#
5238#	5240#	5241#	5243#	5246#	5248#	5357#	5358#	5360#	5392#	5394#	5408#	5410#
5413#	5415#	5417#	5420#	5422#	5424#	5427#	5429#	5431#	5434#	5436#	5438#	5441#
5443#	5445#	5448#	5450#	5452#	5455#	5474#	5475#	5477#	5479#	5480#	5482#	5485#
5500#	5502#	5505#	5515#	5517#	5526#	5528#	5531#	5539#	5541#	5570#	5572#	5580#
5625#	5627#	5631#	5678#	5680#	5681#	5683#	5686#	5688#	5748#	5750#	5755#	5757#
5760#	5762#	5764#	5766#	5767#	5769#	5784#	5786#	5793#	5795#	5797#	5808#	5810#
5812#	5829#	5832#	5834#	5835#	5837#	5842#	5846#	5847#	5849#	5864#	5870#	5872#
5879#	5881#	5893#	5895#	5897#	5899#	5914#	5916#	5928#	5959#	5961#	5964#	5966#
5968#	5971#	5973#	5975#	5978#	5980#	5982#	5985#	5987#	5989#	5992#	5994#	5996#
5999#	6001#	6003#	6006#	6020#	6021#	6023#	6025#	6026#	6028#	6031#	6041#	6043#
6046#	6067#	6069#	6072#	6074#	6076#	6079#	6081#	6083#	6086#	6088#	6090#	6093#
6095#	6097#	6100#	6102#	6104#	6107#	6109#	6111#	6114#	6118#	6120#	6123#	6127#
6129#	6132#	6137#	6139#	6142#	6148#	6150#	6153#	6280#	6282#	6285#	6287#	6290#
6292#	6294#	6295#	6297#	6307#	6309#	6311#	6313#	6314#	6316#	6326#	6328#	6330#
6332#	6334#	6336#	6341#	6342#	6344#	6354#	6356#	6358#	6360#	6367#	6369#	6375#
6376#	6378#	6384#	6386#	6388#	6394#	6406#	6408#	6410#	6417#	6419#	6421#	6424#
6425#	6427#	6435#	6440#	6442#	6447#	6451#	6453#	6455#	6458#	6459#	6461#	6465#
6467#	6469#	6479#	6481#	6483#	6498#	6500#	6512#	6543#	6545#	6548#	6550#	6552#
6555#	6557#	6559#	6562#	6564#	6566#	6569#	6571#	6573#	6576#	6578#	6580#	6583#
6585#	6587#	6590#	6604#	6605#	6607#	6609#	6610#	6612#	6615#	6625#	6627#	6630#
6651#	6653#	6656#	6658#	6660#	6663#	6665#	6667#	6670#	6672#	6674#	6677#	6679#
6681#	6684#	6686#	6688#	6691#	6693#	6695#	6698#	6702#	6704#	6707#	6713#	6715#
6718#	6830#	6832#	6844#	6866#	6868#	6885#	6887#	6894#	6896#	6897#	6899#	6942#
6944#	6948#	6950#	6953#	6960#	6962#	6965#	6967#	6969#	6972#	6975#	6977#	6979#
6981#	7013#	7015#	7018#	7020#	7022#	7025#	7027#	7029#	7032#	7034#	7036#	7039#
7041#	7043#	7046#	7048#	7050#	7053#	7055#	7057#	7060#	7068#	7070#	7073#	7093#
7095#	7098#	7100#	7102#	7105#	7107#	7109#	7112#	7114#	7116#	7119#	7121#	7123#
7126#	7128#	7130#	7133#	7135#	7137#	7140#	7142#	7144#	7147#	7149#	7151#	7154#
7255#	7257#	7269#	7291#	7293#	7318#	7324#	7326#	7327#	7329#	7332#	7338#	7340#
7345#	7347#	7349#	7351#	7354#	7355#	7357#	7367#	7384#	7386#	7391#	7396#	7398#
7400#	7404#	7406#	7408#	7411#	7412#	7414#	7419#	7421#	7423#	7442#	7444#	7447#
7449#	7452#	7455#	7457#	7489#	7491#	7494#	7496#	7498#	7501#	7503#	7505#	7508#
7510#	7512#	7515#	7517#	7519#	7522#	7524#	7526#	7529#	7531#	7533#	7536#	7544#
7546#	7549#	7569#	7571#	7574#	7576#	7578#	7580#	7583#	7586#	7588#	7590#	7592#

7595#	7598#	7600#	7602#	7605#	7607#	7609#	7612#	8006#	8008#	8020#	8063#	8065#
8068#	8074#	8076#	8084#	8086#	8095#	8097#	8105#	8151#	8152#	8154#	8165#	8167#
8180#	8190#	8191#	8193#	8205#	8207#	8209#	8212#	8221#	8223#	8226#	8229#	8237#
8239#	8242#	8279#	8280#	8282#	8295#	8297#	8300#	8307#	8309#	8315#	8319#	8321#
8324#	8326#	8330#	8332#	8340#	8342#	8344#	8348#	8350#	8351#	8353#	8359#	8363#
8367#	8369#	8370#	8372#	8374#	8377#	8379#	8381#	8382#	8384#	8386#	8389#	8397#
8399#	8402#	8406#	8408#	8412#	8414#	8415#	8417#	8421#	8423#	8426#	8430#	8432#
8436#	8438#	8442#	8444#	8452#	8454#	8458#	8460#	8466#	8468#	8476#	8478#	8508#
8510#	8513#	8515#	8517#	8520#	8522#	8524#	8527#	8529#	8531#	8534#	8536#	8538#
8541#	8543#	8545#	8548#	8550#	8552#	8555#	8565#	8566#	8568#	8570#	8571#	8573#
8576#	8592#	8594#	8597#	8617#	8619#	8622#	8624#	8626#	8628#	8631#	8634#	8636#
8638#	8640#	8643#	8646#	8648#	8650#	8653#	8655#	8657#	8660#	8662#	8664#	8667#
8669#	8671#	8674#	8855#	8857#	8858#	8860#	8863#	8865#	8873#	8875#	8877#	8882#
8884#	8885#	8887#	8892#	8894#	8899#	8901#	8902#	8904#	8909#	8911#	8914#	8918#
8920#	8925#	8927#	8928#	8930#	8933#	8935#	8938#	8942#	8944#	8949#	8951#	8952#
8954#	8962#	8964#	8969#	8971#	8972#	8974#	8982#	8984#	9016#	9018#	9024#	9026#
9029#	9037#	9039#	9041#	9043#	9046#	9054#	9056#	9058#	9059#	9061#	9063#	9066#
9068#	9076#	9078#	9081#	9084#	9092#	9094#	9097#	9106#	9108#	9116#	9142#	9144#
9153#	9163#	9164#	9166#	9177#	9179#	9190#	9237#	9239#	9242#	9244#	9246#	9249#
9251#	9253#	9256#	9258#	9260#	9263#	9265#	9267#	9270#	9272#	9274#	9277#	9279#
9281#	9284#	9294#	9295#	9297#	9299#	9300#	9302#	9305#	9315#	9317#	9320#	9341#
9343#	9346#	9348#	9350#	9352#	9355#	9358#	9360#	9362#	9364#	9367#	9370#	9372#
9374#	9377#	9379#	9381#	9384#	9386#	9388#	9391#	9553#	9557#	9562#	9564#	9566#
9567#	9569#	9603#	9605#	9613#	9615#	9620#	9622#	9625#	9627#	9636#	9637#	9639#
9651#	9653#	9655#	9662#	9671#	9672#	9674#	9686#	9688#	9690#	9697#	9706#	9707#
9709#	9721#	9723#	9725#	9732#	9741#	9742#	9744#	9756#	9758#	9760#	9767#	9776#
9777#	9779#	9791#	9793#	9795#	9802#	9811#	9812#	9814#	9824#	9825#	9827#	9842#
9843#	9845#	9860#	9861#	9863#	9878#	9879#	9881#	9900#	9901#	9903#	9956#	9957#
9959#	9978#	9979#	9981#	10000#	10001#	10003#	10022#	10023#	10025#	10044#	10045#	10047#
10066#	10067#	10069#	10086#	10087#	10089#	10106#	10107#	10109#	10126#	10127#	10129#	10152#
10154#	10159#	10161#	10164#	10166#	10210#	10212#	10213#	10215#	10218#	10220#	10296#	10298#
10318#	10325#	10327#	10347#	10354#	10356#	10378#	10418#	10420#	10423#	10459#	10461#	10469#
10482#	10484#	10487#	10500#	10502#	10505#	10520#	10522#	10530#	10544#	10546#	10554#	10565#
10567#	10570#	10606#	10608#	10616#	10629#	10631#	10634#	10647#	10649#	10652#	10666#	10668#
10676#	10690#	10692#	10700#	10711#	10713#	10716#	10752#	10754#	10755#	10757#	10765#	10769#
10778#	10780#	10785#	10787#	10790#	10794#	10803#	10805#	10810#	10812#	10815#	10819#	10831#
10833#	10841#	10855#	10857#	10865#	10910#	10912#	10915#	10917#	10919#	10922#	10924#	10926#
10929#	10931#	10933#	10936#	10938#	10940#	10943#	10945#	10947#	10950#	10952#	10954#	10957#
10970#	10971#	10973#	10975#	10976#	10978#	10981#	11006#	11008#	11011#	11086#	11088#	11091#
11098#	11100#	11102#	11105#	11108#	11110#	11112#	11114#	11117#	11120#	11133#	11134#	11136#
11138#	11139#	11141#	11144#	11221#	11223#	11243#	11250#	11252#	11272#	11279#	11281#	11303#
11343#	11345#	11348#	11384#	11386#	11394#	11407#	11409#	11412#	11425#	11427#	11430#	11445#
11447#	11455#	11469#	11471#	11479#	11490#	11492#	11495#	11533#	11535#	11543#	11556#	11558#
11561#	11574#	11576#	11579#	11593#	11595#	11603#	11617#	11619#	11627#	11638#	11640#	11643#
11679#	11681#	11682#	11684#	11692#	11696#	11705#	11707#	11712#	11714#	11717#	11721#	11730#
11732#	11737#	11739#	11742#	11746#	11758#	11760#	11768#	11782#	11784#	11792#	11837#	11839#
11842#	11844#	11846#	11849#	11851#	11853#	11856#	11858#	11860#	11863#	11865#	11867#	11870#
11872#	11874#	11877#	11879#	11881#	11884#	11898#	11899#	11901#	11903#	11904#	11906#	11909#
11934#	11936#	11939#	12014#	12016#	12019#	12026#	12028#	12030#	12033#	12036#	12038#	12040#
12042#	12045#	12048#	12061#	12062#	12064#	12066#	12067#	12069#	12072#	12147#	12149#	12152#
12182#	12184#	12186#	12192#	12194#	12196#	12202#	12204#	12206#	12209#	12211#	12213#	12215#
12217#	12219#	12220#	12222#	12223#	12225#	12228#	12235#	12237#	12239#	12245#	12247#	12249#
12252#	12254#	12256#	12258#	12270#	12271#	12273#	12275#	12281#	12283#	12284#	12286#	12288#
12294#	12296#	12298#	12301#	12303#	12310#	12312#	12314#	12316#	12332#	12334#	12337#	12348#
12350#	12354#	12356#	12364#	12370#	12372#	12374#	12375#	12377#	12381#	12383#	12391#	12397#
12399#	12401#	12403#	12404#	12406#	12409#	12411#	12419#	12425#	12427#	12429#	12431#	12435#

12437#	12445#	12447#	12449#	12451#	12454#	12456#	12464#	12471#	12478#	12480#	12483#	12494#
12496#	12500#	12502#	12510#	12516#	12518#	12520#	12524#	12526#	12534#	12540#	12542#	12544#
12547#	12549#	12557#	12563#	12565#	12567#	12571#	12573#	12581#	12583#	12585#	12587#	12590#
12592#	12600#	12607#	12639#	12641#	12644#	12646#	12648#	12651#	12653#	12655#	12658#	12660#
12662#	12665#	12667#	12669#	12672#	12674#	12676#	12679#	12681#	12683#	12686#	12691#	12693#
12696#	12701#	12703#	12706#	12714#	12716#	12719#	12749#	12751#	12754#	12756#	12758#	12760#
12763#	12766#	12768#	12770#	12772#	12775#	12778#	12847#	12848#	12850#	12867#	12869#	12872#
12879#	12881#	12883#	12886#	12891#	12893#	12909#	12911#	12913#	12916#	12925#	12927#	12938#
12940#	12941#	12943#	12951#	12953#	12955#	12957#	12960#	12962#	12963#	12965#	12967#	12970#
12972#	12974#	12977#	12986#	12988#	12990#	12993#	12995#	13000#	13002#	13006#	13008#	13011#
13013#	13015#	13054#	13055#	13057#	13059#	13067#	13069#	13071#	13073#	13075#	13078#	13109#
13111#	13114#	13118#	13120#	13128#	13131#	13133#	13141#	13144#	13146#	13149#	13151#	13155#
13157#	13165#	13167#	13171#	13173#	13177#	13179#	13181#	13184#	13192#	13194#	13196#	13198#
13200#	13203#	13205#	13207#	13209#	13211#	13214#	13216#	13222#	13224#	13226#	13232#	13234#
13236#	13238#	13239#	13241#	13242#	13244#	13246#	13249#	13254#	13256#	13264#	13266#	13268#
13270#	13274#	13276#	13278#	13281#	13289#	13291#	13293#	13295#	13297#	13300#	13308#	13310#
13312#	13314#	13315#	13317#	13318#	13320#	13328#	13330#	13332#	13334#	13335#	13337#	13339#
13342#	13350#	13352#	13354#	13356#	13358#	13361#	13369#	13371#	13373#	13375#	13382#	13384#
13392#	13400#	13402#	13405#	13413#	13415#	13417#	13419#	13421#	13424#	13426#	13428#	13431#
13433#	13435#	13437#	13451#	13453#	13461#	13463#	13465#	13468#	13476#	13478#	13480#	13482#
13484#	3487#	13489#	13491#	13494#	13496#	13528#	13530#	13533#	13535#	13537#	13539#	13542#
13545#	13550#	13552#	13555#	13557#	13559#	13562#	13573#	13574#	13576#	13578#	13579#	13581#
13584#	13591#	13593#	13596#	13620#	13622#	13627#	13630#	13632#	13635#	13682#	13684#	13687#
13689#	13691#	13693#	13696#	13699#	13701#	13703#	13705#	13708#	13711#	13716#	13718#	13721#
13723#	13725#	13728#	13732#	13733#	13735#	13737#	13738#	13740#	13745#	13800#	13802#	13803#
13805#	13808#	13810#	13959#	13961#	13964#	13966#	13968#	13971#	13973#	13975#	13978#	13980#
13982#	13985#	13987#	13989#	13992#	13994#	13996#	13999#	14001#	14003#	14006#	14019#	14020#
14022#	14024#	14025#	14027#	14030#	14040#	14042#	14045#	14127#	14129#	14132#	14139#	14141#
14143#	14146#	14149#	14151#	14153#	14155#	14158#	14161#	14183#	14184#	14186#	14188#	14189#
14191#	14194#	14345#	14347#	14350#	14352#	14354#	14357#	14359#	14361#	14364#	14366#	14368#
14371#	14373#	14375#	14378#	14380#	14382#	14385#	14387#	14389#	14392#	14406#	14407#	14409#
14411#	14412#	14414#	14417#	14427#	14429#	14432#	14515#	14517#	14520#	14527#	14529#	14531#
14534#	14537#	14539#	14541#	14543#	14546#	14549#	14571#	14572#	14574#	14576#	14577#	14579#
14582#	14688#	14690#	14692#	14698#	14700#	14702#	14708#	14710#	14712#	14715#	14717#	14719#
14721#	14723#	14725#	14726#	14728#	14729#	14731#	14734#	14741#	14743#	14745#	14751#	14753#
14755#	14758#	14760#	14762#	14764#	14776#	14777#	14779#	14781#	14787#	14789#	14790#	14792#
14794#	14800#	14802#	14804#	14807#	14809#	14816#	14818#	14820#	14822#	14842#	14844#	14845#
14847#	14855#	14860#	14862#	14867#	14869#	14870#	14872#	14890#	14892#	14900#	14904#	14914#
14916#	14917#	14919#	14927#	14932#	14934#	14939#	14941#	14942#	14944#	14962#	14964#	14972#
14976#	15008#	15010#	15013#	15015#	15017#	15019#	15022#	15025#	15027#	15029#	15031#	15034#
15037#	15044#	15046#	15049#	15080#	15082#	15085#	15087#	15089#	15091#	15094#	15097#	15099#
15101#	15103#	15106#	15109#	15178#	15180#	15181#	15183#	15191#	15196#	15198#	15203#	15205#
15206#	15208#	15216#	15218#	15226#	15228#	15239#	15241#	15244#	15251#	15253#	15255#	15258#
15269#	15271#	15274#	15282#	15287#	15289#	15292#	15295#	15297#	15299#	15301#	15303#	15320#
15322#	15323#	15325#	15343#	15345#	15347#	15349#	15362#	15364#	15365#	15367#	15369#	15370#
15372#	15375#	15377#	15379#	15382#	15391#	15400#	15402#	15404#	15407#	15409#	15411#	15413#
15415#	15417#	15422#	15424#	15428#	15430#	15433#	15435#	15437#	15464#	15465#	15467#	15469#
15470#	15472#	15482#	15493#	15495#	15497#	15527#	15529#	15532#	15534#	15536#	15539#	15542#
15544#	15546#	15549#	15552#	15557#	15559#	15562#	15564#	15566#	15569#	15581#	15582#	15584#
15586#	15587#	15589#	15592#	15599#	15601#	15604#	15609#	15611#	15616#	15619#	15621#	15624#
15655#	15657#	15660#	15662#	15664#	15666#	15669#	15672#	15674#	15676#	15678#	15681#	15684#
15689#	15691#	15694#	15696#	15698#	15701#	15705#	15706#	15708#	15710#	15711#	15713#	15718#
15772#	15774#	15775#	15777#	15780#	15782#	15932#	15934#	15937#	15939#	15941#	15944#	15946#
15948#	15951#	15953#	15955#	15958#	15960#	15962#	15965#	15967#	15969#	15972#	15974#	15976#
15979#	15993#	15994#	15996#	15998#	15999#	16001#	16004#	16014#	16016#	16019#	16099#	16101#
16104#	16111#	16113#	16115#	16118#	16121#	16123#	16125#	16127#	16130#	16133#	16155#	16156#

SSFROM= 000000

16158#	16160#	16161#	16163#	16166#	16205#	16207#	16208#	16210#	16213#	16215#	16370#	16372#
16375#	16377#	16379#	16382#	16384#	16386#	16389#	16391#	16393#	16396#	16398#	16400#	16403#
16405#	16407#	16410#	16412#	16414#	16417#	16431#	16432#	16434#	16436#	16437#	16439#	16442#
16452#	16454#	16457#	16541#	16543#	16546#	16553#	16555#	16557#	16560#	16563#	16565#	16567#
16569#	16572#	16575#	16597#	16598#	16600#	16602#	16603#	16605#	16608#	16859#	16861#	16863#
16870#	17262#	17264#	17270#	17272#	17275#	17277#	17284#	17286#	17320#	17327#	17329#	17340#
17402#	17404#	17412#	17422#	17424#	17430#	17432#	17437#	17439#	17440#	17442#	17464#	
720#	1234#	1235	1310#	1311	1349#	1350	1359#	1360	1368#	1369	1375#	1376
1382#	1383	1389#	1390	1396#	1397	1406#	1407	1416#	1417	1446#	1447	1451#
1452	1458#	1459	1465#	1466	1472#	1473	1479#	1480	1486#	1487	1530#	1531
1569#	1570	1579#	1580	1588#	1589	1595#	1596	1602#	1603	1609#	1610	1616#
1617	1623#	1624	1649#	1650	1654#	1655	1661#	1662	1668#	1669	1675#	1676
1682#	1683	1689#	1690	1696#	1697	1809#	1810	1817#	1818	1822#	1823	1833#
1834	1841#	1842	1846#	1847	1883#	1884	1890#	1891	1897#	1898	1904#	1905
1911#	1912	1918#	1919	1925#	1926	1954#	1955	1977#	1978	1997#	1998	2039#
2040	2071#	2072	2075#	2076	2103#	2104	2112#	2113	2119#	2120	2126#	2127
2133#	2134	2140#	2141	2147#	2148	2154#	2155	2161#	2162	2175#	2176	2425#
2426	2433#	2434	2438#	2439	2449#	2450	2457#	2458	2462#	2463	2489#	2490
2496#	2497	2503#	2504	2510#	2511	2517#	2518	2524#	2525	2531#	2532	2562#
2563	2579#	2580	2611#	2612	2631#	2632	2666#	2667	2670#	2671	2698#	2699
2707#	2708	2714#	2715	2721#	2722	2728#	2729	2735#	2736	2742#	2743	2749#
2750	2756#	2757	2769#	2770	2920#	2921	2928#	2929	2933#	2934	2944#	2945
2952#	2953	2957#	2958	3010#	3011	3017#	3018	3024#	3025	3031#	3032	3038#
3039	3045#	3046	3052#	3053	3085#	3086	3108#	3109	3129#	3130	3158#	3159
3190#	3191	3194#	3195	3224#	3225	3233#	3234	3240#	3241	3247#	3248	3254#
3255	3261#	3262	3268#	3269	3275#	3276	3282#	3283	3296#	3297	3450#	3451
3476#	3477	3483#	3484	3490#	3491	3497#	3498	3504#	3505	3511#	3512	3518#
3519	3542#	3543	3544#	3545	3563#	3564	3572#	3573	3579#	3580	3586#	3587
3593#	3594	3600#	3601	3607#	3608	3736#	3737	3762#	3763	3769#	3770	3776#
3777	3783#	3784	3790#	3791	3797#	3798	3804#	3805	3828#	3829	3830#	3831
3849#	3850	3859#	3860	3866#	3867	3873#	3874	3880#	3881	3887#	3888	3894#
3895	4016#	4017	4058#	4059	4065#	4066	4072#	4073	4079#	4080	4086#	4087
4093#	4094	4100#	4101	4124#	4125	4147#	4148	4161#	4162	4170#	4171	4177#
4178	4184#	4185	4191#	4192	4198#	4199	4205#	4206	4329#	4330	4368#	4369
4375#	4376	4382#	4383	4389#	4390	4396#	4397	4403#	4404	4410#	4411	4434#
4435	4452#	4453	4468#	4469	4477#	4478	4484#	4485	4491#	4492	4498#	4499
4505#	4506	4512#	4513	4608#	4609	4612#	4613	4703#	4704	4710#	4711	4717#
4718	4724#	4725	4731#	4732	4738#	4739	4745#	4746	4769#	4770	4773#	4774
4775#	4776	4797#	4798	4804#	4805	4811#	4812	4818#	4819	4825#	4826	4832#
4833	4839#	4840	4956#	4957	4960#	4961	5050#	5051	5057#	5058	5064#	5065
5071#	5072	5078#	5079	5085#	5086	5092#	5093	5116#	5117	5120#	5121	5131#
5132	5145#	5146	5152#	5153	5159#	5160	5166#	5167	5173#	5174	5180#	5181
5187#	5188	5194#	5195	5279#	5280	5324#	5325	5328#	5329	5395#	5396	5411#
5412	5418#	5419	5425#	5426	5432#	5433	5439#	5440	5446#	5447	5453#	5454
5483#	5484	5503#	5504	5529#	5530	5551#	5552	5612#	5613	5735#	5736	5741#
5742	5962#	5963	5969#	5970	5976#	5977	5983#	5984	5990#	5991	5997#	5998
6004#	6005	6029#	6030	6053#	6054	6055#	6056	6070#	6071	6077#	6078	6084#
6085	6091#	6092	6098#	6099	6105#	6106	6112#	6113	6121#	6122	6130#	6131
6140#	6141	6151#	6152	6264#	6265	6270#	6271	6546#	6547	6553#	6554	6560#
6561	6567#	6568	6574#	6575	6581#	6582	6588#	6589	6613#	6614	6637#	6638
6639#	6640	6654#	6655	6661#	6662	6668#	6669	6675#	6676	6682#	6683	6689#
6690	6696#	6697	6705#	6706	6716#	6717	6855#	6856	6883#	6884	6937#	6938
7016#	7017	7023#	7024	7030#	7031	7037#	7038	7044#	7045	7051#	7052	7058#
7059	7079#	7080	7081#	7082	7096#	7097	7103#	7104	7110#	7111	7117#	7118
7124#	7125	7131#	7132	7138#	7139	7145#	7146	7152#	7153	7440#	7441	7492#
7493	7499#	7500	7506#	7507	7513#	7514	7520#	7521	7527#	7528	7534#	7535

7555#	7556	7557#	7558	7572#	7573	7584#	7585	7596#	7597	7603#	7604	7610#
7611	8032#	8033	8511#	8512	8518#	8519	8525#	8526	8532#	8533	8539#	8540
3546#	8547	8553#	8554	8574#	8575	8603#	8604	8605#	8606	8620#	8621	8632#
8633	8644#	8645	8651#	8652	8658#	8659	8665#	8666	8672#	8673	8800#	8801
8819#	8820	9240#	9241	9247#	9248	9254#	9255	9261#	9262	9268#	9269	9275#
9276	9282#	9283	9303#	9304	9327#	9328	9329#	9330	9344#	9345	9356#	9357
9368#	9369	9375#	9376	9382#	9383	9389#	9390	9540#	9541	9914#	9915	9942#
9943	10168#	10169	10268#	10269	10913#	10914	10920#	10921	10927#	10928	10934#	10935
10941#	10942	10948#	10949	10955#	10956	10979#	10980	11018#	11019	11071#	11072	11089#
11090	11106#	11107	11118#	11119	11142#	11143	11189#	11190	11840#	11841	11847#	11848
11854#	11855	11861#	11862	11868#	11869	11875#	11876	11882#	11883	11907#	11908	11946#
11947	11999#	12000	12017#	12018	12034#	12035	12046#	12047	12070#	12071	12099#	12100
12642#	12643	12649#	12650	12656#	12657	12663#	12664	12670#	12671	12677#	12678	12684#
12685	12752#	12753	12764#	12765	12776#	12777	12836#	12837	13531#	13532	13543#	13544
13553#	13554	13560#	13561	13594#	13595	13643#	13644	13685#	13686	13697#	13698	13709#
13710	13719#	13720	13726#	13727	13741#	13742	13858#	13859	13888#	13891	13895	13898#
13901	13905	13908#	13911	13915	13962#	13963	13969#	13970	13976#	13977	13983#	13984
13990#	13991	13997#	13998	14004#	14005	14028#	14029	14052#	14053	14080#	14083	14087
14090#	14093	14097	14100#	14103	14107	14111#	14112	14130#	14131	14147#	14148	14159#
14160	14192#	14193	14239#	14240	14273#	14276	14280	14283#	14286	14290	14293#	14296
14300	14348#	14349	14355#	14356	14362#	14363	14369#	14370	14376#	14377	14383#	14384
14390#	14391	14415#	14416	14439#	14440	14471#	14474	14478	14481#	14484	14488	14491#
14494	14498	14501#	14502	14518#	14519	14535#	14536	14547#	14548	14580#	14581	14609#
14610	14832#	14833	15011#	15012	15023#	15024	15035#	15036	15083#	15084	15095#	15096
15107#	15108	15150#	15151	15530#	15531	15540#	15541	15550#	15551	15560#	15561	15567#
15568	15602#	15603	15633#	15634	15641#	15642	15658#	15659	15670#	15671	15682#	15683
15692#	15693	15699#	15700	15716#	15717	15831#	15832	15861#	15864	15868	15871#	15874
15878	15881#	15884	15888	15935#	15936	15942#	15943	15949#	15950	15956#	15957	15963#
15964	15970#	15971	15977#	15978	16002#	16003	16026#	16027	16054#	16057	16061	16064#
16067	16071	16074#	16077	16081	16084#	16085	16102#	16103	16119#	16120	16131#	16132
16164#	16165	16264#	16265	16298#	16301	16305	16308#	16311	16315	16318#	16321	16325
16373#	16374	16380#	16381	16387#	16388	16394#	16395	16401#	16402	16408#	16409	16415#
16416	16440#	16441	16464#	16465	16496#	16499	16503	16506#	16509	16513	16516#	16519
16523	16526#	16527	16544#	16545	16561#	16562	16573#	16574	16606#	16607		
720#	1267#	1270#	1312#	1314#	1343#	1345#	1346#	1356#	1358#	1365#	1367#	1372#
1374#	1379#	1381#	1386#	1388#	1393#	1400#	1402#	1403#	1413#	1415#	1448#	1450#
1455#	1457#	1462#	1464#	1469#	1471#	1476#	1483#	1532#	1534#	1563#	1565#	1566#
1576#	1578#	1585#	1587#	1592#	1594#	1599#	1601#	1606#	1608#	1613#	1615#	1620#
1622#	1651#	1653#	1658#	1660#	1665#	1667#	1672#	1674#	1679#	1681#	1686#	1688#
1693#	1695#	1760#	1778#	1780#	1806#	1814#	1816#	1830#	1838#	1840#	1852#	1880#
1882#	1887#	1889#	1894#	1896#	1901#	1903#	1908#	1910#	1915#	1917#	1922#	1924#
1940#	1942#	1946#	1948#	1951#	1953#	1969#	1971#	1974#	1976#	1994#	1996#	2003#
2013#	2015#	2016#	2018#	2036#	2038#	2043#	2056#	2100#	2102#	2109#	2111#	2116#
2118#	2123#	2125#	2130#	2132#	2137#	2139#	2144#	2146#	2151#	2153#	2158#	2160#
2172#	2179#	2376#	2394#	2396#	2422#	2430#	2432#	2446#	2454#	2456#	2486#	2488#
2493#	2495#	2500#	2502#	2507#	2509#	2514#	2516#	2521#	2523#	2528#	2530#	2544#
2546#	2554#	2556#	2559#	2561#	2576#	2578#	2587#	2603#	2605#	2608#	2610#	2620#
2622#	2628#	2630#	2638#	2651#	2695#	2697#	2704#	2706#	2711#	2713#	2718#	2720#
2725#	2727#	2732#	2734#	2739#	2741#	2746#	2748#	2753#	2755#	2766#	2773#	2889#
2891#	2917#	2925#	2927#	2941#	2949#	2951#	3007#	3009#	3014#	3016#	3021#	3023#
3028#	3030#	3035#	3037#	3042#	3044#	3049#	3051#	3071#	3073#	3077#	3079#	3082#
3084#	3100#	3102#	3105#	3107#	3126#	3128#	3135#	3143#	3145#	3155#	3157#	3162#
3175#	3221#	3223#	3230#	3232#	3237#	3239#	3244#	3246#	3251#	3253#	3258#	3260#
3265#	3267#	3272#	3274#	3279#	3281#	3293#	3295#	3300#	3397#	3415#	3417#	3473#
3475#	3480#	3482#	3487#	3489#	3494#	3496#	3501#	3503#	3508#	3510#	3515#	3517#
3527#	3562#	3569#	3571#	3576#	3578#	3583#	3590#	3597#	3604#	3683#	3701#	3703#

SSINH 000403

3759#	3761#	3766#	3768#	3773#	3775#	3780#	3782#	3787#	3789#	3794#	3796#	3801#
3803#	3813#	3848#	3856#	3858#	3863#	3865#	3870#	3877#	3884#	3891#	3967#	3981#
3983#	4024#	4026#	4055#	4057#	4062#	4064#	4069#	4071#	4076#	4078#	4083#	4085#
4090#	4092#	4097#	4099#	4109#	4134#	4136#	4160#	4167#	4169#	4174#	4176#	4181#
4188#	4195#	4202#	4280#	4294#	4296#	4337#	4339#	4365#	4367#	4372#	4374#	4379#
4381#	4386#	4388#	4393#	4395#	4400#	4402#	4407#	4409#	4419#	4444#	4446#	4467#
4474#	4476#	4481#	4483#	4488#	4495#	4502#	4509#	4591#	4614#	4616#	4655#	4657#
4665#	4667#	4675#	4677#	4679#	4700#	4702#	4707#	4709#	4714#	4716#	4721#	4723#
4728#	4730#	4735#	4737#	4742#	4744#	4754#	4794#	4796#	4801#	4808#	4815#	4822#
4829#	4836#	4935#	4962#	4964#	4976#	4998#	5004#	5006#	5047#	5049#	5054#	5056#
5061#	5063#	5068#	5070#	5075#	5077#	5082#	5084#	5089#	5091#	5101#	5130#	5144#
5149#	5151#	5156#	5158#	5163#	5170#	5177#	5184#	5191#	5238#	5240#	5241#	5358#
5386#	5390#	5392#	5394#	5408#	5410#	5415#	5417#	5422#	5424#	5429#	5431#	5436#
5438#	5443#	5445#	5450#	5452#	5475#	5477#	5480#	5482#	5500#	5502#	5509#	5515#
5517#	5526#	5528#	5533#	5570#	5625#	5678#	5681#	5748#	5755#	5764#	5767#	5784#
5786#	5795#	5797#	5810#	5812#	5832#	5834#	5835#	5847#	5849#	5870#	5872#	5879#
5914#	5959#	5961#	5966#	5968#	5973#	5975#	5980#	5982#	5987#	5989#	5994#	5996#
6001#	6003#	6021#	6023#	6026#	6028#	6041#	6067#	6074#	6081#	6088#	6095#	6102#
6109#	6118#	6120#	6127#	6129#	6136#	6137#	6139#	6148#	6155#	6280#	6285#	6292#
6294#	6295#	6311#	6313#	6314#	6332#	6334#	6342#	6356#	6358#	6367#	6369#	6376#
6378#	6386#	6388#	6405#	6406#	6408#	6410#	6419#	6421#	6423#	6425#	6427#	6440#
6442#	6453#	6455#	6457#	6459#	6461#	6467#	6469#	6498#	6543#	6545#	6550#	6552#
6557#	6559#	6564#	6566#	6571#	6573#	6578#	6580#	6585#	6587#	6605#	6607#	6610#
6612#	6625#	6651#	6658#	6665#	6672#	6679#	6686#	6693#	6702#	6704#	6713#	6715#
6720#	6830#	6866#	6885#	6897#	6899#	6942#	6944#	6948#	6950#	6952#	6960#	6967#
6969#	7013#	7015#	7020#	7022#	7027#	7029#	7034#	7036#	7041#	7043#	7048#	7050#
7055#	7057#	7068#	7093#	7100#	7107#	7114#	7121#	7128#	7135#	7142#	7144#	7149#
7151#	7255#	7291#	7324#	7327#	7329#	7331#	7338#	7340#	7349#	7351#	7353#	7355#
7357#	7381#	7384#	7398#	7400#	7406#	7408#	7410#	7412#	7414#	7442#	7444#	7447#
7449#	7489#	7491#	7496#	7498#	7503#	7505#	7510#	7512#	7517#	7519#	7524#	7526#
7531#	7533#	7544#	7569#	7576#	7578#	7580#	7588#	7590#	7592#	7600#	7602#	7607#
7609#	8006#	8063#	8065#	8074#	8076#	8095#	8097#	8152#	8165#	8191#	8193#	8205#
8207#	8209#	8221#	8223#	8237#	8239#	8280#	8282#	8295#	8297#	8307#	8319#	8324#
8330#	8348#	8350#	8351#	8367#	8369#	8370#	8372#	8374#	8379#	8382#	8384#	8386#
8397#	8399#	8412#	8414#	8415#	8421#	8436#	8438#	8442#	8458#	8460#	8466#	8508#
8510#	8515#	8517#	8522#	8524#	8529#	8531#	8536#	8538#	8543#	8545#	8550#	8552#
8566#	8568#	8571#	8573#	8592#	8617#	8619#	8624#	8626#	8628#	8636#	8638#	8640#
8648#	8650#	8655#	8657#	8662#	8664#	8669#	8671#	8855#	8858#	8863#	8882#	8884#
8885#	8899#	8901#	8902#	8909#	8911#	8925#	8927#	8928#	8933#	8935#	8949#	8951#
8952#	8969#	8971#	8972#	9016#	9024#	9026#	9028#	9041#	9043#	9054#	9056#	9058#
9059#	9061#	9076#	9078#	9092#	9094#	9106#	9108#	9142#	9164#	9177#	9237#	9239#
9244#	9246#	9251#	9253#	9258#	9260#	9265#	9267#	9272#	9274#	9279#	9281#	9295#
9297#	9299#	9300#	9302#	9315#	9341#	9348#	9350#	9352#	9360#	9362#	9364#	9372#
9374#	9379#	9381#	9386#	9388#	9554#	9562#	9564#	9567#	9613#	9615#	9620#	9637#
9639#	9651#	9653#	9655#	9672#	9674#	9686#	9688#	9690#	9707#	9709#	9721#	9723#
9725#	9742#	9744#	9756#	9758#	9760#	9777#	9779#	9791#	9793#	9795#	9812#	9814#
9825#	9827#	9843#	9845#	9861#	9863#	9879#	9881#	9901#	9957#	9979#	10001#	10023#
10045#	10067#	10087#	10107#	10127#	10152#	10154#	10159#	10210#	10213#	10296#	10325#	10354#
10418#	10420#	10459#	10482#	10484#	10500#	10502#	10520#	10544#	10565#	10567#	10606#	10629#
10631#	10647#	10649#	10666#	10690#	10711#	10713#	10752#	10755#	10778#	10785#	10787#	10803#
10810#	10812#	10831#	10855#	10910#	10912#	10917#	10919#	10924#	10926#	10931#	10933#	10938#
10940#	10945#	10947#	10952#	10954#	10971#	10973#	10976#	10978#	11006#	11086#	11098#	11100#
11102#	11110#	11112#	11114#	11134#	11136#	11139#	11141#	11221#	11250#	11279#	11343#	11345#
11384#	11407#	11409#	11425#	11427#	11445#	11469#	11490#	11492#	11533#	11556#	11558#	11574#
11576#	11593#	11617#	11638#	11640#	11679#	11682#	11705#	11712#	11714#	11730#	11737#	11739#
11758#	11782#	11837#	11839#	11844#	11846#	11851#	11853#	11858#	11860#	11865#	11867#	11872#

11874#	11879#	11881#	11899#	11901#	11904#	11906#	11934#	12014#	12026#	12028#	12030#	12038#	
12040#	12042#	12062#	12064#	12067#	12069#	12147#	12182#	12184#	12192#	12194#	12202#	12204#	
12206#	12215#	12217#	12220#	12222#	12223#	12225#	12235#	12237#	12245#	12247#	12249#	12256#	
12258#	12271#	12273#	12275#	12281#	12283#	12284#	12286#	12294#	12296#	12298#	12332#	12334#	
12348#	12354#	12372#	12374#	12375#	12381#	12401#	12403#	12404#	12409#	12429#	12431#	12435#	
12449#	12451#	12454#	12478#	12480#	12494#	12500#	12518#	12520#	12524#	12542#	12544#	12547#	
12565#	12567#	12571#	12585#	12587#	12590#	12639#	12641#	12646#	12653#	12660#	12667#	12674#	
12681#	12691#	12693#	12701#	12703#	12714#	12749#	12751#	12756#	12758#	12760#	12762#	12768#	
12770#	12772#	12774#	12848#	12867#	12879#	12881#	12891#	12911#	12913#	12925#	12938#	12941#	
12943#	12955#	12963#	12965#	12970#	12972#	12974#	12986#	12988#	12990#	13000#	13006#	13008#	
13013#	13015#	13055#	13057#	13073#	13109#	13111#	13118#	13131#	13144#	13149#	13155#	13171#	
13177#	13179#	13181#	13194#	13196#	13198#	13200#	13209#	13211#	13214#	13224#	13236#	13238#	
13239#	13242#	13244#	13254#	13268#	13274#	13276#	13278#	13291#	13293#	13295#	13297#	13312#	
13314#	13315#	13318#	13332#	13335#	13337#	13339#	13352#	13354#	13356#	13358#	13373#	13375#	
13382#	13400#	13402#	13415#	13417#	13419#	13421#	13426#	13435#	13437#	13451#	13463#	13465#	
13478#	13480#	13482#	13484#	13489#	13528#	13530#	13535#	13537#	13539#	13550#	13552#	13557#	
13559#	13574#	13576#	13579#	13581#	13591#	13620#	13630#	13682#	13684#	13689#	13691#	13693#	
13701#	13703#	13705#	13716#	13718#	13723#	13725#	13733#	13735#	13737#	13738#	13800#	13803#	
13959#	13961#	13966#	13968#	13973#	13975#	13980#	13982#	13987#	13989#	13994#	13996#	14001#	
14003#	14020#	14022#	14025#	14027#	14040#	14127#	14139#	14141#	14143#	14151#	14153#	14155#	
14184#	14186#	14189#	14191#	14345#	14347#	14352#	14354#	14359#	14361#	14366#	14368#	14373#	
14375#	14380#	14382#	14387#	14389#	14407#	14409#	14412#	14414#	14427#	14515#	14527#	14529#	
14531#	14539#	14541#	14543#	14572#	14574#	14577#	14579#	14688#	14690#	14698#	14700#	14708#	
14710#	14712#	14721#	14723#	14726#	14728#	14729#	14731#	14741#	14743#	14751#	14753#	14755#	
14762#	14764#	14777#	14779#	14781#	14787#	14789#	14790#	14792#	14800#	14802#	14804#	14842#	
14845#	14860#	14870#	14872#	14890#	14914#	14917#	14932#	14942#	14944#	14962#	15008#	15010#	
15015#	15017#	15019#	15027#	15029#	15031#	15044#	15080#	15082#	15087#	15089#	15091#	15093#	
15099#	15101#	15103#	15105#	15178#	15181#	15196#	15206#	15208#	15216#	15239#	15251#	15253#	
15269#	15271#	15287#	15289#	15297#	15299#	15301#	15320#	15323#	15325#	15347#	15365#	15367#	
15370#	15375#	15377#	15379#	15400#	15402#	15404#	15411#	15413#	15415#	15422#	15428#	15430#	
15435#	15437#	15465#	15467#	15470#	15527#	15529#	15534#	15536#	15544#	15546#	15557#	15559#	
15564#	15566#	15582#	15584#	15587#	15589#	15599#	15609#	15619#	15655#	15662#	15664#	15666#	
15674#	15676#	15678#	15689#	15691#	15696#	15698#	15706#	15708#	15710#	15711#	15772#	15775#	
15932#	15934#	15939#	15941#	15946#	15948#	15953#	15955#	15960#	15962#	15967#	15969#	15974#	
15976#	15994#	15996#	15999#	16001#	16014#	16099#	16111#	16113#	16115#	16123#	16125#	16127#	
16156#	16158#	16161#	16163#	16205#	16208#	16370#	16372#	16377#	16379#	16384#	16386#	16391#	
16393#	16398#	16400#	16405#	16407#	16412#	16414#	16432#	16434#	16437#	16439#	16452#	16541#	
16553#	16555#	16557#	16565#	16567#	16569#	16598#	16600#	16603#	16605#	16859#	16861#	17262#	
17270#	17284#	17327#	17402#	17422#	17430#	17440#	17442#						
	720#	1268#	1269	1271#	1272	1313#	1314	1344#	1345	1347#	1348	1357#	1358
	1366#	1367	1373#	1374	1380#	1381	1387#	1388	1394#	1395	1401#	1402	1404#
	1405	1414#	1415	1425#	1426	1449#	1450	1456#	1457	1463#	1464	1470#	1471
	1477#	1478	1484#	1485	1533#	1534	1564#	1565	1567#	1568	1577#	1578	1586#
	1587	1593#	1594	1600#	1601	1607#	1608	1614#	1615	1621#	1622	1630#	1631
	1652#	1653	1659#	1660	1666#	1667	1673#	1674	1680#	1681	1687#	1688	1694#
	1695	1761#	1762	1781#	1782	1807#	1808	1815#	1816	1831#	1832	1839#	1840
	1853#	1854	1881#	1882	1888#	1889	1895#	1896	1902#	1903	1909#	1910	1916#
	1917	1923#	1924	1941#	1942	1947#	1948	1949#	1950	1952#	1953	1970#	1971
	1972#	1973	1975#	1976	1995#	1996	2006#	2007	2014#	2015	2017#	2018	2037#
	2038	2046#	2047	2057#	2058	2099#	2100	2101#	2102	2110#	2111	2117#	2118
	2124#	2125	2131#	2132	2138#	2139	2145#	2146	2152#	2153	2159#	2160	2173#
	2174	2182#	2183	2377#	2378	2397#	2398	2423#	2424	2431#	2432	2447#	2448
	2455#	2456	2487#	2488	2494#	2495	2501#	2502	2508#	2509	2515#	2516	2522#
	2523	2529#	2530	2545#	2546	2555#	2556	2557#	2558	2560#	2561	2577#	2578
	2590#	2591	2604#	2605	2606#	2607	2609#	2610	2621#	2622	2629#	2630	2641#
	2642	2652#	2653	2694#	2695	2696#	2697	2705#	2706	2712#	2713	2719#	2720

SSLOC = 057504

2726#	2727	2733#	2734	2740#	2741	2747#	2748	2754#	2755	2767#	2768	2776#
2777	2892#	2893	2918#	2919	2926#	2927	2942#	2943	2950#	2951	3008#	3009
3015#	3016	3022#	3023	3029#	3030	3036#	3037	3043#	3044	3050#	3051	3072#
3073	3078#	3079	3080#	3081	3083#	3084	3101#	3102	3103#	3104	3106#	3107
3127#	3128	3138#	3139	3144#	3145	3156#	3157	3165#	3166	3176#	3177	3220#
3221	3222#	3223	3231#	3232	3238#	3239	3245#	3246	3252#	3253	3259#	3260
3266#	3267	3273#	3274	3280#	3281	3294#	3295	3303#	3304	3396#	3397	3416#
3417	3474#	3475	3481#	3482	3488#	3489	3495#	3496	3502#	3503	3509#	3510
3516#	3517	3528#	3529	3561#	3562	3570#	3571	3577#	3578	3584#	3585	3591#
3592	3598#	3599	3605#	3606	3682#	3683	3702#	3703	3760#	3761	3767#	3768
3774#	3775	3781#	3782	3788#	3789	3795#	3796	3802#	3803	3814#	3815	3847#
3848	3857#	3858	3864#	3865	3871#	3872	3878#	3879	3885#	3886	3892#	3893
3966#	3967	3982#	3983	4025#	4026	4056#	4057	4063#	4064	4070#	4071	4077#
4078	4084#	4085	4091#	4092	4098#	4099	4110#	4111	4135#	4136	4159#	4160
4168#	4169	4175#	4176	4182#	4183	4189#	4190	4196#	4197	4203#	4204	4279#
4280	4295#	4296	4338#	4339	4366#	4367	4373#	4374	4380#	4381	4387#	4388
4394#	4395	4401#	4402	4408#	4409	4420#	4421	4445#	4446	4466#	4467	4475#
4476	4482#	4483	4489#	4490	4496#	4497	4503#	4504	4510#	4511	4590#	4591
4615#	4616	4656#	4657	4666#	4667	4676#	4677	4678#	4679	4701#	4702	4708#
4709	4715#	4716	4722#	4723	4729#	4730	4736#	4737	4743#	4744	4755#	4756
4795#	4796	4802#	4803	4809#	4810	4816#	4817	4823#	4824	4830#	4831	4837#
4838	4934#	4935	4963#	4964	4975#	4976	4999#	5000	5005#	5006	5048#	5049
5055#	5056	5062#	5063	5069#	5070	5076#	5077	5083#	5084	5090#	5091	5102#
5103	5129#	5130	5143#	5144	5150#	5151	5157#	5158	5164#	5165	5171#	5172
5178#	5179	5185#	5186	5192#	5193	5239#	5240	5242#	5243	5359#	5360	5389#
5390	5393#	5394	5409#	5410	5416#	5417	5423#	5424	5430#	5431	5437#	5438
5444#	5445	5451#	5452	5476#	5477	5478#	5479	5481#	5482	5501#	5502	5512#
5513	5516#	5517	5527#	5528	5536#	5537	5571#	5572	5626#	5627	5679#	5680
5682#	5683	5749#	5750	5756#	5757	5765#	5766	5768#	5769	5785#	5786	5796#
5797	5811#	5812	5833#	5834	5836#	5837	5848#	5849	5871#	5872	5880#	5881
5915#	5916	5960#	5961	5967#	5968	5974#	5975	5981#	5982	5988#	5989	5995#
5996	6002#	6003	6022#	6023	6024#	6025	6027#	6028	6042#	6043	6068#	6069
6075#	6076	6082#	6083	6089#	6090	6096#	6097	6103#	6104	6110#	6111	6119#
6120	6128#	6129	6135#	6136	6138#	6139	6149#	6150	6158#	6159	6281#	6282
6286#	6287	6293#	6294	6296#	6297	6312#	6313	6315#	6316	6333#	6334	6335#
6336	6343#	6344	6357#	6358	6359#	6360	6368#	6369	6377#	6378	6387#	6388
6404#	6405	6407#	6408	6409#	6410	6422#	6423	6426#	6427	6441#	6442	6456#
6457	6460#	6461	6468#	6469	6499#	6500	6544#	6545	6551#	6552	6558#	6559
6565#	6566	6572#	6573	6579#	6580	6586#	6587	6606#	6607	6608#	6609	6611#
6612	6626#	6627	6652#	6653	6659#	6660	6666#	6667	6673#	6674	6680#	6681
6687#	6688	6694#	6695	6703#	6704	6714#	6715	6723#	6724	6831#	6832	6867#
6868	6886#	6887	6898#	6899	6943#	6944	6951#	6952	6961#	6962	6970#	6971
7014#	7015	7021#	7022	7028#	7029	7035#	7036	7042#	7043	7049#	7050	7056#
7057	7069#	7070	7094#	7095	7101#	7102	7108#	7109	7115#	7116	7122#	7123
7129#	7130	7136#	7137	7143#	7144	7150#	7151	7256#	7257	7292#	7293	7316#
7317	7325#	7326	7330#	7331	7339#	7340	7352#	7353	7356#	7357	7380#	7381
7385#	7386	7399#	7400	7409#	7410	7413#	7414	7443#	7444	7450#	7451	7490#
7491	7497#	7498	7504#	7505	7511#	7512	7518#	7519	7525#	7526	7532#	7533
7545#	7546	7570#	7571	7581#	7582	7593#	7594	7601#	7602	7608#	7609	8007#
8008	8066#	8067	8075#	8076	8096#	8097	8153#	8154	8166#	8167	8192#	8193
8206#	8207	8208#	8209	8224#	8225	8238#	8239	8281#	8282	8296#	8297	8308#
8309	8320#	8321	8325#	8326	8331#	8332	8349#	8350	8352#	8353	8368#	8369
8371#	8372	8373#	8374	8380#	8381	8383#	8384	8385#	8386	8398#	8399	8413#
8414	8416#	8417	8422#	8423	8437#	8438	8443#	8444	8459#	8460	8467#	8468
8509#	8510	8516#	8517	8523#	8524	8530#	8531	8537#	8538	8544#	8545	8551#
8552	8567#	8568	8569#	8570	8572#	8573	8593#	8594	8618#	8619	8629#	8630

8641#	8642	8649#	8650	8656#	8657	8663#	8664	8670#	8671	8856#	8857	8859#
8860	8864#	8865	8883#	8884	8886#	8887	8900#	8901	8903#	8904	8910#	8911
8926#	8927	8929#	8930	8934#	8935	8950#	8951	8953#	8954	8970#	8971	8973#
8974	9017#	9018	9027#	9028	9042#	9043	9055#	9056	9057#	9058	9060#	9061
9062#	9063	9079#	9080	9093#	9094	9107#	9108	9143#	9144	9165#	9166	9178#
9179	9238#	9239	9245#	9246	9252#	9253	9259#	9260	9266#	9267	9273#	9274
9280#	9281	9296#	9297	9298#	9299	9301#	9302	9316#	9317	9342#	9343	9353#
9354	9365#	9366	9373#	9374	9380#	9381	9387#	9388	9553#	9554	9563#	9564
9565#	9566	9568#	9569	9614#	9615	9621#	9622	9638#	9639	9652#	9653	9654#
9655	9673#	9674	9687#	9688	9689#	9690	9708#	9709	9722#	9723	9724#	9725
9743#	9744	9757#	9758	9759#	9760	9778#	9779	9792#	9793	9794#	9795	9813#
9814	9826#	9827	9844#	9845	9862#	9863	9880#	9881	9902#	9903	9958#	9959
9980#	9981	10002#	10003	10024#	10025	10046#	10047	10068#	10069	10088#	10089	10108#
10109	10128#	10129	10153#	10154	10160#	10161	10211#	10212	10214#	10215	10297#	10298
10326#	10327	10355#	10356	10419#	10420	10443#	10444	10460#	10461	10483#	10484	10501#
10502	10521#	10522	10545#	10546	10566#	10567	10590#	10591	10607#	10608	10630#	10631
10648#	10649	10667#	10668	10691#	10692	10712#	10713	10736#	10737	10753#	10754	10756#
10757	10779#	10780	10786#	10787	10804#	10805	10811#	10812	10832#	10833	10856#	10857
10911#	10912	10918#	10919	10925#	10926	10932#	10933	10939#	10940	10946#	10947	10953#
10954	10972#	10973	10974#	10975	10977#	10978	11007#	11008	11087#	11088	11103#	11104
11115#	11116	11135#	11136	11137#	11138	11140#	11141	11222#	11223	11251#	11252	11280#
11281	11344#	11345	11368#	11369	11385#	11386	11408#	11409	11426#	11427	11446#	11447
11470#	11471	11491#	11492	11517#	11518	11534#	11535	11557#	11558	11575#	11576	11594#
11595	11618#	11619	11639#	11640	11663#	11664	11680#	11681	11683#	11684	11706#	11707
11713#	11714	11731#	11732	11738#	11739	11759#	11760	11783#	11784	11838#	11839	11845#
11846	11852#	11853	11859#	11860	11866#	11867	11873#	11874	11880#	11881	11900#	11901
11902#	11903	11905#	11906	11935#	11936	12015#	12016	12031#	12032	12043#	12044	12063#
12064	12065#	12066	12068#	12069	12148#	12149	12183#	12184	12185#	12186	12193#	12194
12195#	12196	12203#	12204	12205#	12206	12216#	12217	12218#	12219	12221#	12222	12224#
12225	12236#	12237	12238#	12239	12246#	12247	12248#	12249	12257#	12258	12272#	12273
12274#	12275	12282#	12283	12285#	12286	12287#	12288	12295#	12296	12297#	12298	12333#
12334	12349#	12350	12355#	12356	12373#	12374	12376#	12377	12382#	12383	12402#	12403
12405#	12406	12410#	12411	12430#	12431	12436#	12437	12450#	12451	12455#	12456	12479#
12480	12495#	12496	12501#	12502	12519#	12520	12525#	12526	12543#	12544	12548#	12549
12566#	12567	12572#	12573	12586#	12587	12591#	12592	12640#	12641	12647#	12648	12654#
12655	12661#	12662	12668#	12669	12675#	12676	12682#	12683	12692#	12693	12702#	12703
12715#	12716	12750#	12751	12761#	12762	12773#	12774	12849#	12850	12868#	12869	12880#
12881	12882#	12883	12892#	12893	12914#	12915	12926#	12927	12939#	12940	12942#	12943
12956#	12957	12964#	12965	12966#	12967	12971#	12972	12973#	12974	12987#	12988	12989#
12990	13001#	13002	13007#	13008	13014#	13015	13056#	13057	13058#	13059	13074#	13075
13110#	13111	13119#	13120	13132#	13133	13145#	13146	13150#	13151	13156#	13157	13172#
13173	13182#	13183	13195#	13196	13197#	13198	13199#	13200	13210#	13211	13215#	13216
13225#	13226	13237#	13238	13240#	13241	13243#	13244	13245#	13246	13255#	13256	13269#
13270	13279#	13280	13292#	13293	13294#	13295	13296#	13297	13313#	13314	13316#	13317
13319#	13320	13333#	13334	13340#	13341	13353#	13354	13355#	13356	13357#	13358	13374#
13375	13383#	13384	13403#	13404	13416#	13417	13418#	13419	13420#	13421	13427#	13428
13436#	13437	13452#	13453	13466#	13467	13479#	13480	13481#	13482	13483#	13484	13490#
13491	13529#	13530	13540#	13541	13551#	13552	13558#	13559	13575#	13576	13577#	13578
13580#	13581	13592#	13593	13621#	13622	13631#	13632	13683#	13684	13694#	13695	13706#
13707	13717#	13718	13724#	13725	13734#	13735	13736#	13737	13739#	13740	13801#	13802
13804#	13805	13960#	13961	13967#	13968	13974#	13975	13981#	13982	13988#	13989	13995#
13996	14002#	14003	14021#	14022	14023#	14024	14026#	14027	14041#	14042	14128#	14129
14144#	14145	14156#	14157	14185#	14186	14187#	14188	14190#	14191	14346#	14347	14353#
14354	14360#	14361	14367#	14368	14374#	14375	14381#	14382	14388#	14389	14408#	14409
14410#	14411	14413#	14414	14428#	14429	14516#	14517	14532#	14533	14544#	14545	14573#
14574	14575#	14576	14578#	14579	14689#	14690	14691#	14692	14699#	14700	14701#	14702

14709#	14710	14711#	14712	14722#	14723	14724#	14725	14727#	14728	14730#	14731	14742#
14743	14744#	14745	14752#	14753	14754#	14755	14763#	14764	14778#	14779	14780#	14781
14788#	14789	14791#	14792	14793#	14794	14801#	14802	14803#	14804	14843#	14844	14846#
14847	14861#	14862	14871#	14872	14891#	14892	14915#	14916	14918#	14919	14933#	14934
14943#	14944	14963#	14964	15009#	15010	15020#	15021	15032#	15033	15045#	15046	15081#
15082	15092#	15093	15104#	15105	15179#	15180	15182#	15183	15197#	15198	15207#	15208
15217#	15218	15240#	15241	15252#	15253	15254#	15255	15272#	15273	15290#	15291	15298#
15299	15300#	15301	15302#	15303	15321#	15322	15324#	15325	15348#	15349	15366#	15367
15368#	15369	15371#	15372	15376#	15377	15378#	15379	15401#	15402	15403#	15404	15412#
15413	15414#	15415	15416#	15417	15423#	15424	15429#	15430	15436#	15437	15466#	15467
15468#	15469	15471#	15472	15528#	15529	15537#	15538	15547#	15548	15558#	15559	15565#
15566	15583#	15584	15585#	15586	15588#	15589	15600#	15601	15610#	15611	15620#	15621
15656#	15657	15667#	15668	15679#	15680	15690#	15691	15697#	15698	15707#	15708	15709#
15710	15712#	15713	15773#	15774	15776#	15777	15933#	15934	15940#	15941	15947#	15948
15954#	15955	15961#	15962	15968#	15969	15975#	15976	15995#	15996	15997#	15998	16000#
16001	16015#	16016	16100#	16101	16116#	16117	16128#	16129	16157#	16158	16159#	16160
16162#	16163	16206#	16207	16209#	16210	16371#	16372	16378#	16379	16385#	16386	16392#
16393	16399#	16400	16406#	16407	16413#	16414	16433#	16434	16435#	16436	16438#	16439
16453#	16454	16542#	16543	16558#	16559	16570#	16571	16599#	16600	16601#	16602	16604#
16605	16860#	16861	16862#	16863	17023#	17024	17060#	17061	17097#	17098	17132#	17133
17170#	17171	17263#	17264	17271#	17272	17285#	17286	17328#	17329	17403#	17404	17423#
17424	17431#	17432	17441#	17442								
720#	9534#	9920	9936#	10172	16978#	16993	17007#	17031	17044#	17068	17081#	17105
17116#	17140	17154#	17178	17190#	17205	17217#	17232	17246#	17344	17389#	17468	
720#												
720#	9534#	9919	9936#	10171	16978#	16992	17007#	17030	17044#	17067	17081#	17104
17116#	17139	17154#	17177	17190#	17204	17217#	17231	17246#	17343	17389#	17467	
720#	9534#	9918	9936#	10170	16978#	16991	17007#	17029	17044#	17066	17081#	17103
17116#	17138	17154#	17176	17190#	17203	17217#	17230	17246#	17342	17389#	17466	
720#	9534#	9919	9936#	10171	16978#	16992	17007#	17030	17044#	17067	17081#	17104
17116#	17139	17154#	17177	17190#	17204	17217#	17231	17246#	17343	17389#	17467	
720#												
720#	10448#	10556#	10557	10558	10595#	10702#	10703	10704	10741#	10867#	10868	10869
11373#	11481#	11482	11483	11522#	11629#	11630	11631	11668#	11794#	11795	11796	
720#	10455#	10456	10475#	10476	10493#	10494	10512#	10513	10534#	10535	10602#	10603
10622#	10623	10640#	10641	10658#	10659	10680#	10681	10748#	10749	10773#	10774	10798#
10799	10823#	10824	10845#	10846	11380#	11381	11400#	11401	11418#	11419	11437#	11438
11459#	11460	11529#	11530	11549#	11550	11567#	11568	11585#	11586	11607#	11608	11675#
11676	11700#	11701	11725#	11726	11750#	11751	11772#	11773				
720#												
720#	1234#	1310#	1349#	1359#	1368#	1375#	1382#	1389#	1396#	1406#	1416#	1446#
1451#	1458#	1465#	1472#	1479#	1486#	1530#	1569#	1579#	1588#	1595#	1602#	1609#
1616#	1623#	1649#	1654#	1661#	1668#	1675#	1682#	1689#	1696#	1809#	1817#	1822#
1833#	1841#	1846#	1883#	1890#	1897#	1904#	1911#	1918#	1925#	1954#	1977#	1997#
2039#	2071#	2075#	2103#	2112#	2119#	2126#	2133#	2140#	2147#	2154#	2161#	2175#
2425#	2433#	2438#	2449#	2457#	2462#	2489#	2496#	2503#	2510#	2517#	2524#	2531#
2562#	2579#	2611#	2631#	2666#	2670#	2698#	2707#	2714#	2721#	2728#	2735#	2742#
2749#	2756#	2769#	2920#	2928#	2933#	2944#	2952#	2957#	3010#	3017#	3024#	3031#
3038#	3045#	3052#	3085#	3108#	3129#	3158#	3190#	3194#	3224#	3233#	3240#	3247#
3254#	3261#	3268#	3275#	3282#	3296#	3450#	3476#	3483#	3490#	3497#	3504#	3511#
3518#	3542#	3544#	3563#	3572#	3579#	3586#	3593#	3600#	3607#	3736#	3762#	3769#
3776#	3783#	3790#	3797#	3804#	3828#	3830#	3849#	3859#	3866#	3873#	3880#	3887#
3894#	4016#	4058#	4065#	4072#	4079#	4086#	4093#	4100#	4124#	4147#	4161#	4170#
4177#	4184#	4191#	4198#	4205#	4329#	4368#	4375#	4382#	4389#	4396#	4403#	4410#
4434#	4452#	4468#	4477#	4484#	4491#	4498#	4505#	4512#	4608#	4612#	4703#	4710#
4717#	4724#	4731#	4738#	4745#	4769#	4773#	4775#	4797#	4804#	4811#	4818#	4825#

\$\$LOCN= 000000

\$\$REG = 177777

\$\$RETU= 000000

\$\$RTN1= 052407

\$\$RTN2= 052410

\$\$SRC = 000000

\$\$TGSV= 051503

\$\$TGS1= 000001

\$\$TGS2= 000000

\$\$TO = 000000

4832#	4839#	4956#	4960#	5050#	5057#	5064#	5071#	5078#	5085#	5092#	5116#	5120#
5131#	5145#	5152#	5159#	5166#	5173#	5180#	5187#	5194#	5279#	5324#	5328#	5395#
5411#	5418#	5425#	5432#	5439#	5446#	5453#	5483#	5503#	5529#	5551#	5612#	5735#
5741#	5962#	5969#	5976#	5983#	5990#	5997#	6004#	6029#	6053#	6055#	6070#	6077#
6084#	6091#	6098#	6105#	6112#	6121#	6130#	6140#	6151#	6264#	6270#	6546#	6553#
6560#	6567#	6574#	6581#	6588#	6613#	6637#	6639#	6654#	6661#	6668#	6675#	6682#
6689#	6696#	6705#	6716#	6855#	6883#	6937#	7016#	7023#	7030#	7037#	7044#	7051#
7058#	7079#	7081#	7096#	7103#	7110#	7117#	7124#	7131#	7138#	7145#	7152#	7440#
7492#	7499#	7506#	7513#	7520#	7527#	7534#	7555#	7557#	7572#	7584#	7596#	7603#
7610#	8032#	8511#	8518#	8525#	8532#	8539#	8546#	8553#	8574#	8603#	8605#	8620#
8632#	8644#	8651#	8658#	8665#	8672#	8800#	8819#	9240#	9247#	9254#	9261#	9268#
9275#	9282#	9303#	9327#	9329#	9344#	9356#	9368#	9375#	9382#	9389#	9540#	9914#
9942#	10168#	10268#	10913#	10920#	10927#	10934#	10941#	10948#	10955#	10979#	11018#	11071#
11089#	11106#	11118#	11142#	11189#	11840#	11847#	11854#	11861#	11868#	11875#	11882#	11907#
11946#	11999#	12017#	12034#	12046#	12070#	12099#	12642#	12649#	12656#	12663#	12670#	12677#
12684#	12752#	12764#	12776#	12836#	13531#	13543#	13553#	13560#	13594#	13643#	13685#	13697#
13709#	13719#	13726#	13741#	13858#	13888#	13891	13898#	13901	13908#	13911	13962#	13969#
13976#	13983#	13990#	13997#	14004#	14028#	14052#	14080#	14083	14090#	14093	14100#	14103
14111#	14130#	14147#	14159#	14192#	14239#	14273#	14276	14283#	14286	14293#	14296	14348#
14355#	14362#	14369#	14376#	14383#	14390#	14415#	14439#	14471#	14474	14481#	14484	14491#
14494	14501#	14518#	14535#	14547#	14580#	14609#	14832#	15011#	15023#	15035#	15083#	15095#
15107#	15150#	15530#	15540#	15550#	15560#	15567#	15602#	15633#	15641#	15658#	15670#	15682#
15692#	15699#	15716#	15831#	15861#	15864	15871#	15874	15881#	15884	15935#	15942#	15949#
15956#	15963#	15970#	15977#	16002#	16026#	16054#	16057	16064#	16067	16074#	16077	16084#
16102#	16119#	16131#	16164#	16264#	16298#	16301	16308#	16311	16318#	16321	16373#	16380#
16387#	16394#	16401#	16408#	16415#	16440#	16464#	16496#	16499	16506#	16509	16516#	16519
16526#	16544#	16561#	16573#	16606#								
1234#	1235	1310#	1311	1349#	1350	1359#	1360	1368#	1369	1375#	1376	1382#
1383	1389#	1390	1396#	1397	1406#	1407	1416#	1417	1446#	1447	1451#	1452
1458#	1459	1465#	1466	1472#	1473	1479#	1480	1486#	1487	1530#	1531	1569#
1570	1579#	1580	1588#	1589	1595#	1596	1602#	1603	1609#	1610	1616#	1617
1623#	1624	1649#	1650	1654#	1655	1661#	1662	1668#	1669	1675#	1676	1682#
1683	1689#	1690	1696#	1697	1809#	1810	1817#	1818	1822#	1823	1833#	1834
1841#	1842	1846#	1847	1883#	1884	1890#	1891	1897#	1898	1904#	1905	1911#
1912	1918#	1919	1925#	1926	1954#	1955	1977#	1978	1997#	1998	2039#	2040
2071#	2072	2075#	2076	2103#	2104	2112#	2113	2119#	2120	2126#	2127	2133#
2134	2140#	2141	2147#	2148	2154#	2155	2161#	2162	2175#	2176	2425#	2426
2433#	2434	2438#	2439	2449#	2450	2457#	2458	2462#	2463	2489#	2490	2496#
2497	2503#	2504	2510#	2511	2517#	2518	2524#	2525	2531#	2532	2562#	2563
2579#	2580	2611#	2612	2631#	2632	2666#	2667	2670#	2671	2698#	2699	2707#
2708	2714#	2715	2721#	2722	2728#	2729	2735#	2736	2742#	2743	2749#	2750
2756#	2757	2769#	2770	2920#	2921	2928#	2929	2933#	2934	2944#	2945	2952#
2953	2957#	2958	3010#	3011	3017#	3018	3024#	3025	3031#	3032	3038#	3039
3045#	3046	3052#	3053	3085#	3086	3108#	3109	3129#	3130	3158#	3159	3190#
3191	3194#	3195	3224#	3225	3233#	3234	3240#	3241	3247#	3248	3254#	3255
3261#	3262	3268#	3269	3275#	3276	3282#	3283	3296#	3297	3450#	3451	3476#
3477	3483#	3484	3490#	3491	3497#	3498	3504#	3505	3511#	3512	3518#	3519
3542#	3543	3544#	3545	3563#	3564	3572#	3573	3579#	3580	3586#	3587	3593#
3594	3600#	3601	3607#	3608	3736#	3737	3762#	3763	3769#	3770	3776#	3777
3783#	3784	3790#	3791	3797#	3798	3804#	3805	3828#	3829	3830#	3831	3849#
3850	3859#	3860	3866#	3867	3873#	3874	3880#	3881	3887#	3888	3894#	3895
4016#	4017	4058#	4059	4065#	4066	4072#	4073	4079#	4080	4086#	4087	4093#
4094	4100#	4101	4124#	4125	4147#	4148	4161#	4162	4170#	4171	4177#	4178
4184#	4185	4191#	4192	4198#	4199	4205#	4206	4329#	4330	4368#	4369	4375#
4376	4382#	4383	4389#	4390	4396#	4397	4403#	4404	4410#	4411	4434#	4435
4452#	4453	4468#	4469	4477#	4478	4484#	4485	4491#	4492	4498#	4499	4505#

\$\$\$TOTL= 000000

4506	4512#	4513	4608#	4609	4612#	4613	4703#	4704	4710#	4711	4717#	4718
4724#	4725	4731#	4732	4738#	4739	4745#	4746	4769#	4770	4773#	4774	4775#
4776	4797#	4798	4804#	4805	4811#	4812	4818#	4819	4825#	4826	4832#	4833
4839#	4840	4956#	4957	4960#	4961	5050#	5051	5057#	5058	5064#	5065	5071#
5072	5078#	5079	5085#	5086	5092#	5093	5116#	5117	5120#	5121	5131#	5132
5145#	5146	5152#	5153	5159#	5160	5166#	5167	5173#	5174	5180#	5181	5187#
5188	5194#	5195	5279#	5280	5324#	5325	5328#	5329	5395#	5396	5411#	5412
5418#	5419	5425#	5426	5432#	5433	5439#	5440	5446#	5447	5453#	5454	5483#
5484	5503#	5504	5529#	5530	5551#	5552	5612#	5613	5735#	5736	5741#	5742
5962#	5963	5969#	5970	5976#	5977	5983#	5984	5990#	5991	5997#	5998	6004#
6005	6029#	6030	6053#	6054	6055#	6056	6070#	6071	6077#	6078	6084#	6085
6091#	6092	6098#	6099	6105#	6106	6112#	6113	6121#	6122	6130#	6131	6140#
6141	6151#	6152	6264#	6265	6270#	6271	6546#	6547	6553#	6554	6560#	6561
6567#	6568	6574#	6575	6581#	6582	6588#	6589	6613#	6614	6637#	6638	6639#
6640	6654#	6655	6661#	6662	6668#	6669	6675#	6676	6682#	6683	6689#	6690
6696#	6697	6705#	6706	6716#	6717	6855#	6856	6883#	6884	6937#	6938	7016#
7017	7023#	7024	7030#	7031	7037#	7038	7044#	7045	7051#	7052	7058#	7059
7079#	7080	7081#	7082	7096#	7097	7103#	7104	7110#	7111	7117#	7118	7124#
7125	7131#	7132	7138#	7139	7145#	7146	7152#	7153	7440#	7441	7492#	7493
7499#	7500	7506#	7507	7513#	7514	7520#	7521	7527#	7528	7534#	7535	7555#
7556	7557#	7558	7572#	7573	7584#	7585	7596#	7597	7603#	7604	7610#	7611
8032#	8033	8511#	8512	8518#	8519	8525#	8526	8532#	8533	8539#	8540	8546#
8547	8553#	8554	8574#	8575	8603#	8604	8605#	8606	8620#	8621	8632#	8633
8644#	8645	8651#	8652	8658#	8659	8665#	8666	8672#	8673	8800#	8801	8819#
8820	9240#	9241	9247#	9248	9254#	9255	9261#	9262	9268#	9269	9275#	9276
9282#	9283	9303#	9304	9327#	9328	9329#	9330	9344#	9345	9356#	9357	9368#
9369	9375#	9376	9382#	9383	9389#	9390	9540#	9541	9914#	9915	9942#	9943
10168#	10169	10268#	10269	10913#	10914	10920#	10921	10927#	10928	10934#	10935	10941#
10942	10948#	10949	10955#	10956	10979#	10980	11018#	11019	11071#	11072	11089#	11090
11106#	11107	11118#	11119	11142#	11143	11189#	11190	11840#	11841	11847#	11848	11854#
11855	11861#	11862	11868#	11869	11875#	11876	11882#	11883	11907#	11908	11946#	11947
11999#	12000	12017#	12018	12034#	12035	12046#	12047	12070#	12071	12099#	12100	12642#
12643	12649#	12650	12656#	12657	12663#	12664	12670#	12671	12677#	12678	12684#	12685
12752#	12753	12764#	12765	12776#	12777	12836#	12837	13531#	13532	13543#	13544	13553#
13554	13560#	13561	13594#	13595	13643#	13644	13685#	13686	13697#	13698	13709#	13710
13719#	13720	13726#	13727	13741#	13742	13858#	13859	13888#	13890	13895	13898#	13900
13905	13908#	13910	13915	13962#	13963	13969#	13970	13976#	13977	13983#	13984	13990#
13991	13997#	13998	14004#	14005	14028#	14029	14052#	14053	14080#	14082	14087	14090#
14092	14097	14100#	14102	14107	14111#	14112	14130#	14131	14147#	14148	14159#	14160
14192#	14193	14239#	14240	14273#	14275	14280	14283#	14285	14290	14293#	14295	14300
14348#	14349	14355#	14356	14362#	14363	14369#	14370	14376#	14377	14383#	14384	14390#
14391	14415#	14416	14439#	14440	14471#	14473	14478	14481#	14483	14488	14491#	14493
14498	14501#	14502	14518#	14519	14535#	14536	14547#	14548	14580#	14581	14609#	14610
14832#	14833	15011#	15012	15023#	15024	15035#	15036	15083#	15084	15095#	15096	15107#
15108	15150#	15151	15530#	15531	15540#	15541	15550#	15551	15560#	15561	15567#	15568
15602#	15603	15633#	15634	15641#	15642	15658#	15659	15670#	15671	15682#	15683	15692#
15693	15699#	15700	15716#	15717	15831#	15832	15861#	15863	15868	15871#	15873	15878
15881#	15883	15888	15935#	15936	15942#	15943	15949#	15950	15956#	15957	15963#	15964
15970#	15971	15977#	15978	16002#	16003	16026#	16027	16054#	16056	16061	16064#	16066
16071	16074#	16076	16081	16084#	16085	16102#	16103	16119#	16120	16131#	16132	16164#
16165	16264#	16265	16298#	16300	16305	16308#	16310	16315	16318#	16320	16325	16373#
16374	16380#	16381	16387#	16388	16394#	16395	16401#	16402	16408#	16409	16415#	16416
16440#	16441	16464#	16465	16496#	16498	16503	16506#	16508	16513	16516#	16518	16523
16526#	16527	16544#	16545	16561#	16562	16573#	16574	16606#	16607			
720#												
17737*	17741*	17751	17787#									

\$\$\$TAG- 050000
 \$OFILL 061105

\$50000	001570	1268	1277#
\$50001	001570	1271	1275#
\$50002	001626	1302#	1425
\$50003	001654	1313	1322#
\$50004	001664	1321	1330#
\$50005	001732	1344	1354#
\$50006	001730	1347	1351#
\$50007	001746	1353	1363#
\$50010	001746	1357	1361#
\$50011	001760	1366	1370#
\$50012	001772	1373	1377#
\$50013	002004	1380	1384#
\$50014	002016	1387	1391#
\$50015	002026	1394	1398#
\$50016	002050	1401	1411#
\$50017	002046	1404	1408#
\$50020	002062	1410	1420#
\$50021	002062	1414	1418#
\$50022	002130	1449	1453#
\$50023	002142	1456	1460#
\$50024	002154	1463	1467#
\$50025	002166	1470	1474#
\$50026	002176	1477	1481#
\$50027	002206	1484	1488#
\$50030	002252	1522#	1630
\$50031	002300	1533	1542#
\$50032	002310	1541	1550#
\$50033	002356	1564	1574#
\$50034	002354	1567	1571#
\$50035	002372	1573	1583#
\$50036	002372	1577	1581#
\$50037	002404	1586	1590#
\$50040	002416	1593	1597#
\$50041	002430	1600	1604#
\$50042	002442	1607	1611#
\$50043	002454	1614	1618#
\$50044	002466	1621	1625#
\$50045	002532	1652	1656#
\$50046	002544	1659	1663#
\$50047	002556	1666	1670#
\$50050	002570	1673	1677#
\$50051	002602	1680	1684#
\$50052	002614	1687	1691#
\$50053	002626	1694	1698#
\$50054	002760	1761	1766#
\$50055	002762	1765	1770#
\$50056	003020	1779	1782#
\$50057	003026	1781	1787#
\$50060	003036	1786	1792#
\$50061	003102	1807	1812#
\$50062	003124	1811	1826#
\$50063	003120	1815	1820#
\$50064	003124	1819	1824#
\$50065	003144	1831	1836#
\$50066	003166	1835	1850#
\$50067	003162	1839	1844#

\$50070	003166	1843	1848#	
\$50071	003202	1853	1858#	
\$50072	003206	1857	1862#	
\$50073	003240	1881	1885#	
\$50074	003252	1888	1892#	
\$50075	003264	1895	1899#	
\$50076	003276	1902	1906#	
\$50077	003310	1909	1913#	
\$50100	003322	1916	1920#	
\$50101	003334	1923	1927#	
\$50102	003416	1941	1964#	
\$50103	003364	1945#	1960	
\$50104	003414	1947	1949	1961#
\$50105	003410	1952	1956#	
\$50106	003520	1963	2009#	
\$50107	003422	1968#	1983	
\$50110	003452	1970	1972	1984#
\$50111	003446	1975	1979#	
\$50112	003472	1992#	2006	
\$50113	003504	1995	1999#	
\$50114	003520	2004	2007#	
\$50115	003634	2014	2049#	
\$50116	003554	2017	2022#	
\$50117	003562	2021	2026#	
\$50120	003606	2034#	2046	
\$50121	003620	2037	2041#	
\$50122	003634	2044	2047#	
\$50123	003660	2057	2061#	
\$50124	003760	2099	2107#	
\$50125	003760	2101	2105#	
\$50126	003772	2110	2114#	
\$50127	004C 4	2117	2121#	
\$50130	004016	2124	2128#	
\$50131	004030	2131	2135#	
\$50132	004042	2138	2142#	
\$50133	004054	2145	2149#	
\$50134	004070	2152	2156#	
\$50135	004104	2159	2163#	
\$50136	004124	2170#	2182	
\$50137	004134	2173	2177#	
\$50140	004150	2180	2183#	
\$50141	004636	2377	2382#	
\$50142	004640	2381	2386#	
\$50143	004676	2395	2398#	
\$50144	004704	2397	2403#	
\$50145	004714	2402	2408#	
\$50146	004762	2423	2428#	
\$50147	005004	2427	2442#	
\$50150	005000	2431	2436#	
\$50151	005004	2435	2440#	
\$50152	005026	2447	2452#	
\$50153	005050	2451	2466#	
\$50154	005044	2455	2460#	
\$50155	005050	2459	2464#	
\$50156	005110	2487	2491#	
\$50157	005122	2494	2498#	

\$50160	005134	2501	2505#	
\$50161	005146	2508	2512#	
\$50162	005160	2515	2519#	
\$50163	005172	2522	2526#	
\$50164	005204	2529	2533#	
\$50165	005274	2545	2572#	
\$50166	005244	2553#	2568	
\$50167	005272	2555	2557	2569#
\$50170	005266	2560	2564#	
\$50171	005426	2571	2618#	
\$50172	005274	2574#	2590	
\$50173	005306	2577	2581#	
\$50174	005344	2588	2591#	
\$50175	005400	2602#	2615	
\$50176	005426	2604	2606	2616#
\$50177	005424	2609	2613#	
\$50200	005510	2621	2644#	
\$50201	005446	2626#	2641	
\$50202	005460	2629	2633#	
\$50203	005510	2639	2642#	
\$50204	005534	2652	2656#	
\$50205	005634	2694	2702#	
\$50206	005634	2696	2700#	
\$50207	005646	2705	2709#	
\$50210	005660	2712	2716#	
\$50211	005672	2719	2723#	
\$50212	005704	2726	2730#	
\$50213	005716	2733	2737#	
\$50214	005730	2740	2744#	
\$50215	005744	2747	2751#	
\$50216	005760	2754	2758#	
\$50217	005770	2764#	2776	
\$50220	006000	2767	2771#	
\$50221	006014	2774	2777#	
\$50222	006274	2890	2893#	
\$50223	006302	2892	2898#	
\$50224	006312	2897	2903#	
\$50225	006360	2918	2923#	
\$50226	006402	2922	2937#	
\$50227	006376	2926	2931#	
\$50230	006402	2930	2935#	
\$50231	006424	2942	2947#	
\$50232	006446	2946	2961#	
\$50233	006442	2950	2955#	
\$50234	006446	2954	2959#	
\$50235	006470	2971	2974#	
\$50236	006464	2972#	2980	
\$50237	006504	2976	2981#	
\$50240	006562	3008	3012#	
\$50241	006574	3015	3019#	
\$50242	006606	3022	3026#	
\$50243	006620	3029	3033#	
\$50244	006632	3036	3040#	
\$50245	006644	3043	3047#	
\$50246	006656	3050	3054#	
\$50247	006746	3072	3095#	

\$50250	006716	3076#	3091	
\$50251	006744	3078	3080	3092#
\$50252	006740	3083	3087#	
\$50253	007046	3094	3141#	
\$50254	006752	3099#	3115	
\$50255	007000	3101	3103	3116#
\$50256	006774	3106	3111#	
\$50257	007020	3124#	3138	
\$50260	007032	3127	3131#	
\$50261	007046	3136	3139#	
\$50262	007134	3144	3168#	
\$50263	007106	3153#	3165	
\$50264	007120	3156	3160#	
\$50265	007134	3163	3166#	
\$50266	007160	3176	3180#	
\$50267	007264	3220	3228#	
\$50270	007264	3222	3226#	
\$50271	007276	3231	3235#	
\$50272	007310	3238	3242#	
\$50273	007322	3245	3249#	
\$50274	007334	3252	3256#	
\$50275	007346	3259	3263#	
\$50276	007360	3266	3270#	
\$50277	007374	3273	3277#	
\$5030	002656	1720#	2059	
\$50300	007410	3280	3284#	
\$50301	007424	3291#	3303	
\$50302	007436	3294	3298#	
\$50303	007452	3301	3304#	
\$50304	007660	3396	3405#	
\$50305	007674	3404	3413#	
\$50306	007726	3416	3424#	
\$50307	007756	3423	3432#	
\$5031	004534	2337#	2654	
\$50310	010104	3474	3478#	
\$50311	010116	3481	3485#	
\$50312	010130	3488	3492#	
\$50313	010142	3495	3499#	
\$50314	010154	3502	3506#	
\$50315	010166	3509	3513#	
\$50316	010200	3516	3520#	
\$50317	010224	3528	3532#	
\$5032	006146	2841#	3178	
\$50320	010314	3561	3565#	
\$50321	010334	3570	3574#	
\$50322	010346	3577	3581#	
\$50323	010356	3584	3588#	
\$50324	010366	3591	3595#	
\$50325	010376	3598	3602#	
\$50326	010406	3605	3609#	
\$50327	010542	3682	3691#	
\$50330	010556	3690	3699#	
\$50331	010610	3702	3710#	
\$50332	010640	3709	3718#	
\$50333	010766	3760	3764#	
\$50334	011000	3767	3771#	

\$50335	011012	3774	3778#
\$50336	011024	3781	3785#
\$50337	011036	3788	3792#
\$50340	011050	3795	3799#
\$50341	011062	3802	3806#
\$50342	011106	3814	3818#
\$50343	011176	3847	3851#
\$50344	011224	3857	3861#
\$50345	011236	3864	3868#
\$50346	011246	3871	3875#
\$50347	011256	3878	3882#
\$50350	011266	3885	3889#
\$50351	011276	3892	3896#
\$50352	011422	3966	3973#
\$50353	011432	3972	3979#
\$50354	011464	3982	3990#
\$50355	011514	3989	3998#
\$50356	011610	4023#	4029
\$50357	011622	4025	4030#
\$50360	011702	4056	4060#
\$50361	011714	4063	4067#
\$50362	011726	4070	4074#
\$50363	011740	4077	4081#
\$50364	011752	4084	4088#
\$50365	011764	4091	4095#
\$50366	011776	4098	4102#
\$50367	012022	4110	4114#
\$50370	012062	4133#	4139
\$50371	012074	4135	4140#
\$50372	012152	4159	4163#
\$50373	012172	4168	4172#
\$50374	012204	4175	4179#
\$50375	012214	4182	4186#
\$50376	012224	4189	4193#
\$50377	012234	4196	4200#
\$5040	007604	3368#	3530
\$50400	012244	4203	4207#
\$50401	012374	4279	4286#
\$50402	012404	4285	4292#
\$50403	012436	4295	4303#
\$50404	012466	4302	4311#
\$50405	012562	4336#	4342
\$50406	012574	4338	4343#
\$50407	012640	4366	4370#
\$5041	010466	3652#	3816
\$50410	012652	4373	4377#
\$50411	012664	4380	4384#
\$50412	012676	4387	4391#
\$50413	012710	4394	4398#
\$50414	012722	4401	4405#
\$50415	012734	4408	4412#
\$50416	012760	4420	4424#
\$50417	013020	4443#	4449
\$5042	011350	3936#	4112
\$50420	013032	4445	4450#
\$50421	013074	4466	4470#

\$50422	013114	4475	4479#	
\$50423	013126	4482	4486#	
\$50424	013136	4489	4493#	
\$50425	013146	4496	4500#	
\$50426	013156	4503	4507#	
\$50427	013166	4510	4514#	
\$5043	012322	4249#	4422	
\$50430	013332	4590	4595#	
\$50431	013336	4594	4599#	
\$50432	013416	4615	4623#	
\$50433	013440	4622	4630#	
\$50434	013550	4656	4663#	
\$50435	013574	4666	4673#	
\$50436	013622	4676	4678	4682#
\$50437	013654	4701	4705#	
\$5044	013244	4556#	4757	
\$50440	013666	4708	4712#	
\$50441	013700	4715	4719#	
\$50442	013712	4722	4726#	
\$50443	013724	4729	4733#	
\$50444	013736	4736	4740#	
\$50445	013750	4743	4747#	
\$50446	013774	4755	4759#	
\$50447	014106	4795	4799#	
\$5045	014310	4906#	5104	
\$50450	014116	4802	4806#	
\$50451	014126	4809	4813#	
\$50452	014136	4816	4820#	
\$50453	014146	4823	4827#	
\$50454	014156	4830	4834#	
\$50455	014166	4837	4841#	
\$50456	014362	4934	4941#	
\$50457	014372	4940	4947#	
\$50460	014460	4963	4979#	
\$50461	014432	4967#	4975	
\$50462	014602	4999	5023#	
\$50463	014556	5005	5013#	
\$50464	014602	5012	5021#	
\$50465	014654	5048	5052#	
\$50466	014666	5055	5059#	
\$50467	014700	5062	5066#	
\$50470	014712	5069	5073#	
\$50471	014724	5076	5080#	
\$50472	014736	5083	5087#	
\$50473	014750	5090	5094#	
\$50474	014774	5102	5106#	
\$50475	015024	5124#	5129	
\$50476	015070	5143	5147#	
\$50477	015102	5150	5154#	
\$5050	032570	10266#	11009	
\$50500	015114	5157	5161#	
\$50501	015124	5164	5168#	
\$50502	015134	5171	5175#	
\$50503	015144	5178	5182#	
\$50504	015154	5185	5189#	
\$50505	015164	5192	5196#	

\$50506	015246	5239	5248#
\$50507	015246	5242	5246#
\$5051	035366	11187#	11937
\$50510	015334	5281#	5389
\$50511	015516	5336	5339#
\$50512	015512	5337#	5345
\$50513	015532	5341	5346#
\$50514	015556	5357#	5361
\$50515	015570	5359	5362#
\$50516	015654	5387	5390#
\$50517	015672	5393	5398#
\$5052	040154	12097#	12717
\$50520	016224	5397	5541#
\$50521	015722	5409	5413#
\$50522	015734	5416	5420#
\$50523	015746	5423	5427#
\$50524	015760	5430	5434#
\$50525	015772	5437	5441#
\$50526	016004	5444	5448#
\$50527	016016	5451	5455#
\$5053	023072	7233#	7547
\$50530	016046	5474#	5489
\$50531	016074	5476	5478
\$50532	016070	5481	5485#
\$50533	016114	5498#	5512
\$50534	016126	5501	5505#
\$50535	016142	5510	5513#
\$50536	016224	5516	5539#
\$50537	016176	5524#	5536
\$5054	025124	7980#	8595
\$50540	016210	5527	5531#
\$50541	016224	5534	5537#
\$50542	016316	5571	5576#
\$50543	016324	5575	5580#
\$50544	016430	5626	5631#
\$50545	016510	5679	5688#
\$50546	016510	5682	5686#
\$50547	016700	5749	5762#
\$5055	027210	8779#	9318
\$50550	016700	5756	5760#
\$50551	017304	5765	5899#
\$50552	016752	5768	5782#
\$50553	017304	5781	5897#
\$50554	017002	5785	5793#
\$50555	017046	5796	5808#
\$50556	017132	5811	5829#
\$50557	017224	5833	5864#
\$5056	042232	12817#	13633
\$50560	017160	5836	5842#
\$50561	017162	5846#	5861
\$50562	017224	5848	5862#
\$50563	017252	5871	5877#
\$50564	017304	5876	5895#
\$50565	017270	5880	5886#
\$50566	017304	5885	5893#
\$50567	017346	5915	5922#

5490#

\$5057	020214	6244#	6628	
\$50570	017352	5921	5928#	
\$50571	017434	5960	5964#	
\$50572	017446	5967	5971#	
\$50573	017460	5974	5978#	
\$50574	017472	5981	5985#	
\$50575	017504	5988	5992#	
\$50576	017516	5995	5999#	
\$50577	017530	6002	6006#	
\$50600	017544	6020#	6033	
\$50601	017572	6022	6024	6034#
\$50602	017570	6027	6031#	
\$50603	017616	6042	6046#	
\$50604	017662	6068	6072#	
\$50605	017672	6075	6079#	
\$50606	017702	6082	6086#	
\$50607	017712	6089	6093#	
\$50610	017722	6096	6100#	
\$50611	017732	6103	6107#	
\$50612	017742	6110	6114#	
\$50613	017760	6119	6123#	
\$50614	017760	6125#	6135	
\$50615	017772	6128	6132#	
\$50616	020012	6138	6142#	
\$50617	020016	6146#	6158	
\$50620	020026	6149	6153#	
\$50621	020042	6156	6159#	
\$50622	020444	6281	6330#	
\$50623	020340	6286	6290#	
\$50624	020402	6293	6309#	
\$50625	020366	6296	6301#	
\$50626	020402	6300	6307#	
\$50627	020444	6312	6328#	
\$50630	020430	6315	6320#	
\$50631	020444	6319	6326#	
\$50632	020512	6333	6335	6354#
\$50633	020470	6341#	6349	
\$50634	020504	6343	6350#	
\$50635	021072	6357	6359	6483#
\$50636	020604	6368	6384#	
\$50637	020572	6375#	6381	
\$50640	020604	6377	6382#	
\$50641	020632	6387	6394#	
\$50642	020632	6396#	6404	
\$50643	020710	6407	6409	6417#
\$50644	020730	6420	6423#	
\$50645	021004	6422	6451#	
\$50646	020744	6426	6431#	
\$50647	020750	6430	6435#	
\$50650	021000	6441	6447#	
\$50651	021024	6454	6457#	
\$50652	021072	6456	6481#	
\$50653	021042	6460	6465#	
\$50654	021066	6468	6475#	
\$50655	021072	6474	6479#	
\$50656	021134	6499	6506#	

\$50657	021140	6505	6512#	
\$50660	021222	6544	6548#	
\$50661	021234	6551	6555#	
\$50662	021246	6558	6562#	
\$50663	021260	6565	6569#	
\$50664	021272	6572	6576#	
\$50665	021304	6579	6583#	
\$50666	021316	6586	6590#	
\$50667	021332	6604#	6617	
\$50670	021360	6606	6608	6618#
\$50671	021356	6611	6615#	
\$50672	021404	6626	6630#	
\$50673	021450	6652	6656#	
\$50674	021460	6659	6663#	
\$50675	021470	6666	6670#	
\$50676	021500	6673	6677#	
\$50677	021510	6680	6684#	
\$5070	045366	13856#	14043	
\$50700	021520	6687	6691#	
\$50701	021530	6694	6698#	
\$50702	021546	6703	6707#	
\$50703	021552	6711#	6723	
\$50704	021564	6714	6718#	
\$50705	021600	6721	6724#	
\$50706	022020	6831	6838#	
\$50707	022024	6837	6844#	
\$5071	046514	14237#	14430	
\$50710	022116	6867	6876#	
\$50711	022402	6875	6981#	
\$50712	022160	6886	6894#	
\$50713	022160	6896#	6921	
\$50714	022230	6898	6922#	
\$50715	022402	6943	6979#	
\$50716	022330	6949	6952#	
\$50717	022402	6951	6977#	
\$5072	047636	14607#	15047	
\$50720	022360	6961	6965#	
\$50721	022374	6968	6971#	
\$50722	022402	6970	6975#	
\$50723	022450	7014	7018#	
\$50724	022462	7021	7025#	
\$50725	022474	7028	7032#	
\$50726	022506	7035	7039#	
\$50727	022520	7042	7046#	
\$5073	021752	6808#	7071	
\$50730	022532	7049	7053#	
\$50731	022544	7056	7060#	
\$50732	022570	7069	7073#	
\$50733	022634	7094	7098#	
\$50734	022644	7101	7105#	
\$50735	022654	7108	7112#	
\$50736	022664	7115	7119#	
\$50737	022674	7122	7126#	
\$5074	053144	15829#	16017	
\$50740	022704	7129	7133#	
\$50741	022714	7136	7140#	

\$50742	022730	7143	7147#	
\$50743	022744	7150	7154#	
\$50744	023140	7256	7263#	
\$50745	023144	7262	7269#	
\$50746	023236	7292	7301#	
\$50747	023274	7300	7318#	
\$5075	054360	16262#	16455	
\$50750	023260	7309#	7316	
\$50751	023604	7325	7423#	
\$50752	023334	7328	7331#	
\$50753	023374	7330	7347#	
\$50754	023374	7339	7345#	
\$50755	023414	7350	7353#	
\$50756	023524	7352	7396#	
\$50757	023430	7356	7361#	
\$5076	051240	15148#	15622	
\$50760	023444	7360	7367#	
\$50761	023454	7373#	7380	
\$50762	023516	7385	7391#	
\$50763	023542	7399	7404#	
\$50764	023562	7407	7410#	
\$50765	023604	7409	7421#	
\$50766	023604	7413	7419#	
\$50767	023614	7427	7430#	
\$5077	016562	5715#	6044	
\$50770	023612	7428#	7436	
\$50771	023630	7432	7437#	
\$50772	023674	7443	7457#	
\$50773	023666	7448	7451#	
\$50774	023674	7450	7455#	
\$50775	023742	7490	7494#	
\$50776	023754	7497	7501#	
\$50777	023766	7504	7508#	
\$51000	024000	7511	7515#	
\$51001	024012	7518	7522#	
\$51002	024024	7525	7529#	
\$51003	024036	7532	7536#	
\$51004	024062	7545	7549#	
\$51005	024126	7570	7574#	
\$51006	024142	7577	7579	7582#
\$51007	024146	7581	7586#	
\$51010	024162	7589	7591	7594#
\$51011	024166	7593	7598#	
\$51012	024202	7601	7605#	
\$51013	024216	7608	7612#	
\$51014	025204	8007	8014#	
\$51015	025210	8013	8020#	
\$51016	025310	8064	8067#	
\$51017	025316	8066	8072#	
\$51020	025340	8071	8086#	
\$51021	025332	8075	8080#	
\$51022	025340	8079	8084#	
\$51023	025356	8096	8101#	
\$51024	025362	8100	8105#	
\$51025	025430	8151#	8182	
\$51026	025506	8153	8183#	

\$51027	025504	8166	8180#	
\$51030	025506	8190#	8196	
\$51031	025522	8192	8197#	
\$51032	025546	8206	8208	8212#
\$51033	025562	8222	8225#	
\$51034	025570	8224	8229#	
\$51035	025606	8238	8242#	
\$51036	025644	8279#	8285	
\$51037	025656	8281	8286#	
\$51040	025700	8296	8300#	
\$51041	025730	8308	8315#	
\$51042	026002	8320	8344#	
\$51043	026002	8325	8342#	
\$51044	025776	8331	8336#	
\$51045	026002	8335	8340#	
\$51046	026042	8349	8363#	
\$51047	026034	8352	8359#	
\$51050	026170	8368	8408#	
\$51051	026076	8371	8373	8377#
\$51052	026170	8380	8406#	
\$51053	026130	8383	8385	8389#
\$51054	026164	8398	8402#	
\$51055	026240	8413	8432#	
\$51056	026240	8416	8430#	
\$51057	026232	8422	8426#	
\$51060	026302	8437	8454#	
\$51061	026276	8443	8448#	
\$51062	026302	8447	8452#	
\$51063	026352	8459	8478#	
\$51064	026344	8467	8472#	
\$51065	026352	8471	8476#	
\$51066	026434	8509	8513#	
\$51067	026446	8516	8520#	
\$51070	026460	8523	8527#	
\$51071	026472	8530	8534#	
\$51072	026504	8537	8541#	
\$51073	026516	8544	8548#	
\$51074	026530	8551	8555#	
\$51075	026544	8565#	8578	
\$51076	026572	8567	8569	8579#
\$51077	026570	8572	8576#	
\$51100	026622	8593	8597#	
\$51101	026670	8618	8622#	
\$51102	026704	8625	8627	8630#
\$51103	026710	8629	8634#	
\$51104	026724	8637	8639	8642#
\$51105	026730	8641	8646#	
\$51106	026744	8649	8653#	
\$51107	026760	8656	8660#	
\$51110	026774	8663	8667#	
\$51111	027010	8670	8674#	
\$51112	027312	8831	8834#	
\$51113	027310	8832#	8840	
\$51114	027324	8836	8841#	
\$51115	027376	8856	8877#	
\$51116	027376	8859	8875#	

\$51117	027372	8864	8869#	
\$51120	027376	8868	8873#	
\$51121	027422	8883	8894#	
\$51122	027422	8886	8892#	
\$51123	027464	8900	8920#	
\$51124	027464	8903	8918#	
\$51125	027460	8910	8914#	
\$51126	027530	8926	8944#	
\$51127	027530	8929	8942#	
\$51130	027522	8934	8938#	
\$51131	027560	8950	8964#	
\$51132	027554	8953	8958#	
\$51133	027560	8957	8962#	
\$51134	027614	8970	8984#	
\$51135	027606	8973	8978#	
\$51136	027614	8977	8982#	
\$51137	027714	9017	9022#	
\$51140	027744	9021	9039#	
\$51141	027732	9025	9028#	
\$51142	027740	9027	9033#	
\$51143	027744	9032	9037#	
\$51144	027760	9042	9046#	
\$51145	030022	9055	9057	9068#
\$51146	030022	9060	9062	9066#
\$51147	030036	9077	9080#	
\$51150	030044	9079	9084#	
\$51151	030062	9093	9097#	
\$51152	030100	9107	9112#	
\$51153	030104	9111	9116#	
\$51154	030146	9143	9153#	
\$51155	030146	9163#	9192	
\$51156	030210	9165	9193#	
\$51157	030206	9178	9190#	
\$51160	030322	9238	9242#	
\$51161	030334	9245	9249#	
\$51162	030346	9252	9256#	
\$51163	030360	9259	9263#	
\$51164	030372	9266	9270#	
\$51165	030404	9273	9277#	
\$51166	030416	9280	9284#	
\$51167	030432	9294#	9307	
\$51170	030460	9296	9298	9308#
\$51171	030456	9301	9305#	
\$51172	030504	9316	9320#	
\$51173	030550	9342	9346#	
\$51174	030564	9349	9351	9354#
\$51175	030570	9353	9358#	
\$51176	030604	9361	9363	9366#
\$51177	030610	9365	9370#	
\$51200	030624	9373	9377#	
\$51201	030640	9380	9384#	
\$51202	030654	9387	9391#	
\$51203	032016	9918#		
\$51204	032016	9919#		
\$51205	031142	9553	9557#	
\$51206	031270	9563	9565	9605#

\$51207	031202	9568	9574#	
\$51210	031270	9573	9603#	
\$51211	031312	9614	9627#	
\$51212	031312	9621	9625#	
\$51213	031316	9636#	9648	
\$51214	031342	9638	9649#	
\$51215	031366	9652	9654	9662#
\$51216	031370	9671#	9683	
\$51217	031414	9673	9684#	
\$51220	031440	9687	9689	9697#
\$51221	031442	9706#	9718	
\$51222	031466	9708	9719#	
\$51223	031512	9722	9724	9732#
\$51224	031514	9741#	9753	
\$51225	031540	9743	9754#	
\$51226	031564	9757	9759	9767#
\$51227	031566	9776#	9788	
\$51230	031610	9778	9789#	
\$51231	031634	9792	9794	9802#
\$51232	031636	9811#	9821	
\$51233	031656	9813	9822#	
\$51234	031656	9824#	9832	
\$51235	031674	9826	9833#	
\$51236	031676	9842#	9850	
\$51237	031714	9844	9851#	
\$51240	031716	9860#	9868	
\$51241	031734	9862	9869#	
\$51242	031736	9878#	9886	
\$51243	031754	9880	9887#	
\$51244	031764	9900#	9906	
\$51245	031774	9902	9907#	
\$51246	032430	10170#		
\$51247	032430	10171#		
\$51250	032040	9956#	9968	
\$51251	032070	9958	9969#	
\$51252	032072	9978#	9990	
\$51253	032122	9980	9991#	
\$51254	032124	10000#	10012	
\$51255	032154	10002	10013#	
\$51256	032156	10022#	10034	
\$51257	032206	10024	10035#	
\$51260	032210	10044#	10056	
\$51261	032236	10046	10057#	
\$51262	032240	10066#	10076	
\$51263	032262	10068	10077#	
\$51264	032264	10086#	10096	
\$51265	032306	10088	10097#	
\$51266	032310	10106#	10116	
\$51267	032332	10108	10117#	
\$51270	032334	10126#	10136	
\$51271	032356	10128	10137#	
\$51272	032424	10153	10166#	
\$51273	032424	10160	10164#	
\$51274	032472	10211	10220#	
\$51275	032472	10214	10218#	
\$51276	032554	10253	10256#	

\$51277	032550	10254#	10262				
\$51300	032570	10258	10263#				
\$51301	032670	10297	10314#				
\$51302	032674	10313	10318#				
\$51303	032730	10326	10343#				
\$51304	032734	10342	10347#				
\$51305	032774	10355	10374#				
\$51306	033000	10373	10378#				
\$51307	033156	10419	10423#				
\$51310	033224	10445	10447#				
\$51311	033454	10453	10534#				
\$51312	033410	10452	10512#				
\$51313	033344	10451	10493#				
\$51314	033300	10450	10475#				
\$51315	033240	10449	10455#				
\$51316	033524	10448	10556#				
\$51317	033524	10443	10474	10492	10511	10533	10557#
\$51320	033264	10460	10465#				
\$51321	033272	10464	10469#				
\$51322	033334	10483	10487#				
\$51323	033400	10501	10505#				
\$51324	033446	10521	10526#				
\$51325	033452	10525	10530#				
\$51326	033516	10545	10550#				
\$51327	033524	10549	10554#				
\$51330	033550	10566	10570#				
\$51331	033616	10592	10594#				
\$51332	034046	10600	10680#				
\$51333	034002	10599	10658#				
\$51334	033736	10598	10640#				
\$51335	033672	10597	10622#				
\$51336	033632	10596	10602#				
\$51337	034116	10595	10702#				
\$51340	034116	10590	10621	10639	10657	10679	10703#
\$51341	033656	10607	10612#				
\$51342	033664	10611	10616#				
\$51343	033726	10630	10634#				
\$51344	033772	10648	10652#				
\$51345	034040	10667	10672#				
\$51346	034044	10671	10676#				
\$51347	034110	10691	10696#				
\$51350	034116	10695	10700#				
\$51351	034142	10712	10716#				
\$51352	034210	10738	10740#				
\$51353	034472	10746	10845#				
\$51354	034426	10745	10823#				
\$51355	034350	10744	10798#				
\$51356	034272	10743	10773#				
\$51357	034224	10742	10748#				
\$51360	034542	10741	10867#				
\$51361	034542	10736	10772	10797	10822	10844	10868#
\$51362	034270	10753	10769#				
\$51363	034256	10756	10761#				
\$51364	034264	10760	10765#				
\$51365	034346	10779	10794#				
\$51366	034340	10786	10790#				

\$51367	034424	10804	10819#				
\$51370	034416	10811	10815#				
\$51371	034464	10832	10837#				
\$51372	034470	10836	10841#				
\$51373	034534	10856	10861#				
\$51374	034542	10860	10865#				
\$51375	034650	10911	10915#				
\$51376	034662	10918	10922#				
\$51377	034674	10925	10929#				
\$51400	034706	10932	10936#				
\$51401	034720	10939	10943#				
\$51402	034732	10946	10950#				
\$51403	034744	10953	10957#				
\$51404	034760	10970#	10983				
\$51405	035006	10972	10974	10984#			
\$51406	035004	10977	10981#				
\$51407	035046	11007	11011#				
\$51410	035214	11087	11091#				
\$51411	035230	11099	11101	11104#			
\$51412	035234	11103	11108#				
\$51413	035250	11111	11113	11116#			
\$51414	035254	11115	11120#				
\$51415	035270	11133#	11146				
\$51416	035316	11135	11137	11147#			
\$51417	035314	11140	11144#				
\$51420	035352	11174	11177#				
\$51421	035346	11175#	11183				
\$51422	035366	11179	11184#				
\$51423	035476	11222	11239#				
\$51424	035502	11238	11243#				
\$51425	035536	11251	11268#				
\$51426	035542	11267	11272#				
\$51427	035602	11280	11299#				
\$51430	035606	11298	11303#				
\$51431	035764	11344	11348#				
\$51432	036032	11370	11372#				
\$51433	036262	11378	11459#				
\$51434	036216	11377	11437#				
\$51435	036152	11376	11418#				
\$51436	036106	11375	11400#				
\$51437	036046	11374	11380#				
\$51440	036332	11373	11481#				
\$51441	036332	11368	11399	11417	11436	11458	11482#
\$51442	036072	11385	11390#				
\$51443	036100	11389	11394#				
\$51444	036142	11408	11412#				
\$51445	036206	11426	11430#				
\$51446	036254	11446	11451#				
\$51447	036260	11450	11455#				
\$51450	036324	11470	11475#				
\$51451	036332	11474	11479#				
\$51452	036356	11491	11495#				
\$51453	036430	11519	11521#				
\$51454	036660	11527	11607#				
\$51455	036614	11526	11585#				
\$51456	036550	11525	11567#				

\$51457	036504	11524	11549#				
\$51460	036444	11523	11529#				
\$51461	036730	11522	11629#				
\$51462	036730	11517	11548	11566	11584	11606	11630#
\$51463	036470	11534	11539#				
\$51464	036476	11538	11543#				
\$51465	036540	11557	11561#				
\$51466	036604	11575	11579#				
\$51467	036652	11594	11599#				
\$51470	036656	11598	11603#				
\$51471	036722	11618	11623#				
\$51472	036730	11622	11627#				
\$51473	036754	11639	11643#				
\$51474	037022	11665	11667#				
\$51475	037304	11673	11772#				
\$51476	037240	11672	11750#				
\$51477	037162	11671	11725#				
\$51500	037104	11670	11700#				
\$51501	037036	11669	11675#				
\$51502	037354	11668	11794#				
\$51503	037354	11663	11699	11724	11749	11771	11795#
\$51504	037102	11680	11696#				
\$51505	037070	11683	11688#				
\$51506	037076	11687	11692#				
\$51507	037160	11706	11721#				
\$51510	037152	11713	11717#				
\$51511	037236	11731	11746#				
\$51512	037230	11738	11742#				
\$51513	037276	11759	11764#				
\$51514	037302	11763	11768#				
\$51515	037346	11783	11788#				
\$51516	037354	11787	11792#				
\$51517	037462	11838	11842#				
\$51520	037474	11845	11849#				
\$51521	037506	11852	11856#				
\$51522	037520	11859	11863#				
\$51523	037532	11866	11870#				
\$51524	037544	11873	11877#				
\$51525	037556	11880	11884#				
\$51526	037572	11898#	11911				
\$51527	037620	11900	11902	11912#			
\$51530	037616	11905	11909#				
\$51531	037660	11935	11939#				
\$51532	040026	12015	12019#				
\$51533	040042	12027	12029	12032#			
\$51534	040046	12031	12036#				
\$51535	040062	12039	12041	12044#			
\$51536	040066	12043	12048#				
\$51537	040102	12061#	12074				
\$51540	040130	12063	12065	12075#			
\$51541	040126	12068	12072#				
\$51542	040250	12135	12138#				
\$51543	040246	12136#	12144				
\$51544	040262	12140	12145#				
\$51545	040274	12148	12152#				
\$51546	040324	12162	12165#				

\$51547	040322	12163#	12171	
\$51550	040336	12167	12172#	
\$51551	040366	12183	12185	12190#
\$51552	040436	12189	12213#	
\$51553	040412	12193	12195	12200#
\$51554	040436	12199	12211#	
\$51555	040436	12203	12205	12209#
\$51556	040704	12216	12218	12316#
\$51557	040502	12221	12231#	
\$51560	040500	12224	12228#	
\$51561	040704	12230	12314#	
\$51562	040536	12236	12238	12243#
\$51563	040564	12242	12254#	
\$51564	040564	12246	12248	12252#
\$51565	040704	12257	12312#	
\$51566	040610	12270#	12278	
\$51567	040624	12272	12274	12279#
\$51570	040676	12282	12306#	
\$51571	040652	12285	12287	12292#
\$51572	040674	12291	12303#	
\$51573	040674	12295	12297	12301#
\$51574	040704	12305	12310#	
\$51575	040722	12333	12337#	
\$51576	041010	12349	12370#	
\$51577	040776	12355	12360#	
\$51600	041002	12359	12364#	
\$51601	041062	12373	12399#	
\$51602	041062	12376	12397#	
\$51603	041050	12382	12387#	
\$51604	041054	12386	12391#	
\$51605	041130	12402	12427#	
\$51606	041130	12405	12425#	
\$51607	041116	12410	12415#	
\$51610	041122	12414	12419#	
\$51611	041170	12430	12447#	
\$51612	041164	12436	12441#	
\$51613	041170	12440	12445#	
\$51614	041236	12450	12471#	
\$51615	041220	12455	12460#	
\$51616	041224	12459	12464#	
\$51617	041260	12479	12483#	
\$51620	041346	12495	12516#	
\$51621	041334	12501	12506#	
\$51622	041340	12505	12510#	
\$51623	041412	12519	12540#	
\$51624	041400	12525	12530#	
\$51625	041404	12529	12534#	
\$51626	041452	12543	12563#	
\$51627	041440	12548	12553#	
\$51630	041444	12552	12557#	
\$51631	041512	12566	12583#	
\$51632	041506	12572	12577#	
\$51633	041512	12576	12581#	
\$51634	041560	12586	12607#	
\$51635	041542	12591	12596#	
\$51636	041546	12595	12600#	

\$51637	041646	12640	12644#	
\$51640	041656	12647	12651#	
\$51641	041666	12654	12658#	
\$51642	041676	12661	12665#	
\$51643	041706	12668	12672#	
\$51644	041716	12675	12679#	
\$51645	041726	12682	12686#	
\$51646	041750	12692	12696#	
\$51647	041772	12702	12706#	
\$51650	042016	12715	12719#	
\$51651	042104	12750	12754#	
\$51652	042126	12757	12759	12762#
\$51653	042132	12761	12766#	
\$51654	042154	12769	12771	12774#
\$51655	042160	12773	12778#	
\$51656	042276	12847#	12855	
\$51657	042312	12849	12856#	
\$51660	042334	12868	12872#	
\$51661	042366	12880	12882	12886#
\$51662	042424	12892	12909#	
\$51663	042410	12897	12900#	
\$51664	042406	12898#	12906	
\$51665	042424	12902	12907#	
\$51666	042440	12912	12915#	
\$51667	042450	12914	12920#	
\$51670	043026	12919	13071#	
\$51671	042632	12926	12995#	
\$51672	042540	12939	12953#	
\$51673	042532	12942	12947#	
\$51674	042540	12946	12951#	
\$51675	042552	12956	12960#	
\$51676	042552	12962#	12983	
\$51677	042614	12964	12966	12984#
\$51700	042606	12971	12973	12977#
\$51701	042632	12987	12989	12993#
\$51702	043026	13001	13069#	
\$51703	042674	13007	13011#	
\$51704	042744	13014	13034#	
\$51705	042726	13021	13024#	
\$51706	042724	13022#	13030	
\$51707	042742	13026	13031#	
\$51710	043026	13033	13067#	
\$51711	042764	13040	13043#	
\$51712	042762	13041#	13049	
\$51713	043000	13045	13050#	
\$51714	043004	13054#	13064	
\$51715	043026	13056	13058	13065#
\$51716	043044	13074	13078#	
\$51717	043120	13110	13114#	
\$51720	043150	13119	13128#	
\$51721	043200	13132	13141#	
\$51722	043364	13145	13207#	
\$51723	043250	13150	13167#	
\$51724	043244	13156	13161#	
\$51725	043250	13160	13165#	
\$51726	043364	13172	13205#	

\$51727	043316	13178	13180	13183#	
\$51730	043324	13182	13188#		
\$51731	043330	13187	13192#		
\$51732	043364	13195	13197	13199	13203#
\$51733	043446	13210	13234#		
\$51734	043424	13215	13222#		
\$51735	043446	13225	13232#		
\$51736	043662	13237	13310#		
\$51737	043542	13240	13266#		
\$51740	043512	13243	13245	13249#	
\$51741	043536	13255	13260#		
\$51742	043542	13259	13264#		
\$51743	043662	13269	13308#		
\$51744	043604	13275	13277	13280#	
\$51745	043610	13279	13285#		
\$51746	043614	13284	13289#		
\$51747	043650	13292	13294	13296	13300#
\$51750	044040	13313	13371#		
\$51751	043724	13316	13330#		
\$51752	043716	13319	13324#		
\$51753	043724	13323	13328#		
\$51754	044040	13333	13369#		
\$51755	043756	13336	13338	13341#	
\$51756	043762	13340	13346#		
\$51757	043766	13345	13350#		
\$51760	044022	13353	13355	13357	13361#
\$51761	044232	13374	13433#		
\$51762	044102	13383	13388#		
\$51763	044110	13387	13392#		
\$51764	044142	13401	13404#		
\$51765	044152	13403	13409#		
\$51766	044160	13408	13413#		
\$51767	044216	13416	13418	13420	13424#
\$51770	044232	13427	13431#		
\$51771	044434	13436	13496#		
\$51772	044320	13452	13457#		
\$51773	044326	13456	13461#		
\$51774	044344	13464	13467#		
\$51775	044354	13466	13472#		
\$51776	044362	13471	13476#		
\$51777	044420	13479	13481	13483	13487#
\$52000	044434	13490	13494#		
\$52001	044526	13529	13533#		
\$52002	044542	13536	13538	13541#	
\$52003	044546	13540	13545#		
\$52004	044574	13551	13555#		
\$52005	044606	13558	13562#		
\$52006	044626	13573#	13588		
\$52007	044656	13575	13577	13589#	
\$5201	040704	12330#	12704		
\$52010	044652	13580	13584#		
\$52011	044670	13592	13596#		
\$52012	044704	13605	13608#		
\$52013	044702	13606#	13614		
\$52014	044716	13610	13615#		
\$52015	044742	13621	13627#		

\$52016	044754	13631	13635#	
\$52017	045064	13683	13687#	
\$5202	041242	12476#	12694	
\$52020	045100	13690	13692	13695#
\$52021	045104	13694	13699#	
\$52022	045120	13702	13704	13707#
\$52023	045124	13706	13711#	
\$52024	045142	13717	13721#	
\$52025	045154	13724	13728#	
\$52026	045160	13732#	13747	
\$52027	045214	13734	13736	13748#
\$52030	045212	13739	13745#	
\$52031	045270	13801	13810#	
\$52032	045270	13804	13808#	
\$52033	045352	13843	13846#	
\$52034	045346	13844#	13852	
\$52035	045366	13848	13853#	
\$52036	045676	13960	13964#	
\$52037	045710	13967	13971#	
\$52040	045722	13974	13978#	
\$52041	045734	13981	13985#	
\$52042	045746	13988	13992#	
\$52043	045760	13995	13999#	
\$52044	045772	14002	14006#	
\$52045	046006	14019#	14032	
\$52046	046034	14021	14023	14033#
\$52047	046032	14026	14030#	
\$52050	046060	14041	14045#	
\$52051	046320	14128	14132#	
\$52052	046334	14140	14142	14145#
\$52053	046340	14144	14149#	
\$52054	046354	14152	14154	14157#
\$52055	046360	14156	14161#	
\$52056	046416	14183#	14196	
\$52057	046444	14185	14187	14197#
\$52060	046442	14190	14194#	
\$52061	046500	14224	14227#	
\$52062	046474	14225#	14233	
\$52063	046514	14229	14234#	
\$52064	047034	14346	14350#	
\$52065	047046	14353	14357#	
\$52066	047060	14360	14364#	
\$52067	047072	14367	14371#	
\$52070	047104	14374	14378#	
\$52071	047116	14381	14385#	
\$52072	047130	14388	14392#	
\$52073	047144	14406#	14419	
\$52074	047172	14408	14410	14420#
\$52075	047170	14413	14417#	
\$52076	047216	14428	14432#	
\$52077	047466	14516	14520#	
\$52100	047502	14528	14530	14533#
\$52101	047506	14532	14537#	
\$52102	047522	14540	14542	14545#
\$52103	047526	14544	14549#	
\$52104	047564	14571#	14584	

CJKDHAO KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 513
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0512

\$52105	047612	14573	14575	14585#
\$52106	047610	14578	14582#	
\$52107	047732	14646	14649#	
\$52110	047730	14647#	14655	
\$52111	047742	14651	14656#	
\$52112	047776	14668	14671#	
\$52113	047774	14669#	14677	
\$52114	050006	14673	14678#	
\$52115	050036	14689	14691	14696#
\$52116	050106	14695	14719#	
\$52117	050062	14699	14701	14706#
\$52120	050106	14705	14717#	
\$52121	050106	14709	14711	14715#
\$52122	050354	14722	14724	14822#
\$52123	050152	14727	14737#	
\$52124	050150	14730	14734#	
\$52125	050354	14736	14820#	
\$52126	050206	14742	14744	14749#
\$52127	050234	14748	14760#	
\$52130	050234	14752	14754	14758#
\$52131	050354	14763	14818#	
\$52132	050260	14776#	14784	
\$52133	050274	14778	14780	14785#
\$52134	050346	14788	14812#	
\$52135	050322	14791	14793	14798#
\$52136	050344	14797	14809#	
\$52137	050344	14801	14803	14807#
\$52140	050354	14811	14816#	
\$52141	050426	14843	14858#	
\$52142	050420	14846	14851#	
\$52143	050424	14850	14855#	
\$52144	050516	14857	14904#	
\$52145	050440	14861	14867#	
\$52146	050440	14869#	14883	
\$52147	050466	14871	14884#	
\$52150	050510	14891	14896#	
\$52151	050514	14895	14900#	
\$52152	050564	14915	14930#	
\$52153	050556	14918	14923#	
\$52154	050562	14922	14927#	
\$52155	050654	14929	14976#	
\$52156	050576	14933	14939#	
\$52157	050576	14941#	14955	
\$52160	050624	14943	14955#	
\$52161	050646	14963	14968#	
\$52162	050652	14967	14972#	
\$52163	050742	15009	15013#	
\$52164	050756	15016	15018	15021#
\$52165	050762	15020	15025#	
\$52166	050776	15028	15030	15033#
\$52167	051002	15032	15037#	
\$52170	051026	15045	15049#	
\$52171	051114	15081	15085#	
\$52172	051136	15088	15090	15093#
\$52173	051142	15092	15097#	
\$52174	051164	15100	15102	15105#

\$52175	051170	15104	15109#		
\$52176	051326	15179	15194#		
\$52177	051320	15182	15187#		
\$52200	051324	15186	15191#		
\$52201	051402	15193	15228#		
\$52202	051342	15197	15203#		
\$52203	051342	15205#	15213		
\$52204	051362	15207	15214#		
\$52205	051376	15217	15222#		
\$52206	051402	15221	15226#		
\$52207	051424	15240	15244#		
\$52210	051456	15252	15254	15258#	
\$52211	051504	15270	15273#		
\$52212	051512	15272	15278#		
\$52213	051516	15277	15282#		
\$52214	051532	15288	15291#		
\$52215	051540	15290	15295#		
\$52216	052030	15298	15300	15302	15409#
\$52217	051672	15321	15345#		
\$52220	051640	15324	15329#		
\$52221	051672	15378	15343#		
\$52222	051730	15348	15362#		
\$52223	051730	15364#	15397		
\$52224	052012	15366	15368	15398#	
\$52225	052000	15371	15387#		
\$52226	051774	15376	15378	15382#	
\$52227	052004	15386	15391#		
\$52230	052030	15401	15403	15407#	
\$52231	052252	15412	15414	15416	15497#
\$52232	052252	15423	15495#		
\$52233	052116	15429	15433#		
\$52234	052144	15436	15443#		
\$52235	052252	15442	15493#		
\$52236	052210	15464#	15490		
\$52237	052252	15466	15468	15491#	
\$52240	052236	15471	15478#		
\$52241	052242	15477	15482#		
\$52242	052342	15528	15532#		
\$52243	052352	15535	15538#		
\$52244	052356	15537	15542#		
\$52245	052366	15545	15548#		
\$52246	052372	15547	15552#		
\$52247	052420	15558	15562#		
\$52250	052432	15565	15569#		
\$52251	052452	15581#	15596		
\$52252	052502	15583	15585	15597#	
\$52253	052476	15588	15592#		
\$52254	052514	15600	15604#		
\$52255	052540	15610	15616#		
\$52256	052552	15620	15624#		
\$52257	052642	15656	15660#		
\$52260	052656	15663	15665	15668#	
\$52261	052662	15667	15672#		
\$52262	052676	15675	15677	15680#	
\$52263	052702	15679	15684#		
\$52264	052720	15690	15694#		

CJKDHAO KEF11-B CIS DIAGNOSTIC
C IKDHA.P11 29-JUN-81 16:54

MACY11 30A(1052) 14-JUL-81 14:19 PAGE 515
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0514

\$52265	052732	15697	15701#	
\$52266	052736	15705#	15720	
\$52267	052772	15707	15709	15721#
\$52270	052770	15712	15718#	
\$52271	053046	15773	15782#	
\$52272	053046	15776	15780#	
\$52273	053130	15816	15819#	
\$52274	053124	15817#	15825	
\$52275	053144	15821	15826#	
\$52276	053454	15933	15937#	
\$52277	053466	15940	15944#	
\$52300	053500	15947	15951#	
\$52301	053512	15954	15958#	
\$52302	053524	15961	15965#	
\$52303	053536	15968	15972#	
\$52304	053550	15975	15979#	
\$52305	053564	15993#	16006	
\$52306	053612	15995	15997	16007#
\$52307	053610	16000	16004#	
\$52310	053636	16015	16019#	
\$52311	054076	16100	16104#	
\$52312	054112	16112	16114	16117#
\$52313	054116	16116	16121#	
\$52314	054132	16124	16126	16129#
\$52315	054136	16128	16133#	
\$52316	054174	16155#	16168	
\$52317	054222	16157	16159	16169#
\$52320	054220	16162	16166#	
\$52321	054262	16206	16215#	
\$52322	054262	16209	16213#	
\$52323	054344	16249	16252#	
\$52324	054340	16250#	16258	
\$52325	054360	16254	16259#	
\$52326	054700	16371	16375#	
\$52327	054712	16378	16382#	
\$52330	054724	16385	16389#	
\$52331	054736	16392	16396#	
\$52332	054750	16399	16403#	
\$52333	054762	16406	16410#	
\$52334	054774	16413	16417#	
\$52335	055010	16431#	16444	
\$52336	055036	16433	16435	16445#
\$52337	055034	16438	16442#	
\$52340	055062	16453	16457#	
\$52341	055332	16542	16546#	
\$52342	055346	16554	16556	16559#
\$52343	055352	16558	16563#	
\$52344	055366	16566	16568	16571#
\$52345	055372	16570	16575#	
\$52346	055430	16597#	16610	
\$52347	055456	16599	16601	16611#
\$52350	055454	16604	16608#	
\$52351	056332	16860	16862	16870#
\$52352	056620	16991#		
\$52353	056620	16992#		
\$52354	056662	17029#		

CJKDHA0 KEF11-B CIS DIAGNOSTIC
CJKDHA.P11 29-JUN-81 16:54

MACY11

30A(1052) 14-JUL-81 14:19 PAGE 516
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0515

I 8

\$52355	056662	17030#																				
\$52356	056642	17016#	17023																			
\$52357	056726	17066#																				
\$52360	056726	17067#																				
\$52361	056704	17053#	17060																			
\$52362	056772	17103#																				
\$52363	056772	17104#																				
\$52364	056750	17090#	17097																			
\$52365	057036	17138#																				
\$52366	057036	17139#																				
\$52367	057014	17125#	17132																			
\$52370	057100	17176#																				
\$52371	057100	17177#																				
\$52372	057060	17163#	17170																			
\$52373	057132	17203#																				
\$52374	057132	17204#																				
\$52375	057164	17230#																				
\$52376	057164	17231#																				
\$52377	057416	17342#																				
\$52400	057416	17343#																				
\$52401	057212	17263	17268#																			
\$52402	057226	17267	17277#																			
\$52403	057226	17271	17275#																			
\$52404	057372	17285	17320#																			
\$52405	057410	17328	17336#																			
\$52406	057416	17335	17340#																			
\$52407	057540	17466#																				
\$52410	057540	17467#																				
\$52411	057452	17403	17408#																			
\$52412	057456	17407	17412#																			
\$52413	057466	17423	17428#																			
\$52414	057540	17427	17464#																			
\$52415	057500	17431	17437#																			
\$52416	057500	17439#	17453																			
\$52417	057526	17441	17454#																			
\$5401	025640	8272#	8582																			
\$5402	025664	8270	8293#																			
\$5403	026576	8298	8584#																			
\$5601	043076	13103#	13599																			
\$5602	043104	13101	13107#																			
\$5603	044674	13112	13601#																			
.	076006	981#	982	985#	990#	994#	997#	1002	1003#	1005#	1007#	1013	1014#	1016#								
		1018#	1036#	1046#	1170#	1178	1192	1268	1269	1271	1272	1313	1314	1344								
		1345	1347	1348	1357	1358	1366	1367	1373	1374	1380	1381	1387	1388								
		1394	1395	1401	1402	1404	1405	1414	1415	1425	1426	1449	1450	1456								
		1457	1463	1464	1470	1471	1477	1478	1484	1485	1533	1534	1564	1565								
		1567	1568	1577	1578	1586	1587	1593	1594	1600	1601	1607	1608	1614								
		1615	1621	1622	1630	1631	1652	1653	1659	1660	1666	1667	1673	1674								
		1680	1681	1687	1688	1694	1695	1761	1762	1781	1782	1807	1808	1815								
		1816	1831	1832	1839	1840	1853	1854	1881	1882	1888	1889	1895	1896								
		1902	1903	1909	1910	1916	1917	1923	1924	1941	1942	1947	1948	1949								
		1950	1952	1953	1970	1971	1972	1973	1975	1976	1995	1996	2006	2007								
		2014	2015	2017	2018	2037	2038	2046	2047	2057	2058	2099	2100	2101								
		2102	2110	2111	2117	2118	2124	2125	2131	2132	2138	2139	2145	2146								
		2152	2153	2159	2160	2173	2174	2182	2183	2377	2378	2397	2398	2423								
		2424	2431	2432	2447	2448	2455	2456	2487	2488	2494	2495	2501	2502								

2508	2509	2515	2516	2522	2523	2529	2530	2545	2546	2555	2556	2557
2558	2560	2561	2577	2578	2590	2591	2604	2605	2606	2607	2609	2610
2621	2622	2629	2630	2641	2642	2652	2653	2694	2695	2696	2697	2705
2706	2712	2713	2719	2720	2726	2727	2733	2734	2740	2741	2747	2748
2754	2755	2767	2768	2776	2777	2892	2893	2918	2919	2926	2927	2942
2943	2950	2951	3008	3009	3015	3016	3022	3023	3029	3030	3036	3037
3043	3044	3050	3051	3072	3073	3078	3079	3080	3081	3083	3084	3101
3102	3103	3104	3106	3107	3127	3128	3138	3139	3144	3145	3156	3157
3165	3166	3176	3177	3220	3221	3222	3223	3231	3232	3238	3239	3245
3246	3252	3253	3259	3260	3266	3267	3273	3274	3280	3281	3294	3295
3303	3304	3396	3397	3416	3417	3474	3475	3481	3482	3488	3489	3495
3496	3502	3503	3509	3510	3516	3517	3528	3529	3561	3562	3570	3571
3577	3578	3584	3585	3591	3592	3598	3599	3605	3606	3682	3683	3702
3703	3760	3761	3767	3768	3774	3775	3781	3782	3788	3789	3795	3796
3802	3803	3814	3815	3847	3848	3857	3858	3864	3865	3871	3872	3878
3879	3885	3886	3892	3893	3966	3967	3982	3983	4025	4026	4056	4057
4063	4064	4070	4071	4077	4078	4084	4085	4091	4092	4098	4099	4110
4111	4135	4136	4159	4160	4168	4169	4175	4176	4182	4183	4189	4190
4196	4197	4203	4204	4279	4280	4295	4296	4338	4339	4366	4367	4373
4374	4380	4381	4387	4388	4394	4395	4401	4402	4408	4409	4420	4421
4445	4446	4466	4467	4475	4476	4482	4483	4489	4490	4496	4497	4503
4504	4510	4511	4590	4591	4615	4616	4656	4657	4666	4667	4676	4677
4678	4679	4701	4702	4708	4709	4715	4716	4722	4723	4729	4730	4736
4737	4743	4744	4755	4756	4795	4796	4802	4803	4809	4810	4816	4817
4823	4824	4830	4831	4837	4838	4934	4935	4963	4964	4975	4976	4999
5000	5005	5006	5048	5049	5055	5056	5062	5063	5069	5070	5076	5077
5083	5084	5090	5091	5102	5103	5129	5130	5143	5144	5150	5151	5157
5158	5164	5165	5171	5172	5178	5179	5185	5186	5192	5193	5239	5240
5242	5243	5359	5360	5389	5390	5393	5394	5409	5410	5416	5417	5423
5424	5430	5431	5437	5438	5444	5445	5451	5452	5476	5477	5478	5479
5481	5482	5501	5502	5512	5513	5516	5517	5527	5528	5536	5537	5571
5572	5626	5627	5679	5680	5682	5683	5749	5750	5756	5757	5765	5766
5768	5769	5785	5786	5796	5797	5811	5812	5833	5834	5836	5837	5848
5849	5871	5872	5880	5881	5915	5916	5960	5961	5967	5968	5974	5975
5981	5982	5988	5989	5995	5996	6002	6003	6022	6023	6024	6025	6027
6028	6042	6043	6068	6069	6075	6076	6082	6083	6089	6090	6096	6097
6103	6104	6110	6111	6119	6120	6128	6129	6135	6136	6138	6139	6149
6150	6158	6159	6281	6282	6286	6287	6293	6294	6296	6297	6312	6313
6315	6316	6333	6334	6335	6336	6343	6344	6357	6358	6359	6360	6368
6369	6377	6378	6387	6388	6404	6405	6407	6408	6409	6410	6422	6423
6426	6427	6441	6442	6456	6457	6460	6461	6468	6469	6499	6500	6544
6545	6551	6552	6558	6559	6565	6566	6572	6573	6579	6580	6586	6587
6606	6607	6608	6609	6611	6612	6626	6627	6652	6653	6659	6660	6666
6667	6673	6674	6680	6681	6687	6688	6694	6695	6703	6704	6714	6715
6723	6724	6831	6832	6867	6868	6886	6887	6898	6899	6943	6944	6951
6952	6961	6962	6970	6971	7014	7015	7021	7022	7028	7029	7035	7036
7042	7043	7049	7050	7056	7057	7069	7070	7094	7095	7101	7102	7108
7109	7115	7116	7122	7123	7129	7130	7136	7137	7143	7144	7150	7151
7214	7256	7257	7292	7293	7316	7317	7325	7326	7330	7331	7339	7340
7352	7353	7356	7357	7380	7381	7385	7386	7399	7400	7409	7410	7413
7414	7443	7444	7450	7451	7490	7491	7497	7498	7504	7505	7511	7512
7518	7519	7525	7526	7532	7533	7545	7546	7570	7571	7581	7582	7593
7594	7601	7602	7608	7609	8007	8008	8066	8067	8075	8076	8096	8097
8153	8154	8166	8167	8192	8193	8206	8207	8208	8209	8224	8225	8238
8239	8281	8282	8296	8297	8308	8309	8320	8321	8325	8326	8331	8332
8349	8350	8352	8353	8368	8369	8371	8372	8373	8374	8380	8381	8383

8384	8385	8386	8398	8399	8413	8414	8416	8417	8422	8423	8437	8438
8443	8444	8459	8460	8467	8468	8509	8510	8516	8517	8523	8524	8530
8531	8537	8538	8544	8545	8551	8552	8567	8568	8569	8570	8572	8573
8593	8594	8618	8619	8629	8630	8641	8642	8649	8650	8656	8657	8663
8664	8670	8671	8760#	8856	8857	8859	8860	8864	8865	8883	8884	8886
8887	8900	8901	8903	8904	8910	8911	8926	8927	8929	8930	8934	8935
8950	8951	8953	8954	8970	8971	8973	8974	9017	9018	9027	9028	9042
9043	9055	9056	9057	9058	9060	9061	9062	9063	9079	9080	9093	9094
9107	9108	9143	9144	9165	9166	9178	9179	9238	9239	9245	9246	9252
9253	9259	9260	9266	9267	9273	9274	9280	9281	9296	9297	9298	9299
9301	9302	9316	9317	9342	9343	9353	9354	9365	9366	9373	9374	9380
9381	9387	9388	9442#	9446#	9450#	9460#	9472#	9480#	9501#	9553	9554	9563
9564	9565	9566	9568	9569	9614	9615	9621	9622	9638	9639	9652	9653
9654	9655	9673	9674	9687	9688	9689	9690	9708	9709	9722	9723	9724
9725	9743	9744	9757	9758	9759	9760	9778	9779	9792	9793	9794	9795
9813	9814	9826	9827	9844	9845	9862	9863	9880	9881	9902	9903	9958
9959	9980	9981	10002	10003	10024	10025	10046	10047	10068	10069	10088	10089
10108	10109	10128	10129	10153	10154	10160	10161	10211	10212	10214	10215	10297
10298	10326	10327	10355	10356	10419	10420	10443	10444	10460	10461	10483	10484
10501	10502	10521	10522	10545	10546	10566	10567	10590	10591	10607	10608	10630
10631	10648	10649	10667	10668	10691	10692	10712	10713	10736	10737	10753	10754
10756	10757	10779	10780	10786	10787	10804	10805	10811	10812	10832	10833	10856
10857	10911	10912	10918	10919	10925	10926	10932	10933	10939	10940	10946	10947
10953	10954	10972	10973	10974	10975	10977	10978	11007	11008	11087	11088	11103
11104	11115	11116	11135	11136	11137	11138	11140	11141	11222	11223	11251	11252
11280	11281	11344	11345	11368	11369	11385	11386	11408	11409	11426	11427	11446
11447	11470	11471	11491	11492	11517	11518	11534	11535	11557	11558	11575	11576
11594	11595	11618	11619	11639	11640	11663	11664	11680	11681	11683	11684	11706
11707	11713	11714	11731	11732	11738	11739	11759	11760	11783	11784	11838	11839
11845	11846	11852	11853	11859	11860	11866	11867	11873	11874	11880	11881	11900
11901	11902	11903	11905	11906	11935	11936	12015	12016	12031	12032	12043	12044
12063	12064	12065	12066	12068	12069	12148	12149	12183	12184	12185	12186	12193
12194	12195	12196	12203	12204	12205	12206	12216	12217	12218	12219	12221	12222
12224	12225	12236	12237	12238	12239	12246	12247	12248	12249	12257	12258	12272
12273	12274	12275	12282	12283	12285	12286	12287	12288	12295	12296	12297	12298
12333	12334	12349	12350	12355	12356	12373	12374	12376	12377	12382	12383	12402
12403	12405	12406	12410	12411	12430	12431	12436	12437	12450	12451	12455	12456
12479	12480	12495	12496	12501	12502	12519	12520	12525	12526	12543	12544	12548
12549	12566	12567	12572	12573	12586	12587	12591	12592	12640	12641	12647	12648
12654	12655	12661	12662	12668	12669	12675	12676	12682	12683	12692	12693	12702
12703	12715	12716	12750	12751	12761	12762	12773	12774	12794#	12849	12850	12868
12869	12880	12881	12882	12883	12892	12893	12914	12915	12926	12927	12939	12940
12942	12943	12956	12957	12964	12965	12966	12967	12971	12972	12973	12974	12987
12988	12989	12990	13001	13002	13007	13008	13014	13015	13056	13057	13058	13059
13074	13075	13110	13111	13112	13120	13132	13133	13145	13146	13150	13151	13156
13157	13172	13173	13182	13183	13195	13196	13197	13198	13199	13200	13210	13211
13215	13216	13225	13226	13237	13238	13240	13241	13243	13244	13245	13246	13255
13256	13269	13270	13279	13280	13292	13293	13294	13295	13296	13297	13313	13314
13316	13317	13319	13320	13333	13334	13340	13341	13353	13354	13355	13356	13357
13358	13374	13375	13383	13384	13403	13404	13416	13417	13418	13419	13420	13421
13427	13428	13436	13437	13452	13453	13466	13467	13479	13480	13481	13482	13483
13484	13490	13491	13529	13530	13540	13541	13551	13552	13558	13559	13575	13576
13577	13578	13580	13581	13592	13593	13621	13622	13631	13632	13683	13684	13694
13695	13706	13707	13717	13718	13724	13725	13734	13735	13736	13737	13739	13740
13801	13802	13804	13805	13960	13961	13967	13968	13974	13975	13981	13982	13988
13989	13995	13996	14002	14003	14021	14022	14023	14024	14026	14027	14041	14042

14128	14129	14144	14145	14156	14157	14185	14186	14187	14188	14190	14191	14346
14347	14353	14354	14360	14361	14367	14368	14374	14375	14381	14382	14388	14389
14408	14409	14410	14411	14413	14414	14428	14429	14516	14517	14532	14533	14544
14545	14573	14574	14575	14576	14578	14579	14689	14690	14691	14692	14699	14700
14701	14702	14709	14710	14711	14712	14722	14723	14724	14725	14727	14728	14730
14731	14742	14743	14744	14745	14752	14753	14754	14755	14763	14764	14778	14779
14780	14781	14788	14789	14791	14792	14793	14794	14801	14802	14803	14804	14843
14844	14846	14847	14861	14862	14871	14872	14891	14892	14915	14916	14918	14919
14933	14934	14943	14944	14963	14964	15009	15010	15020	15021	15032	15033	15045
15046	15081	15082	15092	15093	15104	15105	15125#	15179	15180	15182	15183	15197
15198	15207	15208	15217	15218	15240	15241	15252	15253	15254	15255	15272	15273
15290	15291	15298	15299	15300	15301	15302	15303	15321	15322	15324	15325	15348
15349	15366	15367	15368	15369	15371	15372	15376	15377	15378	15379	15401	15402
15403	15404	15412	15413	15414	15415	15416	15417	15423	15424	15429	15430	15436
15437	15466	15467	15468	15469	15471	15472	15528	15529	15537	15538	15547	15548
15558	15559	15565	15566	15583	15584	15585	15586	15588	15589	15600	15601	15610
15611	15620	15621	15656	15657	15667	15668	15679	15680	15690	15691	15697	15698
15707	15708	15709	15710	15712	15713	15773	15774	15776	15777	15933	15934	15940
15941	15947	15948	15954	15955	15961	15962	15968	15969	15975	15976	15995	15996
15997	15998	16000	16001	16015	16016	16100	16101	16116	16117	16128	16129	16157
16158	16159	16160	16162	16163	16206	16207	16209	16210	16371	16372	16378	16379
16385	16386	16392	16393	16399	16400	16406	16407	16413	16414	16433	16434	16435
16436	16438	16439	16453	16454	16542	16543	16558	16559	16570	16571	16599	16600
16601	16602	16604	16605	16695#	16704#	16718#	16732#	16743#	16749#	16756#	16761#	16780#
16804#	16808#	16812#	16823#	16860	16861	16862	16863	17023	17024	17060	17061	17097
17098	17132	17133	17170	17171	17263	17264	17271	17272	17285	17286	17328	17329
17403	17404	17423	17424	17431	17432	17441	17442	17522	17546	17932#	17988#	18131#
18138#	18139#	18140#	18141#	18144#	18146#	18148#	18150#	18152#	18154#	18156#	18158#	18206#
18209#	18215#	18218#	18222#	18237#	18242#	18255#	18258#	18267#	18447#	18503#	18510#	18537#
18567#	18573#	18579#	18591#	18595#	18623#	18625#						
.SX	000300	1013#	1018									

. ABS. 076006 000

ERRORS DETECTED: 0

DE1:CJKDHA,DE2:CJKDHA/SOL/CRF:SYM/NL:TOC=DE1:SPMACJ.M11/ML,DE1:CJKDHA.P11
RUN-TIME: 571 595 35 SECONDS
RUN-TIME RATIO: 4497/1203-3.7
CORE USED: 70K (139 PAGES)