



DATA GENERAL
CORPORATION

Southboro,
Massachusetts 01772
(617) 485-9100

PROGRAM

High Speed Reader/Punch Test

TAPES

Binary 095-000014-06

ABSTRACT

High Speed Reader/Punch Test is a maintenance program designed to test the Type 4012A high speed punch and the Type 6013 or 4011B high speed readers. Tapes may be punched or read on the Teletype if one of the high speed devices does not exist in the user's system.

TTO BAUDRATE : 110 CR

0001 .MAIN

```
01  
02  
03  
04  
05  
06  
07  
08  
09 *****  
10 ;  
11 ; NAME1 HSRP,SR PART NUMBER: 004-000012  
12 ;  
13 ; DESCRIPTION: HIGH SPEED READER/PUNCH TEST  
14 ;  
15 ; REVISION HISTORY:  
16 ;  
17 ; REV. DATE  
18 ;  
19 ; 00 07/01/69  
20 ; 01 04/07/70  
21 ; 02 12/00/70  
22 ; 03 11/02/72  
23 ; 04 11/29/72  
24 ; 05 04/09/73  
25 ; 06 03/01/74  
26 ;  
27 ;  
28 ; COPYRIGHT (C) DATA GENERAL CORPORATION, 1969, 1970, 1972,  
29 ; 1973, 1974  
30 ; ALL RIGHTS RESERVED.  
31 *****
```

0002 .MAIN

```
01  
02 ; HIGH SPEED READER/PUNCH TEST  
03 ; REV 006,0  
04  
05 ;  
06  
07 ; ***** AUTO-RUN AUTO LOAD MODIFIED 2/25/72  
08  
09 ;  
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 ;  
49 ;
```

11. ABSTRACT
HIGH SPEED READER PUNCH TEST IS A MAINTENANCE PROGRAM DESIGNED TO TEST THE TYPE 4012A HIGH SPEED PUNCH AND THE TYPE 6013 OR 4011B HIGH SPEED READERS. TAPES MAY BE PUNCHED OF READ ON THE TELETYPE IF ONE OF THE HIGH SPEED DEVICES DOES NOT EXIST IN THE USERS SYSTEM. IF A CRT DISPLAY IS IN USE BOTH THE HIGH SPEED READER AND PUNCH MUST BE IN THE SYSTEM FOR A MEANINGFULL TEST.

12. MACHINE REQUIREMENTS
12.1 NOVA/SUPERNOVA FAMILY PROCESSOR
12.2 2K READ/WRITE MEMORY, MINIMUM
12.3 TELETYPE OR CRT FOR CONSOLE I/O
12.4 OPTIONAL EQUIPMENT
12.4.1 TYPE 4012A PUNCH AND CONTROL
12.4.2 TYPE 4011A HIGH SPEED READER
12.4.3 TYPE 4011B HIGH SPEED READER

13. SWITCH SETTINGS
13.1 STARTING ADDRESS
13.1.1 40=START PUNCH DIAGNOSTIC
13.1.2 41=START READER DIAGNOSTIC
13.1.3 42=START PUNCH FROM SWITCHES
13.1.4 43=START PUNCH COUNT ROUTINE
13.1.5 44=START READ COUNT ROUTINE
13.1.6 45=START PUNCH ALTERNATE ROUTINE
13.1.7 46=START READ ALTERNATE ROUTINE
13.1.8 47=START PUNCH FLOATING ZERO
13.1.9 50=START READ FLOATING ZERO
13.2 SWITCH 0(1)=INHIBIT DELAYS WHEN PUNCHING
SWITCH 0(1)=PROCEED FROM ERROR IN DIAGNOSTICS.
13.3 SWITCH 1(1)=INHIBIT PRINTING IN THE DIAGNOSTIC
13.4 SWITCH 2(1)=PRINT FAILURE RATE
13.5 SWITCH 8=15=DATA FOR THE PUNCH SWITCHS ROUTINE

IF THE HS READER AND PUNCH ARE IN THE SYSTEM AND THEIR DEVICE CODES ARE NOT 12, AND 13; LOAD AC0 WITH THE EVEN DEVICE CODE AND START AT LOC 4. THE PROGRAM WILL MODIFY ITSELF AND HALT. SET SWITCHS TO DESIRED STARTING ADDRESS PRESS RESET AND START.

A 0003 .MAIN

```
01
02 14. OPERATING PROCEEDURE
03 14.1 LOAD THE PROGRAM VIA THE BINARY LOADER
04 14.2 TESTING THE PAPER TAPE PUNCH
05 14.2.1 SET THE SWITCHS TO 40
06 14.2.2 PRESS START TO EXECUTE THE DIAGNOSTIC
07 14.2.3 ALLOW THE PUNCH DIAGNOSTIC TO RUN FOR
08 | AT LEAST ONE PASS, THE PROGRAM WILL
09 | TYPE A PASS MSG AT THE COMPLETION
10 | OF EACH PASS.
11 14.2.4 PRESS RESET
12 14.2.5 SET THE SWITCHES TO 43
13 14.2.6 PRESS START TO EXECUTE THE PUNCH
14 | COUNT PROGRAM.
15 14.2.7 ALLOW THE PUNCH TO PUNCH FOR AT LEAST
16 | 10 MINUTES.
17 14.2.8 PRESS STOP AND REMOVE THE TAPE
18 | FROM THE PUNCH.
19 14.2.9 INSERT THE TAPE JUST PUNCHED INTO THE
20 | HIGH SPEED READER, THE TELETYPE READER
21 | MAY BE USED IF THE SYSTEM DOES NOT
22 | CONTAIN A HIGH SPEED READER.
23 14.2.10 POSITION THE TAPE SUCH THAT THE COUNT
24 | PATTERN IS OVER THE PHOTO DIODES.
25 14.2.11 CLOSE THE READER LEVER
26 14.2.12 SET THE SWITCHES TO 44
27 14.2.13 PRESS START TO CHECK THE TAPE.
28 14.2.14 THE PUNCH SHOULD ALSO BE TESTED WITH
29 | THE EXERCISER PROGRAM.
30 14.3 TESTING THE HIGH SPEED READER
31 14.3.1 LOAD THE READER WITH A CONTINUES LOOP
32 | OF TAPE WITH ALTERNATING CHARACTORS
33 | OF ALL ONES AND ALL ZEROS(A TAPE MAY
34 | BE OBTAINED BY STARTING AT LOCATION 45).
35 14.3.2 CLOSE THE READER LEVER
36 14.3.3 SET THE SWITCHES TO 41
37 14.3.4 PRESS START TO EXECUTE THE DIAGNOSTIC
38 14.3.5 ALLOW THE DIAGNOSTIC TO RUN FOR AT LEAST
39 | ONE PASS, THE PROGRAM WILL TYPE A
40 | PASS MSG AT THE END OF EACH PASS.
41 14.3.6 PRESS RESET
42 14.3.7 SET THE SWITCHES TO 46
43 14.3.8 PRESS START TO CHECK THE TAPE
44 14.3.9 THE OPERATOR SHOULD MARK THE TAPE LOOP
45 | AT THE SPLICE. ERRORS OCCURING AT THIS
46 | SPLICE SHOULD BE IGNORED.
47 14.3.10 ALLOW THE READER TO CHECK THE TAPE FOR
48 | AT LEAST 5 MINUTES.
49 14.3.11 THE READER SHOULD ALSO BE CHECKED
50 | WITH THE EXERCISER PROGRAM.
```

A 0004 .MAIN

```
01
02 15. PROGRAM OUTPUT/ERROR DISCRPTION
03 15.1 ERRORS DETECTED VIA THE DIAGNOSTICS.
04 15.1.1 IF A MALFUNCTION IS DETECTED THE PROG-
05 | RAM WILL HALT AT LOCATION "AUTOER*6".
06 | THE OPERATOR MAY CHANGE SWITCH SETTINGS
07 | AND EXAMINE ACS AT THIS TIME IF
08 | DESIRED. IF SWITCHES 0 AND 1 ARE ZERO
09 | PRESSING CONTINUE WILL CAUSE A PRINT-
10 | OUT OF THE ERROR LOCATION. THE ROUTINE
11 | WILL ENTER A LOOP SUITABLE FOR SCOPING.
12 15.1.2 WHEN THE PROGRAM IS IN A SCOPE LOOP,
13 | SETTING SWITCH 2(1) WILL CAUSE THE
14 | FAILURE RATE TO BE PRINTED, SETTING
15 | SWITCH 0(1) WILL CAUSE THE PROGRAM TO
16 | PROCEED TO THE NEXT TEST.
17 15.2 DATA ERRORS DETECTED DURING THE READ ROUTINES
18 | CAUSE A PRINTOUT OF GOOD AND BAD DATA.
19
20 16. PROGRAM DISCRPTION/THEORY OF OPERATION
21 16.1 THE READER AND PUNCH DIAGNOSTICS CONSIST OF
22 | NUMERIOUS SMALL ROUTINES, EACH ROUTINE BEGINS
23 | WITH A INITIALIZING ROUTINE (SETUP) AND ENDS
24 | WITH A ITERATION ROUTINE (LOOP). BOTH THE
25 | SETUP AND LOOP ROUTINES ISSUE A I/O RESET
26 | PULSE. SYNCING ON I/O RESET WILL PERMIT THE
27 | LOGIC UNDER TEST TO BE OBSERVED. AT THE END
28 | OF EACH PASS OF THE DIAGNOSTIC A PASS MSG
29 | WILL BE TYPED. THE DIAGNOSTIC ROUTINES
30 | SHOULD BE EXECUTED PRIOR TO OTHER READER,
31 | PUNCH TEST.
32 16.2 THE PUNCH FROM SWITCHS ROUTINE IS PROVIDED
33 | SUCH THAT THE OPERATOR MAY SCOPE THE VARIOUS
34 | SIGNALS IN THE PUNCH AND PUNCH CONTROL.
35 16.3 THE PUNCH ALTERNATE ROUTINE IS PROVIDED TO
36 | PRODUCE A TAPE FOR THE READER DIAGNOSTIC.
37 16.4 THE PUNCH COUNT ROUTINE IS THE PRIME TEST
38 | OF THE PUNCHS ABILITY TO PUNCH DATA, A COUNT
39 | PATTERN IS USED SO DOUBLE PUNCHING AND FAILURE
40 | TO PUNCH MAY BE OBSERVED.
41 16.5 THE READ COUNT IS PROVIDED TO CHECK THE TAPE
42 | PRODUCED BY THE PUNCH COUNT ROUTINE.
43 16.6 THE READ ALTERNATE ROUTINE IS PROVIDED TO
44 | CHECK THE READERS ABILITY TO READ CHARACTORS
45 | OF ALL ONES AND ALL ZEROS UNDER VARIOUS
46 | START STOP CONDITIONS.
47
48 17. RESTRICTIONS/MISC
49 | IF THE HIGH SPEED READER OR HIGH SPEED
50 | PUNCH DOES NOT EXIST IN THE SYSTEM THE PROG-
51 | RAM WILL USE THE TELETYPE READER OR PUNCH)
52 | IF THEY ARE IN THE SYSTEM. PARTIAL TESTING
53 | MAY BE ACCOMPLISHED IF THE SYSTEM CONTAINS
54 | EITHER A PUNCH OR READER AND A CRT DISPLAY,
55 | BUT THIS IS NOT RECOMMENDED. THE DEVICES USED
56 | ARE A FUNCTION OF THE LOGIC EXISTING NOT
57 | WHETHER THE DEVICE ITSELF EXISTS.
```

```

A 0005 .MAIN
01
02      000001      .LOC 1
03 00001 100000    100000
04 00002 002003    JMP 0,+1
05 00003 000101    HERE
06 00004 040112    STA 0,XSAV
07 00005 000177    JMP START
08
09      002040      .LOC 40
10 00040 002001    JMP 0,+11          IPUNCH DIAGNOSTIC
11 00041 002052    JMP 0,+11          IREADER DIAGNOSTIC
12 00042 002053    JMP 0,+11          IPUNCH FROM SWITCHS
13 00043 002054    JMP 0,+11          IPUNCH COUNTER
14 00044 002055    JMP 0,+11          IREAD COUNTER
15 00045 002056    JMP 0,+11          IALTERNATE PUNCH
16 00046 002057    JMP 0,+11          IALTERNATE READ
17 00047 002060    JMP 0,+11          IPUNCH FLOATING ZERO.
18 00050 002061    JMP 0,+11          IREAD FLOATING ZERO.
19 00051 000753    ARA
20 00052 001366    RAR
21 00053 004404    SWPUN
22 00054 000402    CTRPUN
23 00055 000447    CTRMED
24 00056 000400    ALTPUN
25 00057 000472    ALTRED
26 00060 000666    PFLT
27 00061 000711    RFLT
28
29 00062 000267    CIDIV:  IGTV
30 00063 000327    DELI:   XDEL
31 00064 000323    DEL11: XDEL1
32 00065 000344    TIME:   DELAY
33 00066 002562    WHO:    PTIME
34 00067 077400    C0774:  077400
35 00070 000000    CALIBR: 0          JTIME FOR 100 MS
36 00071 000000    CAL10:  0          JTIME FOR 1 MS
37 00072 000000    LEOR:   0
38 00073 000304    WAIT:   WAITY
39 00074 002101    SETUP:  FNTER
40 00075 002122    LOOP:   CYCLE
41 00076 002173    ER:     ERR
42      000076    EHALT=JSR 0ER
43 00077 000004    C4:     4
44 00100 000006    C6:     6
45 00101 000325    W100:   X=100
46 00102 000000    TIMEX:  0
47 00103 000326    WAIT5:  X=AIT5
48 00104 000401    C401:   401
49 00105 000144    W144:   144
50 00106 000000    PUNDAT: 0
51 00107 000000    PASS:   0
52 00110 070477    CREADS: READS 2
53

```

```

A 0006 .MAIN
01
02 00111 160077    CIOT:   160077
03 00112 000012    XSAV:   12
04 00113 060012    DVCD:   060012
05 00114 000012    DVCD:   12
06 00115 050012    C60K:   050012
07 00116 000324    FIRST:  WAITY
08 00117 002100    LAST:   ENDIT
09 00120 000013    XX13:   13
10 00121 000012    CPTR:   12
11 00122 000000    DEVRET: 0
12 00123 000560    IINIT:  INIT
13 00124 012435    C5405:  5405.
14 00125 000066    C54:    54.
15 00126 015530    C7K:    7000.
16 00127 040011    C1630:  16303.
17 00130 000243    C16:    163.
18 00131 000024    IPUT:   PBT
19 00132 000017    K17:    17
20 00133 000645    IGET:   GET
21 00134 000017    IRDER:  RDERR
22 00135 000666    IPFLT:  PFLT
23 00136 000711    IRFLT:  RFLT
24 00137 000177    C77X:   177
25 00140 000000    RZCTR:  0
26 00141 001000    C1000:  1000
27 00142 000001    LENGTH: 1
28 00143 123456    TRAN:   123456
29 00144 000000    PUNRET: 0
30 00145 000000    RDAT:   0
31 00146 000000    GOOD:   0
32 00147 000000    BAD:    0
33 00150 000377    X377:   377
34 00151 000000    RER:    0
35 00152 002415    XCRLF:  CRLF
36 00153 002255    XMESS:  MESS
37 00154 002303    IZOCT:  ZOCT
38 00155 001400    C1400:  1400
39 00156 000000    HSPREAD: 0
40 00157 000000    HSPUNI: 0
41 00160 000012    C12:    12
42
43      000013    PTP.=13
44      000012    PTR.=12
45
46 00161 030045    HERE:   LDA 2,45
47 00162 025000    LDA 1,0,2
48 00163 125005    MOV 1,1,SNR
49 00164 002106    JMP 0,+2
50 00165 125005    MOV 1,1,SNR
51 00166 000753    ARA
52 00167 025001    LDA 1,1,2
53 00170 020160    LDA 0,C12
54 00171 106415    SUP4 0,1,SNR
55 00172 002166    JMP 0,-4
56 00173 044112    STA 1,XSAV
57 00174 000177    JMP START

```

```

A 0007 ,MAIN
01
02 00175 000000 XORDEV: 0
03 00176 000040 K40: 40
04
05 00177 030112 START: LDA 2,XSAV
06 00200 151400 INC 2,2
07 00201 024115 LDA 1,C60K
08 00202 125400 INC 1,1
09 00203 044113 STA 1,COVCD
10 00204 050120 STA 2,XX13
11 00205 050114 STA 2,DVCD
12 00206 004231 JSR DEVCD
13 00207 024115 LDA 1,C60K
14 00210 044113 STA 1,COVCD
15 00211 030112 LDA 2,XSAV
16 00212 050121 STA 2,CPTR,
17 00213 050114 STA 2,DVCD
18 00214 004231 JSR DEVCD
19 00215 024322 LDA 1,M100
20 00216 020115 LDA 0,C60K
21 00217 123400 AND 1,0
22 00220 024114 LDA 1,DVCD
23 00221 123000 AND 1,0
24 00222 040115 STA 0,C60K
25 00223 030045 LDA 2,45
26 00224 025000 LDA 1,0,2
27 00225 125005 MOV 1,1,SNR
28 00226 063077 HALT
29 00227 002230 JMP 0,+1
30 00230 000753 ARR
31
32 00231 054122 DEVC1: STA 3,DEVKET
33 00232 030116 LDA 2,FIRST
34 00233 021000 LDA 0,0,2
35 00234 024111 LDA 1,C10T
36 00235 123400 AND 1,0
37 00236 024113 LDA 1,COVCD
38 00237 106415 SUB# 0,1,SNR
39 00240 000246 JMP DEVC2
40
41 00241 151400 DEVC1: INC 2,2
42 00242 020117 LDA 0,LAST
43 00243 112414 SUB# 0,2,SZR
44 00244 000233 JMP DEVC2+2
45 00245 002122 JMP DEVRET
46 00246 021000 DEVC2: LDA 0,0,2
47 00247 024322 LDA 1,M100
48 00250 123400 AND 1,0
49 00251 024114 LDA 1,DVCD
50 00252 123200 ADD 1,0
51 00253 041000 STA 0,0,2
52 00254 000241 JMP DEVC1
53

```

```

A 0008 ,MAIN
31 00255 122460 MULTI: SURC 0,0 ;MULTIPLY C(1)+C(2)
32 00256 054303 STA 3,MSAV ;RESULT TO C(0),C(1)
33 00257 034302 LDA 3,MOCTR
34 00258 125203 MLOOP: MOVR 1,1,SNR
35 00259 101201 MOVR 0,0,SKP
36 00262 143220 ADDZR 2,0
37 00263 175404 INC 3,3,SZR
38 00264 000260 JMP MLOOP
39 00265 125200 MOVCR 1,1
40 00266 002303 JMP 0MSAV
41 00267 102400 IOIV: SUB 0,0 ;DIVIDE C(0),C(1)/C(2)
42 00270 054303 STA 3,MSAV ;C(0)=REMAINDER
43 00271 034302 LDA 3,MOCTR ;C(1)=QUOTIENT
44 00272 125120 MOVZL 1,1
45 00273 101100 DLOOP: MOVL 0,0
46 00274 142412 SUB# 2,0,SZR
47 00275 142400 SUR 2,0
48 00276 125100 MOVL 1,1
49 00277 175404 INC 3,3,SZR
50 00300 000273 JMP DLOOP
51 00301 002303 JMP 0MSAV
52 00302 177700 MOCTR: =20
53 00303 000000 MSAV: 0
54
55 00304 024322 WAITX: LDA 1,M100 ;WAIT ABOUT 6 SECONDS
56 00305 020070 LDA 0,CALIBR ;FOR UNTILL BUSY FLAG
57 00306 100420 NEGZ 0,0 ;BECOMES ZERO.
58 00307 063413 SKPBN PTP ;IF TIME OUT SET
59 00310 000316 JMP WAITY ;ITERATION COUNTER TO
60 00311 101403 INC 0,0,SNR ;+1,IF NO TIME OUT
61 00312 000307 JMP 0,-3 ;SET COUNT TO 100
62 00313 125404 INC 1,1,SZR ;DECIMAL
63 00314 000305 JMP WAITX+1
64 00315 102521 SUBZL 0,0,SKP
65 00316 020105 WAITY: LDA 0,M144
66 00317 042321 STA 0,0,+2
67 00320 001400 JMP 0,3
68 00321 002111 ITR
69 00322 177700 M100: =100
70
71 00323 030071 XDEL1: LDA 2,CAL10 ;C(1) IS DELAY IN MS
72 00324 000330 JMP XDEL+1
73 00325 126521 XW100: SUBZL 1,1,SKP ;DELAY IS 100MS
74 00326 024343 XWAIT5: LDA 1,CD50 ;DELAY IS 5 SEC
75 00327 030070 XDEL: LDA 2,CALIBR ;X(1) DELAY 100MS
76 00330 125113 MOVVL# 1,1,SNR
77 00331 124400 NEG 1,1
78 00332 140420 NEGZ 2,0
79 00333 101000 MOV 0,0
80 00334 101405 INC 0,0,SNR
81 00335 000340 JMP 0,+3
82 00336 063400 SKPBN 0
83 00337 000333 JMP 0,-4
84 00340 125404 INC 1,1,SZR
85 00341 000332 JMP XDEL+3
86 00342 001403 JMP 0,3
87 00343 000062 CD50: 50.

```

```

A 0009 .MAIN
01
02 00344 021400 DELAY: LDA 0,0,3      )TIME THE INSTRUCTION
03 00345 040353      STA 0,+6      )FOLLOWING THE CALL
04 00346 102040      ADCO 0,0
05 00347 040102      STA 0,TIMEY
06 00350 101000      MOV 0,0
07 00351 101402      INC 0,0,SZC
08 00352 001401      JMP 1,3
09 00353 000000      0
10 00354 000350      JMP ,-4
11 00355 040102      STA 0,TIMEY
12 00356 001401      JMP 1,3
13

```

```

A 0010 .MAIN
01
02 00400      .LOC 400
03
04 00400 020426 ALTPUN: LDA 0,CJPUN      )PUNCH ALTERNATE ONES/ZEROS
05 00401 101001      MOV 0,0,SKP
06 00402 020104 CTRPUN: LDA 0,C401      )PUNCH A COUNTER
07 00403 101001      MOV 0,0,SKP
08 00404 020110 SWPUN: LDA 0,CREADS      )PUNCH SWITCHES
09 00405 040410      STA 0,SWPN1
10 00406 152000      ADC 2,2
11 00407 050106      STA 2,PUNDAT
12 00410 004552      JSR INIT      )INITIALIZE
13 00411 004416      JSR PUNDEL      )PUNCH DELAY
14 00412 030106      LDA 2,PUNDAT
15 00413 151400      INC 2,2
16 00414 050106      STA 2,PUNDAT
17 00415 000000 SWPN1: 0
18 00416 0006131      JSP #IPUT      )PUNCH THE CHARACTER
19 00417 014142      DSZ LENGTH
20 00420 000772      JMP SWPN1-3
21 00421 000770      JMP SWPN1-4
22 00422 151223 ALTER: MOVZR 2,2,SNC      )DATA GENERATOR FOR
23 00423 152401      SUB 2,2,SKP      )ALTERNATE
24 00424 152000      ADC 2,2
25 00425 000771      JMP SWPN1+1
26 00426 000405 CJPUN: JMP .+5
27
28
29 00427 054144 PUNDEL: STA 3,PUNRET      )PUNCH DELAY LOGIC
30 00430 074477      READS 3
31 00431 175113      MOVL# 3,3,SNC      )IF SWITCH 0(1),NO DELAY.
32 00432 000400      JMP .+4
33 00433 170520      SUBZL 3,3
34 00434 054142      STA 3,LENGTH      )SET LENGTH TO 1 CHAR
35 00435 022144      JMP #PUNRET
36

```

A 0011 .MAIN

```
01
02 00436 004476 JSR RDWT
03 00437 000143 LDA 0,TRAN
04 00440 001113 MOVLM 0,0,SNR
05 00441 002144 JMP 0PUNRET JNO FURTHER DELAY
06 00442 024137 LDA 1,C77X
07 00443 107405 AND 0,1,SNR
08 00444 125400 INC 1,1
09 00445 006063 JSR 0DEL JDELAY IN 100MS STEPS
10 00446 002144 JMP 0PUNRET
11
12
13 00447 004513 CTRR0: JSR INIT
14 00450 004575 JSR GET JREAD A COUNTER
15 00451 040145 CTRR1: STA 0,RDAT JSYNC
16 00452 004573 JSR GET
17 00453 101005 MOV 0,0,SNR
18 00454 000775 JMP CTRR1
19 00455 024150 LDA 1,X377
20 00456 030145 LDA 2,RDAT
21 00457 151400 INC 2,2
22 00460 050145 STA 2,RDAT
23 00461 147400 AND 2,1
24 00462 106414 SUB# 0,1,SZR JCHECK THE COUNT READ
25 00463 000405 JMP CTRR2 JERROR
26 00464 014142 DSZ LENGTH
27 00465 000765 JMP CTRR1+1
28 00466 004446 JSR RDWT JDELAY
29 00467 000763 JMP CTRR1+1
30
31 00470 004427 CTRR2: JSR 0DERR JREADER ERROR
32 00471 000757 JMP CTRR1-1 JRESYNC THE READER
33
34 00472 004470 ALTRED: JSR INIT JREAD ALTERNATE ONES/ZEROS
35 00473 101005 MOV 0,0,SKP
36 00474 004423 JSR 0DERR
37 00475 004550 JSR GET JGET A CHARACTER
38 00476 126000 ADC 1,1
39 00477 107000 ADC 0,1
40 00500 107404 AND 0,1,SZR
41 00501 126000 ADC 1,1
42 00502 044145 STA 1,RDAT JSYNC THE READER
43
44 00503 004431 ALTR1: JSR RDWT JDELAY LOGIC
45 00504 004541 JSR GET JGET CHARACTER
46 00505 024150 LDA 1,X377
47 00506 030145 LDA 2,RDAT
48 00507 150000 COM 2,2
49 00510 050145 STA 2,RDAT
50 00511 147400 AND 2,1
51 00512 106414 SUB# 0,1,SZR JCHECK DATA
52 00513 000761 JMP ALTRED+2 JERROR DETECTED
53 00514 014142 DSZ LENGTH
54 00515 000767 JMP ALTR1+1
55 00516 000765 JMP ALTR1
56
```

A 0012 .MAIN

```
01
02 00517 040147 0DERR: STA 0,BAD JERROR PRINTER
03 00520 044146 STA 1,GOOD
04 00521 054151 STA 3,RER
05 00522 006152 JSR 0XCRLF JPRINT CARRIAGE
06 00523 006153 JSR 0XMESS
07 00524 001355 GOODEQ
08 00525 024146 LDA 1,GOOD
09 00526 006154 JSR 0IZOCT JGOOD DATA PRINTER
10 00527 006153 JSR 0XMESS
11 00530 001360 RADEQ
12 00531 024147 LDA 1,BAD
13 00532 006154 JSR 0IZOCT
14 00533 002151 JMP 0RER
15
16 00534 054151 RDWT: STA 3,RER JRANDOM NUMBER DELAY
17 00535 024143 LDA 1,TRAN
18 00536 000067 LDA 0,C0774
19 00537 135120 MOVZL 1,3
20 00540 175120 MOVZL 3,3
21 00541 137000 ADD 1,3
22 00542 024776 LDA 1,-2
23 00543 137000 ADD 1,3
24 00544 054143 STA 3,TRAN
25 00545 163400 AND 3,0
26 00546 024155 LDA 1,C1400
27 00547 106432 SUBZ# 0,1,SZR
28 00550 137000 ADD 1,3
29 00551 030157 LDA 2,C77X JLOW ORDER BITS
30 00552 173405 AND 3,2,SNR
31 00553 151400 INC 2,2
32 00554 050142 STA 2,LENGTH JSET BLOCK LENGTH
33 00555 024067 LDA 1,C0774
34 00556 167705 ANDS 3,1,SNR
35 00557 000756 JMP RDWT+1
36 00560 006064 JSR 0DEL1 JTIME DELAY
37 00561 002151 JMP 0RER JEXIT
38
39
```

A 0013 .MAIN

```
01
02 00562 002677 INIT: TORST
03 00563 000112 NIOS PTR. INITIALIZE FOR HIGH
04 00564 003412 SKPBN PTR. FOR LOW SPEED READER
05 00565 102401 SUB 0,0,SKP PUNCH.
06 00566 102000 ADC 0,0
07 00567 000156 STA 0,HSREAD
08 00570 004432 STA 3,INTEM
09 00571 024070 LUA 1,CALIBR 1100 MS RPT COUNT
10 00572 125005 MOV 1,1,SNR ISKP IF ALREADY DONE
11 00573 006000 JSR 0,0
12 00574 125004 MOV 1,1, SZR
13 00575 000414 JMP OUT
14 00576 000152 JSR 0,XCRLF
15 00577 006153 JSR 0,XMESS
16 00600 002547 CANNOT ICANNOT IDENTIFY CPU
17 00601 000045 LDA 2,45
18 00602 021000 LDA 0,0,2
19 00603 101004 MOV 0,0, SZR
20 00604 000403 JMP ,+3
21 00605 003077 HALT
22 00606 000777 JMP ,-1
23 00607 005004 LDA 3,4,2
24 00610 001400 JMP 0,3
25 00611 044070 OUT: STA 1,CALIBR
26 00612 000411 LDA 2,CD100
27 00613 000062 JSR 0,CIDIV
28 00614 044071 STA 1,CALIBR
29
30 00615 020141 LDA 0,C1000
31 00616 040140 STA 0,RZCTR
32 00617 102000 SUBZR 0,0
33 00620 040157 STA 0,HSPUN
34 00621 002401 JMP 0,INTEM
35 00622 000000 INTEM: 0
36 00623 000144 CD100: 100.
37
38
```

A 0014 .MAIN

```
01
02 00624 024157 PUT: LDA 1,HSPUN PUNCH A CHARACTER
03 00625 124524 NEGZL 1,1, SZR
04 00626 000405 JMP ,+5
05 00627 005113 DDAS 1,PTP.
06 00630 005513 SKPBZ PTP.
07 00631 126000 ADC 1,1
08 00632 044157 STA 1,HSPUN
09 00633 125225 MOVZR 1,1,SNR ISEE IF PUNCH IN SYSTEM.
10 00634 000405 JMP PUT1 FOR HS PUNCH IF ITS IN
11 00635 005513 SKPBZ PTP. THE SYSTEM
12 00636 000777 JMP ,-1
13 00637 071113 DDAS 2,PTP.
14 00640 001400 JMP 0,3
15 00641 005511 PUT1: SKPBZ TTP
16 00642 000777 JMP ,-1
17 00643 071111 DDAS 2,TTP
18 00644 001400 JMP 0,3
19
20 00645 020156 GET: LDA 0,HSREAD GET A CHARACTER
21 00646 101005 MOV 0,0,SNR FROM HS READER IF ITS
22 00647 000405 JMP GET1 IN THE SYSTEM
23 00650 005512 SKPBZ PTP.
24 00651 000777 JMP ,-1
25 00652 000512 DIAS 0,PTR.
26 00653 000404 JMP GET2
27 00654 005510 GET1: SKPBZ TTP
28 00655 000777 JMP ,-1
29 00656 000510 DIAS 0,TTP
30
31 00657 024141 GET2: LDA 1,C1000
32 00658 101004 MOV 0,0, SZR
33 00661 044140 STA 1,RZCTR
34 00662 001410 DSZ RZCTR
35 00663 001400 JMP 0,3
36 00664 003077 HALT
37 00665 001400 JMP 0,3 ITO MUCH LEADER?
```



```

A 0015 .MAIN
01 00665 036123 PFLT1 JSR #IIINT      /PUNCH A FLOATING
02 00667 020150 LDA 0,X377 /ZERO PATTERN.
03 00670 040462 STA 0,FLTCT
04 00671 152400 SUB 2,2
05 00672 006131 JSR #IPUT /PUNCH LEADER.
06 00673 014457 DSZ FLTCT
07 00674 000775 JMP ,-3
08
09 00675 010455 PFLT1: ISZ FLTCT /FLOATING COUNTER
10 00676 101000 MOV 0,0
11 00677 030453 LDA 2,FLTCT
12 00700 004403 JSR FLOT
13 00701 006131 JSR #IPUT /PUNCH
14 00702 000775 JMP PFLT1
15
16 00703 020132 FLOT: LDA 0,K17 /TRANSLATE COUNT TO
17 00704 113400 AND 0,2 /FLOATING NUMBER.
18 00705 020424 LDA 0,CTABZ
19 00706 113000 ADD 0,2
20 00707 031000 LDA 2,0,2
21 00710 001400 JMP 0,3
22
23 00711 006123 RFLT1: JSR #IIINT /READ THE FLOATING
24 00712 006133 JSR #IGET /ZERO PATTERN.
25 00713 101005 MOV 0,0,SNR /TAPE MUST START IN
26 00714 000776 JMP ,-2 /LEADER.
27 00715 152520 SUBZL 2,2
28 00716 050434 STA 2,FLTCT
29
30 00717 010433 RFLT1: ISZ FLTCT
31 00720 101000 MOV 0,0
32 00721 030431 LDA 2,FLTCT
33 00722 004761 JSR FLOT /GET FLOATING NUMBER.
34 00723 006133 JSR #IGET /GET CHARACTER.
35 00724 145000 MOV 2,1
36 00725 106415 SUB# 0,1,SNR /CHECK IF 0,K.
37 00726 000771 JMP RFLT1
38 00727 006134 JSR #IPDER /ERROR
39 00730 000767 JMP RFLT1
40 00731 000732 CTABZ: .+1
41 00732 000177 TABZ: 177
42 00733 000277 277
43 00734 000337 337
44 00735 000357 357
45 00736 000367 367
46 00737 000373 373
47 00740 000375 375
48 00741 000376 376
49 00742 000376 376
50 00743 000375 375
51 00744 000373 373
52 00745 000367 367
53 00746 000357 357
54 00747 000337 337
55 00750 000277 277
56 00751 000177 177
57 00752 000000 FLTCT: 0

```

```

A 0016 .MAIN
01
02 00753 004607 A00: JSR INIT
03 00754 006074 JSR #SETUP /THE PUNCH BUSY FLAG
04 00755 063513 SKPBZ PTP. /SHOULD BE ZERO. CHECK
05 00756 006075 EHALT /SELB LINE(A02).
06 00757 006075 JSR #LOOP
07
08 00760 006074 A02: JSR #SETUP /THE PUNCH DONE FLAG
09 00761 063713 SKPDZ PTP. /SHOULD BE ZERO. CHECK
10 00762 006075 EHALT /SELB LINE (A00).
11 00763 006075 JSR #LOOP
12
13 00764 020175 LDA 0,K40
14 00765 040175 STA 0,XORDEV
15 00766 006074 A04: JSR #SETUP /CHECK DEVICE SELECTION
16 00767 060113 NJOS PTP. /SHOULD NOT AFFECT PTP.
17 00770 006065 JSR #TIME
18 00771 063513 SKPBZ PTP.
19 00772 060113 NJOS PTP.
20 00773 020417 LDA 0,CNIOC /CNIOC PTP.
21 00774 024175 LDA 1,XORDEV
22 00775 131000 MOV 1,2
23 00776 113520 ANDZL 0,2
24 00777 107000 ADD 0,1
25 01000 140400 SUB 2,1
26 01001 044401 STA 1,.-+1
27 01002 000000 0
28 01003 063413 SKPBN PTP.
29 01004 006075 EHALT
30 01005 006075 JSR #LOOP
31 01006 020175 LDA 0,XORDEV
32 01007 101224 MOVZB 0,0,5ZB
33 01010 000755 JMP A04-1
34 01011 000402 JMP .+2
35 01012 060213 CNIOC: NIOC PTP.
36
37 01013 102400 A17: SUB 0,0
38 01014 051013 DOA 0,PTP.

```

▲ 0017 .MAIN

```
01
02 01015 006074 A18: JSR #SETUP      IWITH THE DONE FLAG
03 01016 102520      SUBZL 0,0        IZERO, CHECK THAT
04 01017 005477      INTA 1          IINTA DOES NOT READ
05 01020 107414      AND# 0,1,SZM   IBACK BIT 15, CHECK
06 01021 006076      EHALT          I(R66) AT INT ACK
07 01022 006075      JSR #LOOP      ITIME, E71 4-5-6,E57-13
08
09 01023 006074 A20: JSR #SETUP      IWITH THE DONE FLAG
10 01024 005477      INTA 1          IZERO, CHECK THAT
11 01025 125004      MOV 1,1,SZR   IINTA DOES NOT READ
12 01026 006076      EHALT          IBACK ANY BITS.
13 01027 006075      JSR #LOOP
14
15 01030 006074 A21: JSR #SETUP      IFA TEST TO INSURE THAT
16 01031 000013      NIO PTP.      IPUN SELECT WITHOUT A
17 01032 003513      SKPBZ PTP.    ISTART PULSE DOES NOT
18 01033 006076      EHALT          ISET BUSY.
19 01034 006075      JSR #LOOP
20
21 01035 006074 A24: JSR #SETUP      ISEND A START PULSE
22 01036 000113      NIOS PTP.     IAND SEE IF BUSY SETS.
23 01037 003413      SKPBN PTP.    ICHECK THE NOT
24 01040 006076      EHALT          IPUN START LEVEL TO
25 01041 006073      JSR #WAIT     ITHE SET SIDE OF BUSY.
26 01042 006075      JSR #LOOP     IALSO SELB(A02) LEVEL.
27
28 01043 006074 A26: JSR #SETUP      ISTART THE PUNCH
29 01044 000113      NIOS PTP.     ITHEN DO A NO
30 01045 000013      NIO PTP.      IOPERATION, IF BUSY
31 01046 003413      SKPBN PTP.    IIS ZERO,PERHAPS THE
32 01047 006076      EHALT          ICLEAR LINE IS OPEN,
33 01050 006073      JSR #WAIT     ICHECK E59 8-9-10
34 01051 006075      JSR #LOOP
35
36 01052 006074 A28: JSR #SETUP      ISTART THE PUNCH
37 01053 000113      NIOS PTP.     ITHEN PRODUCE A CLEAR
38 01054 000200      NIOC 0        IPULSE TO A DEVICE
39 01055 003413      SKPBN PTP.    IOTHER THAN PTP, CHECK
40 01056 006076      EHALT          IAND GATE OF (LLEAR,
41 01057 006073      JSR #WAIT     IPUN SELECT).
42 01060 006075      JSR #LOOP
43
44 01061 006074 A30: JSR #SETUP      ISET THE BUSY FLAG
45 01062 000113      NIOS PTP.     ITHEN TRY TO CLEAR IT.
46 01063 000213      NIOC PTP.     ICHECK E58 3-4
47 01064 003513      SKPBZ PTP.    IE59 4-6,E59 8-9-10 AND
48 01065 006076      EHALT          ITHE CLEAR SIDE OF THE
49 01066 000113      NIOS PTP.     IBUSY FLOP.
50 01067 006073      JSR #WAIT
51 01070 006075      JSR #LOOP
```

▲ 0018 .MAIN

```
01
02 01071 006074 A32: JSR #SETUP      ISET THE BUSY FLOP
03 01072 000113      NIOS PTP.     ITHEN TRY TO CLEAR IT
04 01073 002677      IORST         IVIA I/O RESET, CHECK
05 01074 003513      SKPBZ PTP.    IE59 5-6,E58 8-9,
06 01075 006076      EHALT          ITHE I/O RESET PATH
07 01076 000113      NIOS PTP.     ITO CLEAR PUN BUSY.
08 01077 006073      JSR #WAIT
09 01100 006075      JSR #LOOP
10
11 01101 006074 A34: JSR #SETUP      IFA CHECK TO INSURE
12 01102 000113      NIOS PTP.     II/O RESET WILL CLEAR
13 01103 006073      JSR #WAIT     ITHE DONE FLAG.
14 01104 002677      IORST         ICHECK E58 5-6,
15 01105 003713      SKPDZ PTP.    IE60 5-6,TO THE
16 01106 006076      EHALT          ICLEAR SIDE OF THE
17 01107 006075      JSR #LOOP     IDONE FLAG.
18
19 01114 006074 A36: JSR #SETUP      ICHECK THAT A START
20 01111 000113      NIOS PTP.     IPULSE WILL CLEAR
21 01112 006073      JSR #WAIT     ITHE DONE FLAG.
22 01113 000113      NIOS PTP.     ICHECK E60 1-2-6,
23 01114 003713      SKPDZ PTP.    ITO THE CLEAR SIDE
24 01115 006076      EHALT          IOF DONE.
25 01116 006073      JSR #WAIT
26 01117 006075      JSR #LOOP
27
28 01120 006074 A38: JSR #SETUP      IFA TEST TO INSURE
29 01121 000113      NIOS PTP.     ITHAT THE DONE FLOP
30 01122 006073      JSR #WAIT     IIS RESET BY A
31 01123 000213      NIOC PTP.     ICLEAR PULSE.
32 01124 003713      SKPDZ PTP.    ICHECK OR GATE E60
33 01125 006076      EHALT          ICLEAR SIDE OF THE
34 01126 006075      JSR #LOOP     IDONE.
35
36 01127 006074 A40: JSR #SETUP      ITHE PUN COMPLETE
37 01130 000113      NIOS PTP.     ILEVEL FAILED TO
38 01131 006073      JSR #WAIT     ISET DONE OR CLEAR
39 01132 003613      SKPBN PTP.    IBUSY.
40 01133 003513      SKPBZ PTP.    ICHECK 4,5 MS DELAY
41 01134 101011      MOV# 0,0,SKP IAND ITS INPUTS
42 01135 006076      EHALT
43 01136 006075      JSR #LOOP
44
45 01137 006074 A42: JSR #SETUP      ITHE PUN COMPLETE
46 01140 000113      NIOS PTP.     ILEVEL FAILED TO
47 01141 006073      JSR #WAIT     ICLEAR BUSY, CHECK
48 01142 003513      SKPBZ PTP.    ITHE CLOCK INPUT
49 01143 006076      EHALT          ITO BUSY, THE FLOP
50 01144 006075      JSR #LOOP     IMAY HAVE FAILED.
51
52 01145 006074 A44: JSR #SETUP      ITRY TO SET DONE VIA
53 01146 000113      NIOS PTP.     IPUN COMPLETE. CHECK
54 01147 006073      JSR #WAIT     IPUN DONE FLOP,CLOCK
55 01150 003613      SKPBN PTP.    IDATA AND CLEAR INPUTS.
56 01151 006076      EHALT          IALSO CHECK SELD(A80),
57 01152 006075      JSR #LOOP     ITHE AND OF SELECT, DONE.
```

```

A 0019 ,MAIN
31
02 01153 006074 A481 JSR #SETUP ;CHECK THE SET TO
SUBZR 0,0 ;PUN INT REQ FLOP,
STA 0,1 ;WITH DONE (1) AND
NIOB PTP. ;PUN INT DISABLE
JSR #WAIT ;HOPEFULLY(0). SET
NIOB CPU ;INTERRUPT ENABLE IN CPU
MOV 0,0 ;IF ITS CLEARED INTERRUPT
SKPBZ CPU ;OCCURED, CHECK INTR
EHALT ;(B29) ETC,ETC.
JSR #LOOP
12
13 01165 006074 A481 JSR #SETUP ;WILL PUN INT DISABLE
NIOB PTP. ;PREVENT PUN INT REQ
LDA 0,C4 ;FROM SETTING? NO.
MSKO 0 ;CHECK AND OF (DONE(1)),
JSR #WAIT ;PUN INT DISABLE(0) TO
INTA 1 ;DATA TERMINAL OF PUN
MOV# 1,1,SZR ;INT REQ. ALSO PUN
EHALT ;INT DISABLE FLOP,AND
JSR #LOOP ;ITS INPUTS.
22
23 01176 006074 A501 JSR #SETUP ;A UNKOWN INTERRUPT
NIOB CPU ;OCCURED. CHECK OC GATE
MOV 0,0 ;TO INTR (B29).
SKPBN CPU
EHALT
JSR #LOOP
29
30 01204 006074 A521 JSR #SETUP ;A CHECK TO INSURE
NIOB PTP. ;THAT I/O RESET
JSR #WAIT ;WILL CLEAR PUN
NIOB CPU
DIO 0,CPU ;RESET PULSE
SKPBN CPU
EHALT
JSR #LOOP
38
39 01214 006074 A541 JSR #SETUP ;A MASK OUT INSTRUCTION
NIOB CPU ;WITHOUT A BIT 13
ADC 0,0
MSKO 0
SUB 0,0 ;SHOULD NOT SET
MSKO 0 ;PUN INT DISABLE
NIOB PTP. ;FLOP, CHECK "0"
JSR #WAIT ;INPUT TO FLOP.
NIOB CPU
MOV 0,0
SKPBZ CPU
EHALT
JSR #LOOP
51

```

```

A 0020 ,MAIN
31
42 01231 006074 A561 JSR #SETUP ;CHECK TO INSURE
LDA 0,C4 ;THAT I/O RESET
MSKO 0 ;WILL CLEAR THE
IDRST ;PUN INT DISABLE
NIOB PTP. ;FLOP.
JSR #WAIT
NIOB CPU ;ENABLE INTERRUPTS
MOV 0,0
SKPBZ CPU
EHALT
JSR #LOOP
12
13
14 01244 006074 A581 JSR #SETUP ;WITH PUN INT REQ(1)
NIOB PTP. ;SEE IF INTA READS
JSR #WAIT ;BACK ANY BITS.
INTA 1 ;CHECK AND OF
MOV# 1,1,SNR ;(INT ACK,PUN INT REQ(1))
EHALT ;AND THE FOLLOWING
JSR #LOOP ;INVERTER. ETC,ETC.
21
22 01253 006074 A601 JSR #SETUP ;CHECK THE DATA 15
NIOB PTP. ;LINE ON INTA.
JSR #WAIT ;CHECK GATE E71 4-5-6
SUBZL 0,0
INTA 1
AND# 0,1,SNR
EHALT
JSR #LOOP
30
31 01263 006074 A621 JSR #SETUP ;CHECK THE DATA 14
NIOB PTP. ;LINE ON INTA.
JSR #WAIT ;CHECK GATE E71 1-2-3.
SUBZL 0,0
MOVZL 0,0
INTA 1
AND# 0,1,SNR
EHALT
JSR #LOOP
39
40
41 01274 006074 A641 JSR #SETUP ;CHECK FOR INTA
NIOB PTP. ;CODE 13 (FOR PTP)
JSR #WAIT
LDA 0,XX13
INTA 1
SUB# 0,1,SZR
EHALT
JSR #LOOP
49
50 01304 006074 A661 JSR #SETUP ;THE PUN DONE FLAG
NIOB PTP. ;WAS SET VIA PUN
NIOB CPU ;COMPLEAT EVEN
JSR #X100 ;WITH BUSY ZERO.
SKPDZ PTP.
EHALT
JSR #LOOP
56

```

```

A 0021 .MAIN
01
02 01313 006074 A60: JSR #SETUP      I/CHECK THAT 6
03 01314 006113      NIOS PTR.      I/CHARACTORS ARE
04 01315 006073      JSR #WAIT      I/PUNCHED IN 100MS
05 01316 006113      NIOS PTR.      I/OF TIME.
06 01317 006065      JSR #TIME
07 01320 003513      SKPBZ PTR.
08 01321 024070      LDA 1,CALIBR
09 01322 030102      LDA 2,TIMEY
10 01323 006062      JSR #CIDIV
11 01324 020100      LDA 0,C6
12 01325 106433      SUBZM 0,1,SNC
13 01326 006076      EHALT
14 01327 006075      JSR #LOOP
15
16 01330 006103 A70: JSR #WAIT5     I/WAIT 5 SECONDS
17 01331 006113      NIOS PTR.      I/FOR THE PUNCH TO
18 01332 006073      JSR #WAIT      I/STOP.
19 01333 006113      NIOS PTR.
20 01334 006065      JSR #TIME
21 01335 003513      SKPBZ PTR.      I/CHECK FOR 6 CHARACTORS
22 01336 024070      LDA 1,CALIBR   I/IN 100 MS OF TIME.
23 01337 030102      LDA 2,TIMEY
24 01340 006062      JSR #CIDIV
25 01341 020100      LDA 0,C6
26 01342 106433      SUBZM 0,1,SNC
27 01343 006074      EHALT
28
29
30 01344 006152      JSR #XCRLF
31 01345 006153      JSR #XMESS
32 01346 001363      PASSE          I/PASS MSG
33 01347 030045      LDA 2,45
34 01350 025000      LDA 1,0,2
35 01351 125005      MOV 1,1,SNC
36 01352 002402      JMP #,+2
37 01353 000413      JMP R00
38 01354 000753      A00
39
40      GOODEQ: .TXT IGOOD=1
    01355 047507
    01356 042117
    01357 000075
41      BADEQ: .TYT I BAD=1
    01360 041040
    01361 042101
    01362 000075
42      PASSE: .TXTE IPASSI
    01363 040520
    01364 051523
    01365 000000

```

```

A 0022 .MAIN
01
02 01366 006123 R00: JSR #IIINT
03 01367 006274      JSR #SETUP
04 01370 063512      SKPBZ PTR.
05 01371 006076      EHALT
06 01372 006075      JSR #LOOP
07
08 01373 006074 R02: JSR #SETUP
09 01374 063712      SKPDZ PTR.
10 01375 006076      EHALT
11 01376 006075      JSR #LOOP
12
13 01377 006074 R04: JSR #SETUP
14 01400 000100      NIOS 0
15 01401 063512      SKPBZ PTR.
16 01402 006076      EHALT
17 01403 006075      JSR #LOOP
18
19 01404 006074 R06: JSR #SETUP
20 01405 063012      NIO PTR.
21 01406 063512      SKPBZ PTR.
22 01407 006074      EHALT
23 01410 006075      JSR #LOOP
24
25 01411 006074 R08: JSR #SETUP
26 01412 020522      LDA 0,C20
27 01413 062077      MSKO 0
28 01414 102620      SUBZR 0,0
29 01415 000177      NIOS CPU
30 01416 040001      STA 0,1
31 01417 063477      SKPBN CPU
32 01420 006076      EHALT
33 01421 006075      JSR #LOOP
34
35 01422 006074 R10: JSR #SETUP
36 01423 102620      SUBZR 0,0
37 01424 000177      NIOS CPU
38 01425 040001      STA 0,1
39 01426 063477      SKPBN CPU
40 01427 006076      EHALT
41 01430 006075      JSR #LOOP
42
43 01431 006074 R11: JSR #SETUP
44 01432 060112      NIOS PTR.
45 01433 006065      JSR #TIME
46 01434 063512      SKPBZ PTR.
47 01435 101002      MOV 0,0,SZC
48 01436 006076      EHALT
49 01437 006075      JSR #LOOP
50
51 01440 006074 R12: JSR #SETUP
52 01441 060112      NIOS PTR.
53 01442 063412      SKPBN PTR.
54 01443 006076      EHALT
55 01444 006065      JSR #TIME
56 01445 063512      SKPBZ PTR.
57 01446 006075      JSR #LOOP

```

```

I/PAPER TAPE READER
I/BUSY FLAG SHOULD BE 0.
I/CHECK SELB LINE(A82).
I/ETC,ETC.

I/PTR DONE FLAG
I/SHOULD BE 0.
I/CHECK SELD LINE(A80)
I/ETC.

I/A START PULSE TO A
I/DEVICE NOT THE PTR
I/SET PTR BUSY! CHECK
I/AND GATE OF (START,
I/RD SELECT).

I/SELECTING THE PTR
I/WITHOUT A START
I/PULSE SET THE BUSY
I/FLOP. CHECK AND OF
I/(START, RD SELECT).

I/TURN ON THE
I/INTERRUPT SYSTEM.
I/CHECK FOR NO INTERRUPT.
I/BUT!
I/INTERRUPT OCCURED.
I/CHECK INTR LINE(B29).
I/AND RD INT RQ FLOP.

I/TURN ON THE INTERRUPT.
I/CHECK FOR NO INTERRUPT.
I/INTERRUPT OCCURED.
I/CHECK E82=0 THE
I/DONE (1) INPUT TO AND
I/OF (RD INT DISABLE(0),
I/RD DONE(1)).

I/BUSY FAILED TO
I/CLEAR. PAPER IN READER?
I/IS LEVER DOWN?
I/SWITCH IN RUN POSITION?
I/CHECK SELB(A82) AND
I/RD STROBE INPUT TO BUSY.

I/TRY TO SET THE
I/BUSY FLOP VIA
I/START.CHECK AND GATE
I/OF(START,RD SELECT)
I/AND RD BUSY.

```

```

A 0023 .MAIN
01 01447 006074 R14: JSR #SETUP      JSET READER BUSY FLAG
02 01450 050112      NIOS PTR,          ITHEN TRY TO CLEAR VIA
03 01451 062677      IORST             I/O RESET, CHECK BUSY
04 01452 063512      SKPBZ PTR.        IFLOP THE CLEAR INPUT,
05 01453 006076      EHALT             IE46-13,ETC.
06 01454 006075      JSR #LOOP
07
08 01455 006074 R16: JSR #SETUP      JSET READER BUSY FLAG.
09 01456 050112      NIOS PTR,          ITHEN TRY TO RESET VIA
10 01457 060212      NIOC PTR.         IRD SELECT AND A
11 01460 063512      SKPBZ PTR.        ICLEAR PULSE.
12 01461 006076      EHALT
13 01462 006075      JSR #LOOP
14
15 01463 006074 R18: JSR #SETUP      JSET READER BUSY FLOP.
16 01464 060112      NIOS PTR,          ITHEN CHECK THAT A
17 01465 060200      NIOC 0            ICLEAR PULSE TO A DEVICE
18 01466 053412      SKPBN PTR.        IOTHER THAN THE READER
19 01467 006076      EHALT             IWILL NOT CLEAR READER
20 01470 006065      JSR #TIME
21 01471 063512      SKPBZ PTR.
22 01472 006075      JSR #LOOP
23
24 01473 006074 R20: JSR #SETUP      JSET READER BUSY FLOP.
25 01474 060112      NIOS PTR,          ITHEN CHECK THAT
26 01475 060012      NIO PTR.          IRD SELECT WITHOUT A
27 01476 053412      SKPBN PTR.        ICLEAR PULSE WILL NOT
28 01477 006076      EHALT             ICLEAR READER BUSY.
29 01500 006065      JSR #TIME
30 01501 063512      SKPBZ PTR.
31 01502 006075      JSR #LOOP
32
33 01503 020176      LDA 0,K40
34 01504 040175      STA 0,XDRDEV
35 01505 006074 R22: JSR #SETUP      ICHECK DEVICE SELECTION
36 01506 060112      NIOS PTR,
37 01507 006065      JSR #TIME
38 01510 063512      SKPBZ PTR.
39 01511 060112      NIOS PTR,
40 01512 020421      LDA 0,CCNIO
41 01513 024175      LDA 1,XORDEV
42 01514 131000      MOV 1,2
43 01515 113520      ANDZL 0,2
44 01516 107000      ADD 0,1
45 01517 146400      SUB 2,1
46 01520 044401      STA 1,.*1
47 01521 000000      0
48 01522 063412      SKPBN PTR.
49 01523 006075      EHALT
50 01524 006065      JSR #TIME
51 01525 063512      SKPBZ PTR.
52 01526 006075      JSR #LOOP
53 01527 020175      LDA 0,XORDEV
54 01530 101224      MOVZR 0,0,SZR
55 01531 000753      JMP R22-1
56 01532 000403      JMP .*3
57 01533 060212 CCNIO: NIOC PTR.
58 01534 000020 C20: 20

```

```

A 0024 .MAIN
01
02 01535 006074 R35: JSR #SETUP      I/O RESET FAILED
03 01536 050112      NIOS PTR,          ITO CLEAR THE
04 01537 006065      JSR #TIME         IRD DONE FLAG.
05 01540 063512      SKPBZ PTR.
06 01541 062677      IORST
07 01542 063712      SKPDZ PTR.
08 01543 006076      EHALT
09 01544 006075      JSR #LOOP
10
11 01545 006074 R38: JSR #SETUP      IA CLEAR PULSE
12 01546 050112      NIOS PTR,          ITO THE PAPER TAPE
13 01547 006065      JSR #TIME         IREADER FAILED TO
14 01550 063512      SKPBZ PTR.        ICLEAR RD DONE.
15 01551 060212      NIOC PTR.
16 01552 063712      SKPDZ PTR.
17 01553 006075      EHALT
18 01554 006075      JSR #LOOP
19
20 01555 006074 R40: JSR #SETUP      IA START PULSE
21 01556 050112      NIOS PTR,          ITO THE PAPER TAPE
22 01557 006065      JSR #TIME         IREADER FAILED TO
23 01560 063512      SKPBZ PTR.        ICLEAR RD DONE.
24 01561 060112      NIOS PTR.
25 01562 063712      SKPDZ PTR.
26 01563 006076      EHALT
27 01564 006075      JSR #LOOP

```

```

A 0025 .MAIN
01
02 01565 006074 R42: JSR #SETUP      ;THE RD DONE FLOP
03 01566 060112      NIOS PTR.      ;FAILED TO SET.
04 01567 006065      JSR #TIME      ;CHECK THE CLOCK,
05 01570 063512      SKPBZ PTR.     ;DATA,AND RESET
06 01571 063512      SKPUN PTR.     ;INPUTS, ALSO CHECK
07 01572 006076      EHALT      ;THE SELD LINE(A80).
08 01573 006075      JSR #LOOP
09
10 01574 006074 R44: JSR #SETUP      ;SETTING THE READER
11 01575 102020      SUBZR R,0      ;DONE FLAG SHOULD
12 01576 062077      MSKO 0       ;CAUSE A INTERRUPT.
13 01577 062677      IORST      ;CHECK THE INTR LINE
14 01600 060177      NIOS CPU     ;(B29),RD INT REQ FLOP,
15 01601 040001      STA R,1      ;AND ITS INPUT GATES.
16 01602 060112      NIOS PTR.
17 01603 006065      JSR #TIME
18 01604 063512      SKPBZ PTR.
19 01605 063577      SKPBZ CPU
20 01606 006076      EHALT
21 01607 006075      JSR #LOOP
22
23 01610 006074 R46: JSR #SETUP      ;THE MSKO
24 01611 102000      ADC 0,0
25 01612 062077      MSKO 0
26 01613 102620      SUBZR R,0
27 01614 062077      MSKO 0
28 01615 060177      NIOS CPU     ;INSTRUCTION WITH
29 01616 040001      STA R,1      ;BIT 11 A ZERO SHOULD
30 01617 060112      NIOS PTR.     ;NOT PREVENT INTERRUPTS.
31 01620 006065      JSR #TIME     ;CHECK THE DATA 11
32 01621 063512      SKPBZ PTR.     ;INPUT TO THE
33 01622 063577      SKPBZ CPU     ;RD INTY DISABLE FLOP.
34 01623 006076      EHALT
35 01624 006075      JSR #LOOP
36
37 01625 006074 R48: JSR #SETUP      ;THE RD INT DISABLE
38 01626 020706      LDA R,C20     ;FLOP IS SET VIA
39 01627 062077      MSKO 0       ;MSKO, I/O RESET
40 01630 062677      IORST      ;SHOULD THEN CLEAR IT.
41 01631 060177      NIOS CPU     ;THE DONE FLAG FAILED.
42 01632 060112      NIOS PTR.     ;TO INTERRUPT,SUGGESTING
43 01633 006065      JSR #TIME     ;I/O RESET DID NOT CLEAR
44 01634 063512      SKPBZ PTR.     ;RD INT DISABLE.
45 01635 063577      SKPBZ CPU
46 01636 006076      EHALT
47 01637 006075      JSR #LOOP
48
49 01640 006074 R50: JSR #SETUP      ;USING MSKO WITH
50 01641 020673      LDA R,C20     ;BIT 11(1) SHOULD SET
51 01642 062077      MSKO 0       ;RD INT DISABLE AND
52 01643 060177      NIOS CPU     ;PREVENT THE DONE
53 01644 060112      NIOS PTR.     ;FLAG FROM INTERRUPTING.
54 01645 006065      JSR #TIME     ;A INTERRUPT OCCURED
55 01646 063512      SKPBZ PTR.     ;HOWEVER.
56 01647 063477      SKPBN CPU
57 01650 006076      EHALT
58 01651 006075      JSR #LOOP

```

```

A 0026 .MAIN
01
02 01652 006074 R52: JSR #SETUP      ;ANOTHER CHECK OF
03 01653 020661      LDA R,C20     ;RD INT DISABLE
04 01654 060177      NIOS CPU     ;THE CLOCK INPUT.
05 01655 064112      NIOS PTR.
06 01656 006065      JSR #TIME
07 01657 063512      SKPBZ PTR.
08 01658 063577      SKPBZ CPU
09 01661 006076      EHALT
10 01662 006075      JSR #LOOP
11
12 01663 006074 R54: JSR #SETUP      ;TEST I/O RESET
13 01664 060277      NIOC CPU     ;ABILITY TO CLEAR
14 01665 060112      NIOS PTR.     ;RD INT REQ FLOP.
15 01666 006065      JSR #TIME
16 01667 063512      SKPBZ PTR.
17 01670 060177      NIOS CPU
18 01671 062477      DIC R,CPU
19 01672 063477      SKPBN CPU
20 01673 006076      EHALT
21 01674 006075      JSR #LOOP
22
23 01675 006074 R56: JSR #SETUP      ;AFTER A I/O RESET
24 01676 062677      IORST      ;A INTA SHOULD READ
25 01677 061477      INTA 0       ;BACK ALL ZEROS.
26 01704 101004      MOV R,0,R,SZR
27 01701 006076      EHALT
28 01702 006075      JSR #LOOP
29
30 01703 006074 R58: JSR #SETUP      ;INTA FAILED TO
31 01704 062112      NIOS PTR.     ;READ BACK A BIT 14
32 01705 006065      JSR #TIME     ;WHEN THE DONE FLAG
33 01706 063512      SKPBZ PTR.     ;WAS SET.
34 01707 061477      INTA 0
35 01710 126520      SUBZL 1,1
36 01711 126120      MOVZL 1,1
37 01712 107415      ANDW 0,1,SNR
38 01713 006076      EHALT
39 01714 006075      JSR #LOOP
40
41 01715 006074 R60: JSR #SETUP      ;CHECK THE CODE
42 01716 064112      NIOS PTR.     ;READ BACK ON
43 01717 006065      JSR #TIME     ;INTA FROM THE
44 01720 063512      SKPBZ PTR.     ;READER.
45 01721 024121      LDA I,CPTR.
46 01722 061477      INTA 0
47 01723 106404      SUB 0,1,SZR
48 01724 006076      EHALT
49 01725 006075      JSR #LOOP

```

```

A 0027 ,MAIN
01
02 01720 006074 R62: JSR #SETUP      ;CHECK THAT SOME,
03 01727 152020      SUBZR 2,2      ;ANY, BITS ARE READ
04 01730 151225      MOVZR 2,2,SNR ;BACK IN A SET OF
05 01731 006075      EHALT          ;15 CHARACTORS.
06 01732 060112      NIOS PTR.
07 01733 063512      SKPBZ PTR.
08 01734 000777      JMP , -1
09 01735 060412      DIA 0,PTR.
10 01736 101005      MOV 0,0,SNR
11 01737 000771      JMP R62+2
12 01740 006075      JSR #LOOP
13
14 01741 006074 R64: JSR #SETUP      ;READ UNTILL A CHAR
15 01742 152020      SUBZR 2,2      ;NOT ZERO IS OBTAINED.
16 01743 151225      MOVZR 2,2,SNR ;IF DIA TO DEVICE 0
17 01744 060412      JMP R62A      ;READS THE SAME CHAR
18 01745 060112      NIOS PTR.
19 01746 063512      SKPBZ PTR.
20 01747 000777      JMP , -1
21 01750 060412      DIA 0,PTR.
22 01751 101005      MOV 0,0,SNR
23 01752 000773      JMP , -5
24 01753 064400      DIA 1,0
25 01754 106415      SUBR 0,1,SNR
26 01755 006076      EHALT
27 01756 006075 R62A: JSR #LOOP
28
29 01757 006074 R66: JSR #SETUP      ;CHECK THAT THE DATA
30 01760 176620      SUBZR 3,3      ;BUFFER HOLDS BOTH ZEROS
31 01761 060112      NIOS PTR.
32 01762 126000      ADC 1,1
33 01763 152000      ADC 2,2
34 01764 063612      SKPDN PTR.
35 01765 000777      JMP , -1
36 01766 060512      DIAS 0,PTR.
37 01767 113400      AND 0,2      ;LOOK FOR BITS STUCK(1)
38 01770 100000      COM 0,0
39 01771 107400      AND 0,1      ;LOOK FOR BITS STUCK(0)
40 01772 175224      MOVZR 3,3,SZR
41 01773 000771      JMP R66+5
42

```

```

A 0028 ,MAIN
01
02
03 01774 125242      MOVOR 1,1,SZC
04 01775 006074      EHALT          ;CH1 STUCK TO 0
05 01776 125202      MOVOR 1,1,SZC ;CH2 STUCK TO 0
06 01777 006076      EHALT
07 02000 125202      MOVOR 1,1,SZC ;CH3 STUCK TO 0
08 02001 006076      EHALT
09 02002 125202      MOVOR 1,1,SZC ;CH4 STUCK TO 0
10 02003 006076      EHALT
11 02004 125202      MOVOR 1,1,SZC ;CH5 STUCK TO 0
12 02005 006076      EHALT
13 02006 125202      MOVOR 1,1,SZC ;CH6 STUCK TO 0
14 02007 006076      EHALT
15 02010 125202      MOVOR 1,1,SZC ;CH7 STUCK TO 0
16 02011 006076      EHALT
17 02012 125202      MOVOR 1,1,SZC ;CH8 STUCK TO 0
18 02013 006076      EHALT
19
20 02014 151202      MOVOR 2,2,SZC ;CH1 STUCK TO 1
21 02015 006076      EHALT
22 02016 151202      MOVOR 2,2,SZC ;CH2 STUCK TO 1
23 02017 006076      EHALT
24 02020 151202      MOVOR 2,2,SZC ;CH3 STUCK TO 1
25 02021 006076      EHALT
26 02022 151202      MOVOR 2,2,SZC ;CH4 STUCK TO 1
27 02023 006076      EHALT
28 02024 151202      MOVOR 2,2,SZC ;CH5 STUCK TO 1
29 02025 006076      EHALT
30 02026 151202      MOVOR 2,2,SZC ;CH6 STUCK TO 1
31 02027 006076      EHALT
32 02030 151202      MOVOR 2,2,SZC ;CH7 STUCK TO 1
33 02031 006076      EHALT
34 02032 151202      MOVOR 2,2,SZC ;CH8 STUCK TO 1
35 02033 006076      EHALT
36 02034 006075      JSR #LOOP      ;ITERATE TEST.
37

```

```

A 0029 .MAIN
31
02 02035 006074 R68: JSR #SETUP      /THE READER DONE
03 02036 060112      NIOS PTR,      /FLAG WAS SET VIA
04 02037 060212      NIOC PTR,      /RD STROBE(1) WHEN
05 02040 006065      JSR #TIME      /THE BUSY FLAG WAS
06 02041 063012      SKPDN PTR,    /ZERO, CHECK THE
07 02042 101003      MOV 0,0,SNC   /DATA INPUT TO
08 02043 006076      EHALL        /RD DONE.
09 02044 020447      LDA 0,ESWIT   /RESTART PROGRAM AFTER
10 02045 101004      MOV 0,0,SZR
11 02046 000770      JMP R68*1    /FIXING THIS TROUBLE.
12 02047 000414      JMP R69
13          002056      .LOC 2056
14 02056 000000 EGGS: 0
15 02057 000000      0
16 02058 000000      0
17 02059 000000      0
18 02062 000000      0
19
20 02063 006152 R69: JSR #XCRLF
21 02064 006153      JSR #XMESS
22 02065 001303      PASSE
23 02066 030045      LDA 2,45     /PASS MSG
24 02067 025000      LDA 1,0,2
25 02070 125005      MOV 1,1,SNR
26 02071 002405      JMP 0,+5
27 02072 015003      DSZ 3,2
28 02073 002404      JMP 0,+4
29 02074 035004      LDA 3,4,2
30 02075 001400      JMP 0,3
31 02076 001306      R00
32 02077 000753      A00
33

```

```

A 0030 .MAIN
31
02 02100 000000 ENDIT: 0
33
04 02101 054420 ENTER: STA 3,LOOPR      /LOOP ITERATE RETURN
05 02102 034407      LDA 3,ITR      /THIS ROUTINE INITIALIZES
06 02103 054407      STA 3,ITRCT   /FEACH TEST
07 02104 176400      SUB 3,3
08 02105 054406      STA 3,ESWIT
09 02106 054406      STA 3,ERRCT
10 02107 062477      DIC 0,CPU     /I/O RESET
11 02110 002411      JMP #LOOPR
12
13 02111 000144 ITR: 144
14 02112 000000 ITRCT: 0
15 02113 000000 ESWIT: 0
16 02114 000000 ERRCT: 0
17 02115 000000 RETURN: 0
18 02116 000000 SAV2: 0
19 02117 000000 SAV1: 0
20 02120 000000 SAV0: 0
21 02121 000000 LOOPR: 0
22
23 02122 054773 CYCLE: STA 3,RETURN   /END OF TEST ITERATION
24 02123 050773      STA 2,SAV2   /ROUTINE
25 02124 044773      STA 1,SAV1   /SAVE THE ACS!
26 02125 040773      STA 0,SAV0
27 02126 014764      DSZ ITRCT
28 02127 000432      JMP CYCTS
29
30 02130 034761      LDA 3,ITR
31 02131 054761      STA 3,ITRCT
32 02132 074477      READS 3
33 02133 030760      LDA 2,ESWIT
34 02134 175120      MOVZL 3,3
35 02135 175100      MOVL 3,3
36 02136 151005      MOV 2,2,SNR
37 02137 000413      JMP NOEX
38 02140 175103      MOVL 3,3,SNC
39 02141 000415      JMP PCENT-1
40
41 02142 006506      JSR #ICRLF    /PRINT CARRIAGE
42 02143 024751      LDA 1,ERRCT
43 02144 006506      JSR #IPDEC   /PRINT VALUE
44 02145 020412      LDA 0,PCENT  /EXAMPLE: 89X
45 02146 006500      JSR #ICHR
46 02147 063611      SKPDN TIO
47 02150 000777      JMP 0,-1
48 02151 000405      JMP PCENT-1
49
50 02152 020746 NOEX: LDA 0,SAV0   /NORMAL EXIT,NO ERR
51 02153 024744      LDA 1,SAV1
52 02154 030742      LDA 2,SAV2
53 02155 002740      JMP #RETURN
54

```



```

A 0031 .MAIN
01 02155 102401 SUB 0,0,SKP
02 02157 000245 PCENT: 245
03 02160 040734 STA 0,ERRCT /CHARACTER
04 02161 020737 CYCTS: LDA 0,SAV0 /RESET ERROR COUNT
05 02162 024735 LDA 1,SAV1 /RESTORE ACS
06 02163 030733 LDA 2,SAV2
07 02164 034727 LDA 3,ESWIT
08 02165 175004 MOV 3,3,SZR
09 02166 074477 READS 3
10 02167 062477 DIC 0,CPU /I/O RESET
11 02174 175113 MOVLM 3,3,SNC /SWITCH 0
12 02171 002730 JMP *LOOPR /I(1)=LOOP ROUTINE
13 02172 002723 JMP #RETURN /I(0)=PROCEED TO NEXT TEST
14
15
16 02173 054722 ERR: STA 3,RETURN /ERROR SUBROUTINE
17 02174 050722 STA 2,SAV2
18 02175 044722 STA 1,SAV1
19 02176 040722 STA 0,SAV0
20
21 02177 034714 LDA 3,ESWIT
22 02200 175005 MOV 3,3,SNR
23 02201 000407 JMP ERR
24 02202 030714 ERET: LDA 2,SAV2 /RESTORE ACS
25 02203 024714 LDA 1,SAV1
26 02204 020714 LDA 0,SAV0
27 02205 010707 ISZ ERCT /ICOUNT
28 02206 062477 DIC 0,CPU /ERRORS,I/O RESET
29 02207 002706 JMP #RETURN /EXIT
30
31 02210 034705 ERR1: LDA 3,RETURN /ERROR, C(3)=PC
32 02211 004407 JSR AUTOER /OPERATOR,SET SWITCHS:
33 02212 054701 STA 3,ESWIT
34 02213 074477 READS 3
35 02214 175100 MOVLM 3,3
36 02215 175113 MOVLM 3,3,SNC /LOOK AT SWITCH 1
37 02216 004410 JSR EPRINT /PRINT ERROR DATA
38 02217 000763 JMP ERET
39
40 02220 054413 AUTOER: STA 3,AA03
41 02221 024635 LDA 1,EGGS
42 02222 125004 MOV 1,1,SZR
43 02223 020405 JMP ,+5
44 02224 024673 LDA 1,SAV1
45 02225 034670 LDA 3,RETURN
46 02226 063077 HALT
47 02227 002404 JMP #,AA03
48 02230 004404 JSR EPRINT
49 02231 034631 LDA 3,EGGS+4
50 02232 001400 JMP 0,3
51 02233 000000 .AA03: 0

```

```

A 0032 .MAIN
01
02 02234 054657 EPRINT: STA 3,ESWIT /ERROR MESSAGE PRINTER
03 02235 006413 JSR #ICRLF /PRINT CARRIAGE
04 02236 006407 JSR #IMESS /AND HEADER
05 02237 002253 HEADER
06 02240 020655 LDA 0,RETURN
07 02241 125000 ADD 1,1
08 02242 107000 ADD 0,1
09 02243 006406 JSR #IPOCT /PC OF ERROR
10 02244 002647 JMP #ESWIT /RETURN TO CALL
11
12 02245 002255 IMESS: MESS
13 02246 002370 ICHAR: CHAR
14 02247 002427 ITYPE: TYPE
15 02250 002415 ICRLF: CRLF
16 02251 002304 IPOCT: POCT
17 02252 002310 IPDEC: PUEC
18
19 HEADER: .TXT 1
20 02253 041520 PC 1
02254 000011

```

```

A 0033 .MAIN
01 TELETYPE NON INTERRUPT PACKAGE
02 JAC1,AC2 SAVED
03 JMESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLER
04 JCRLF" PRINTS A CARRIAGE RETURN
05 JPOCT" PRINTS C(1) IN OCTAL
06 JZOC" PRINTS C(1) IN OCTAL, LEADING ZEROS SUPPRESSED
07 JPDEC" PRINTS C(1) IN DECIMAL, LEADING ZEROS SUPPRESSED,
08 JTHE ABOVE THREE ARE FOLLOWED BY THE TAB IN P.TAB
09 J"TI"ND" ACCEPTS OCTAL, AND
10 J"TI"ND" ACCEPTS DECIMAL SINGLE PRECISION SIGNED INTEGERS
11 JINTO AC1 FROM THE TTI. LEADING NULLS, TABS,
12 JAND SPACES ARE IGNORED. A 16 BIT UNSIGNED INTEGER IS
13 JFORMED, THEN NEGATED IF A MINUS SIGN IS TYPED.
14 JEXIT AT CALL+1 IF INPUT ERROR WITH ACC#BAD CHARACTER.
15 J (NOT A LEGAL DIGIT OR TERMINATING CHARACTER)
16 JEXIT AT CALL+2 UPON TERMINATING CHARACTER
17 J WITH ACC#0, 0, 40, 12, 55
18 J FOR NULL, TAB, SPACE, CARRIAGE RETURN, COMMA
19 JTHE ABOVE WAIT FOR TTD DONE, THEN CLEAR TTD.
20 J"CHAR" PRINTS ASCII CHARACTER IN C(0)R; C(0)L MUST BE 0.
21 JEXITS CALL +2 IF C(0)R#0, CORRECTS THE PARITY,
22 JSIMULATES TAB ON AS033.
23 J"TYPE" PRINTS C(0)R. MUST HAVE PROPER PARITY. EXITS AT
24 JCALL+1. REPLACE "TYPE" WITH INTERRUPT TYPE IF DESIRED.
25
26 02255 054551 MESS1 STA 3,MESSR JPRINT A TEXT MESSAGE
27 02256 044505 STA 1,P,AC1
28 02257 050505 STA 2,P,AC2
29 02260 010546 ISZ MESSR
30 02261 031400 LDA 2,3 JC(2) POINTS TO MESSAGE
31 02262 024505 LDA 1,P,377 JA 8 BIT MASK
32 02263 021000 LDA 0,0,2 JC(2)=DATA WORD
33 02264 125112 MOVLM 1,1,SZC
34 02265 123701 ANDS 1,0,SKP
35 02266 123401 AND 1,0,SKP JC(0)=DATA CHARACTER RIGHT
36 02267 151400 INC 2,2 JINC TO NEXT WORD
37 02270 124000 COM 1,1 JFLIP MASK
38 02271 004477 JSR CHAR JPRINT
39 02272 000771 JMP MESS+6 JANOTHER
40 02273 000402 JMP +2
41 02274 004474 P.LST: JSR CHAR
42 02275 024466 PEXIT: LDA 1,P,AC1
43 02276 030466 LDA 2,P,AC2
44 02277 063511 SKPBZ TTD
45 02300 000777 JMP -1
46 02301 060211 NIQC TTD
47 02302 002524 JMP 0MESSR JLAST
48

```

```

A 0034 .MAIN
01 02303 102401 ZOCT: SUP 2,0,SKP
02 02304 020462 POCT: LDA 0,P,C00
03 02305 050457 STA 2,P,AC2
04 02306 030435 LDA 2,OCTAB JPRINT C(1) IN OCTAL
05 02307 000404 JMP +4
06 02310 050454 PDECT: STA 2,P,AC2
07 02311 030440 LDA 2,DECTA JPRINT C(1) IN DECIMAL
08 02312 102400 SUB 0,0
09 02313 054513 STA 3,MESSR JTBOTH ENTHYS PRINT NUMBER
10 02314 044447 STA 1,P,AC1
11 02315 040445 STA 0,ZSUPP JTHEN TAB TO NEXT POSITION
12 02316 050401 STA 2,+.1
13 02317 000000 DECOCT: 0 J"LOA 2, TABLE" INSTRUCTION
14 02320 010777 ISZ -1
15 02321 020444 LDA 0,P,TAB
16 02322 151025 MOV 2,2,SNR JIF TABLE ENTRY#0
17 02323 000752 JMP PEXIT JEXIT WITH NO TAB
18 02324 034436 LDA 3,ZSUPP JZEROS SUPPRESS STUF
19 02325 102400 SUB 0,0
20 02326 146492 DECOY: SUB0# 2,1,SZC
21 02327 000405 JMP DECP
22 02330 146400 SUB 2,1 JFORM THE DIGIT
23 02331 034435 LDA 3,P,C00
24 02332 101400 INC 0,0
25 02333 000773 JMP DECOY
26 02334 151235 DECP: MOVZRW 2,2,SNR
27 02335 034431 LDA 3,P,C00
28 02336 054424 STA 3,ZSUPP JC(0)=DIGIT
29 02337 163000 ADD 3,0 JMAKE ASCII
30 02340 175004 MOV 3,3,SZR
31 02341 004407 JSR CHAR JPRINT
32 02342 000755 JMP DECOCT JGET NEXT DIGIT
33
34 02343 030425 OCTAB: LDA 2,+.1+.=DECOCT
35 02344 100000 100000
36 02345 010000 100000
37 02346 001000 100000
38 02347 000100 100000
39 02350 000010 100000
40 02351 000001 100000
41 02352 000000 0
42 02353 030435 DECTA: LDA 2,+.1+.=DECOCT
43 000012 ,NDX 10
44 02354 023420 100000
45 02355 001750 100000
46 02356 000144 100000
47 02357 000012 100000
48 02350 000001 100000
49 02361 000000 0
50 000010 ,NDX 0
51 02362 000000 ZSUPP: 0
52 02363 000000 P,AC1: 0
53 02364 000000 P,AC2: 0
54 02365 000011 P,TAB: 11 JCHARACTER PRINTED AFTER NUMBERS
55 02366 000000 P,C00: 0
56 02367 000377 P,377: 377

```

```

A 0035 .MAIN
01 02370 054434 CHAR: STA 3,CHRET IPRINT C(0) RIGHT
02 02371 001305 MOV# 0,0,SNR IRETURN +2 IF NULL
03 02372 001401 JMP 1,3
04 02373 115120 MOVZL 0,3 ICOMPUTE EVEN PARITY
05 02374 177004 ADD 3,3,SZR
06 02375 000777 JMP ,=-1
07 02376 103200 ADDR 0,0
08 02377 101300 MOV# 0,0
09 02400 034452 CHAR:1: LDA 3,P,C11 IIS THIS A TAB
10 02401 116415 SUB# 0,3,SNR
11 02402 000403 JMP CHA,3 IYES
12 02403 004424 JSR TYPE INO PRINT IT
13 02404 002420 JMP #CHRET IEXIT
14 02405 000450 CHA,3: LDA 0,P,240 ISIMULATE A TAB
15 02406 004421 JSR TYPE IWITH 1 TO 7 SPACES
16 02407 000416 LDA 0,CHORZ
17 02410 034441 LDA 3,P,C7
18 02411 103404 AND 3,0,SZR
19 02412 000773 JMP CHA,3
20 02413 040412 STA 0,CHORZ
21 02414 002410 JMP #CHRET
22
23 02415 054411 CRLF: STA 3,MESSR ISAVE RETURN
24 02416 044745 STA 1,P,AC1
25 02417 050745 STA 2,P,AC2
26 02420 000434 LDA 0,P,C15
27 02421 004747 JSR CHAR IPRINT CARRIAGE AND LF
28 02422 023431 LDA 0,P,C12
29 02423 000601 JMP P,LST
30
31 02424 000000 CHRET: 0
32 02425 000000 CHORZ: 0
33 02426 000000 MESSR: 0

```

```

A 0036 .MAIN
01 02427 054430 TYPE: STA 3,TYPRET ITYPE THE C(0)R IF
02 02430 074477 READS 3 ISWITCH 1(0).
03 02431 177120 ADDZL 3,3,SZC
04 02432 000404 JMP ,+4 IINHIBIT TYPE EXIT.
05 02433 063511 SKPBZ TTD
06 02434 000777 JMP ,=-1
07 02435 061111 DDAS 0,TTD
08 02436 034731 LDA 3,P,377
09 02437 175220 MOVZR 3,3
10 02440 163400 AND 3,0
11 02441 116043 ADDC 0,3,SNC
12 02442 034414 LDA 3,P,C40
13 02443 102432 SUBZ# 3,0,SZC ISKIP NON-PRINTING CHAR
14 02444 010761 ISZ CHORZ
15 02445 034407 LDA 3,P,C15
16 02446 116445 SUBD 0,3,SNR
17 02447 054706 STA 3,CHORZ ICLR HORZ POS
18 02450 002407 JMP #TYPRET
19 02451 000007 P,C7: 7
20 02452 000011 P,C11: 11
21 02453 000012 P,C12: 12
22 02454 000015 P,C15: 15
23 02455 000020 P,240: 240
24 02456 000040 P,C40: 40
25 02457 000000 TYPRET: 0

```

```

A 0037 .MAIN
01 02460 020773 TINC.1 LDA 0,P.C10
02 02461 004746 JSR TYPE
03 02462 010744 TINC.1 TSZ MESSR
04 02463 024700 TINC.1 LDA 1,P.AC1
05 02464 034676 LDA 3,ZSUPP
06 02465 175102 MOVL 3,3,SZC
07 02466 124400 NEG 1,1
08 02467 000607 JMP PEXIT+1
09
10 02470 102121 TINC.1 ADCZL 0,0,3KP OCTAL ENTRY
11 02471 102400 TINC.1 SUBO 0,0 DECIMAL ENTRY
12 02472 054734 STA 3,MESSR
13 02473 050671 STA 2,P.AC2 FAC2 IS SAVED
14 02474 030757 LDA 2,P.C12
15 02475 113000 ADD 0,2
16 02476 102400 SUBO 0,0
17 02477 040663 STA 0,ZSUPP MINUS SIGN AND LEADING SPACES FLAG
18 02500 034662 TINC.1 LDA 3,ZSUPP
19 02501 175004 MOV 3,3,SZR
20 02502 000760 JMP TINC.
21 02503 054660 TINC.1 STA 3,P.AC1
22 02504 063610 SKPDN TTI
23 02505 000777 JMP .-1
24 02506 060610 OIAC 0,TTI
25 02507 004661 JSR CHAR
26 02510 034740 LDA 3,P.C40
27 02511 116414 SUBW 0,3,SZR
28 02512 101015 MOVW 0,0,SNR
29 02513 000765 JMP TINC. ISPACE, TAB, OR NULL
30 02514 024432 LDA 1,TIN2.
31 02515 106015 ADCW 0,1,SNR COMMA
32 02516 000744 JMP TINC.
33 02517 100424 SURZ 0,1,SZR MINUS
34 02520 000405 JMP TINC. INO
35 02521 034641 LDA 3,ZSUPP
36 02522 177200 ADDR 3,3 COMPLEMENT SIGN
37 02523 054607 STA 3,ZSUPP
38 02524 000760 JMP TINC.+1
39 02525 136415 TINC.1 SUBW 1,3,SNR IS IT A CARRIAGE RETURN?
40 02526 000732 JMP TINC.
41 02527 024416 TINC.1 LDA 1,TIN1.
42 02530 107022 ADDZ 0,1,SZC SKIP IF NOT A DIGIT
43 02531 146513 SUBLW 2,1,SNR SKIP IF DIGIT
44 02532 000731 JMP TINC.
45 02533 010627 TSZ ZSUPP OUT OF LEADING SPACES
46 02534 020627 LDA 0,P.AC1
47 02535 101120 MOVZL 0,0
48 02536 115120 MOVZL 0,3
49 02537 175120 MOVZL 3,3
50 02540 137000 ADD 1,3 IS OLD P.AC1'S + NEW DIGIT
51 02541 145220 MOVZR 2,1
52 02542 105232 MOVZRW 1,1,SZC SKIP IF OCTAL MODE
53 02543 117000 ADD 0,3 ADD 2 OLD P.AC1'S
54 02544 000737 JMP TINC.
55 02545 177720 TIN1.1 -60
56 02546 000055 TIN2.1 55
57 CANNOT.1 .TXTE ICANNOT IDENTIFY CPU 1
02547 040703
02550 047115

```

```

0038 .MAIN
02551 152317
02552 144640
02553 142504
02554 152116
02555 143311
02556 120131
02557 050303
02558 120125
02561 000240

```

```

A 0039 .MAIN
01 ;PROCESSOR TIMER PACKAGE
02 ;THIS PACKAGE IS CALLED WHENEVER IT IS NECESSARY TO
03 ;IDENTIFY THE MEAN TIME BASE OF THE COMPUTER IN
04 ;WHICH THE PROGRAM RESIDES. THE MEAN TIME BASE MAY
05 ;THEN BE UTILIZED TO VERIFY OR CALCULATE THE RELATIONSHIPS
06 ;OF OTHER PERIPHERAL FUNCTIONS.
07 ;
08 ;THE PACKAGE RETURNS TO THE CALL INSTRUCTION WITH
09 ;THE CONTENTS OF AC1* TO A CALIBRATION COUNT
10 ;THAT MAY BE INCREMENTED TO OVERFLOW IN 100 MILLI-
11 ;SECONDS BY THE FOLLOWING DELAY LOOP.
12 ;TYPE1: MOV 0,0
13 ;     INC 0,0,SZC     ;SKIP=NOT OVERFLOW
14 ;     JMP 1,3         ;EXIT LOOP
15 ;     0               ;ANY FLAVOR IO SKP
16 ;     JMP TYPE1
17 ;
18 ;THE CONTENTS OF AC2 WILL CONTAIN A SIMILAR 100 MS
19 ;ITERATION COUNT BUT FOR THE FOLLOWING LOOP:
20 ;TYPE2: NIO 0
21 ;     DIA 1,,DEV     ;GET DEVICE STATUS
22 ;     AND# 2,1,SZR   ;ANY STATUS COMPARE
23 ;     JMP ,+4        ;EXPECTED STATUS EXIT
24 ;     INC 0,0,SZR   ;SKIP OUT ON LOOP OFLOW
25 ;     JMP TYPE2
26 ;
27 ;THE VALUES RETURNED MAY BE ARITHMETICALLY
28 ;PROCESSED (MULTIPLIED/DIVIDED) FOR LONGER OR
29 ;SHORTER DELAYS AS LONG AS THE STANDARD LOOPS
30 ;LISTED ABOVE ARE UTILIZED.
31 ;
32 ;IT IS RECOMMENDED THAT ALL TIMING FUNCTIONS
33 ;BE PERFORMED WITHIN THE SAME GENERAL AREA OF
34 ;MEMORY AS THIS TIMING PACKAGE.

```

```

A 0040 .MAIN
01 ;
02 ;INITIALLY, THE TIMER PACKAGE ATTEMPTS TO
03 ;DETERMINE IF THE COMPUTER HAS A REAL TIME CLOCK
04 ;AVAILABLE. THE ITERATION COUNTS ARE DEVELOPED SIMPLY
05 ;BY SYNCING WITH THE CLOCK AND COUNTING
06 ;THE #OF LOOP ITERATIONS AT 10 HZ.
07 ;
08 ;IF, HOWEVER THERE IS NO REAL TIME CLOCK THE MEAN
09 ;TIME BASE OF THE LOOPS MUST BE CALCULATED. THIS
10 ;IS PERFORMED BY COUNTING THE #OF TIMES THE
11 ;STANDARD LOOPS ARE ITERATED FOR ONE OUTPUT
12 ;CHARACTER TO DEVICE "TTO" AND REQUESTING THE
13 ;BAUD RATE OF DEVICE TTD TO BE TYPED IN BY THE
14 ;TEST OPERATOR.
15 ;
16 02562 054557 PTIME: STA 3,SVTIME
17 02563 063514 SKPBZ RTC ;TEST FOR CAS/RTC
18 02564 000416 JMP SCORE ;CAS RTC NONEXIS.
19 02565 060114 NIOS RTC ;TURN CLOCK ON
20 02566 063514 SKPBZ RTC ;BUSY #1 IS RTC
21 02567 000403 JMP ,+3
22 02570 063514 SKPDN RTC ;DONE #0 NO RTC
23 02571 000411 JMP SCORE ;AND DEV TTD IS USED
24 02572 062677 TORST
25 02573 102520 SUBZL 0,0 ;#10 HZ FOR RTC
26 02574 004473 JSR TYME
27 02575 061114 DOAS 0,RTC ;PASSED TO "TYME"
28 02576 063514 SKPBZ RTC ;FOR EXECUTION
29 02577 002542 JMP #SVTIME ;AC1 AND AC2=LOOP COUNTS
30 ;
31 ;THE FOLLOWING SUBROUTINE IS TO RETAIN COMPATABILITY
32 ;WITH THE OLD PROCESSOR IDENTIFICATION AND TIMING
33 ;PACKAGE TO RETRIEVE THE LOOP COUNT FOR
34 ;THE DIA,B OR C LOOP TYPE 2
35 02600 024540 ;DIA: LDA 1,NUCAL
36 02601 001400 JMP 0,3

```

```

A 0041 .MAIN
01
02
03
04 02602 062677 SCORE: IORST
05 02603 102400 SUB 0,0 JAC0=NULL CHARACTER
06 02604 004463 JSR TYME
07 02605 061111 DOAS 0,TYO JPASSED TO TYME
08 02606 063511 SKPBZ TYO JFOR EXECUTION
09 02607 006533 SCORA: JSR @TUMBLER JOUT TEXT
10 02610 002752 SESOUT
11
12 JTHE FOLLOWING SERIES OF INSTRUCTIONS WILL
13 JCALCULATE THE ITERATION COUNT FOR
14 J1 BIT OF TYO OUTPUT AFTER RETRIEVING
15 JTHE CONSOLE BAUD RATE FROM THE
16 JTEST OPERATOR--REQUIRES SUBROUTINE TIND
17 02611 006533 JSR @KEYS
18 02612 000775 JMP SCORA JINPUT ERROR
19 02613 044530 STA 1,LOCK JSAVE BAUD RATE
20 02614 030535 LDA 2,S,3D1 J10
21 JROUTINE ASSUMES AN 11 BIT CHARACTER
22 02615 151400 INC 2,2 JASSUME 11 BITS
23 02616 024521 LDA 1,ORDINAL JCOUNT FOR FULL CHAR
24 02617 102400 SUB 0,0
25 02620 006526 JSR @KEYS+2 JCHAR TIME/#BITS
26 02621 101004 MOV 0,0,SZR JIF ANY REM.
27 02622 125400 INC 1,1 JFUDGE BIT COUNT
28 02623 020520 LDA 0,LOCK
29 02624 044517 STA 1,LOCK JSAVE ITR COUNT 1 BIT
30 02625 131000 MOV 1,2
31 02626 105000 MOV 0,1 JAC1 = BAUD RATE
32 JBAUD RATE TIMES COUNT FOR 1 BIT
33 JWILL EQUAL ITERATION COUNT FOR 1 SECOND
34 02627 102400 SUB 0,0
35 02630 006515 JSR @KEYS+1 JMWL AC1*AC2
36 02631 040517 STA 0,KN JSAVE DOUBLE LENGTH
37 02632 044515 STA 1,KS JRESULT
38 J1 SECOND DIVIDED BY 10 = 100 MILLISECONDS
39 02633 030516 LDA 2,S,3D1 J10
40 02634 006512 JSR @KEYS+2
41 02635 030502 LDA 2,ORDINAL JCOUNT FOR 1 CHAR
42 02636 044501 STA 1,ORDINAL JORDINAL=100 MS TYPE 1
JCONTINUE CALCULATIONS NEXT PAGE

```

```

A 0042 .MAIN
01
02
03
04 02637 020511 LDA 0,KN
05 02640 024507 LDA 1,KS J1 SEC. RESTORED
06 02641 006505 JSR @KEYS+2 JDIVIDE BY CHAR.
07 02642 044505 STA 1,KS J# CHAR. 1 SEC
08
09 JCALC RELATIONSHIP OF REM. TO 1 CHAR TO FILL SECOND
10 02643 145120 MOVZL 2,1
11 02644 111005 MOV 0,2,SNR
12 02645 151400 INC 2,2
13 02646 102400 SUB 0,0
14 02647 006477 JSR @KEYS+2 JDIVIDE REM INTO CHAR
15 JAC1=FUDGE FACTOR 1 RELEATIONSHIP OF CHAR TO TOTAL 1 SEC
16 02650 131000 MOV 1,2 JFINISH CALCULATIONS ON LOOP TYPE 1 TO= 1SECOND
17 02651 024467 LDA 1,NUCAL JFUDGE FACTOR
18 02652 125120 MOVZL 1,1 JINTO CHARACTER TIME
19 02653 102400 SUB 0,0
20 02654 006472 JSR @KEYS+2 JWILL =
21 02655 121000 MOV 1,0 JPORTION OF CHAR
22 02656 024462 LDA 1,NUCAL JTO COMPLETE 1 SECOND
23 02657 030470 LDA 2,KS J1 CHAR. TYPE 2 LOOP
24 02658 006465 JSR @KEYS+1 J# CHARS IN 1 SEC
25 JDOUBLE LENGTH AC1,AC1=1 SECOND FOR TYPE 2 LOOP
26 02661 030470 LDA 2,S,3D1 JDIVIDE BY 10 FOR 100 MS
27 02662 006464 JSR @KEYS+2
28 02663 044465 STA 1,NUCAL
29 02664 131000 MOV 1,2 JAC2=100MS LOOP2
30 02665 024452 LDA 1,ORDINAL JAC1 =100MS LOOP 1
31 02666 002453 JMP @SVTIME

```

```

A 0043 .MAIN
01 ;
02 ;THE FOLLOWING SUBROUTINE DETERMINES THE ITERATION
03 ;COUNTS FOR THE DEVICE SPECIFIED BY THE INSTRUCTIONS
04 ;FOLLOWING THE JSR CALL TO TYME
05 ;DOAS 0,RTC OR DOAS 0,TTD
06 ;SKPBZ RTC OR SKPBZ TTD
07 ;
08 02667 024001 TYME1 LDA 1,1 ;SAVE INTR. LINK
09 02670 044446 STA 1,RVTHP
10 02671 024437 LDA 1,ENTYM
11 02672 044001 STA 1,1
12 02673 025400 LDA 1,0,3 ;FOR LOOP 2 INTR.
13 02674 044410 STA 1,TIMA ;GET DOAS
14 02675 044414 STA 1,TIMB
15 02676 044421 STA 1,TIMC ;FOR EXECUTE
16 02677 025401 LDA 1,1,3 ;GET SKPBZ
17 02700 044405 STA 1,TIMA+1
18 02701 044414 STA 1,TIMB+4 ;FOR EXECUTE
19 02702 152400 SUB 2,2
20 02703 126400 SUB 1,1 ;CLR CTRS
21 02704 061114 TIMA1 DOAS 0,RTC ;OR TTD
22 02705 063514 SKPBZ RTC
23 02706 000777 JMP ,-1 ;WAIT FOR DONE
24 02707 124004 COM 1,1,SZR ;AND 2ND DONE
25 02710 000774 JMP TIMA ;THEN START COUNTING
26 02711 061114 TIMB1 DOAS 0,RTC ;THE THIRD DONE
27 ;THE FOLLOWING COMPRISES LOOP TYPE 1
28 02712 101000 MOV 0,0
29 02713 125405 INC 1,1,SNR ;WATCH FOR OFLOW
30 02714 000403 JMP ,+3
31 02715 063514 SKPBZ RTC
32 02716 000774 JMP TIMB+1
33 ;LOOP TYPE 2 IS COUNTED UNTIL PI FROM DEVICE
34 02717 061114 TIMC1 DOAS 0,RTC
35 02720 000177 INTEN
36 ;THE FOLLOWING INSTR. COMPRISE THE LOOP TYPE 2
37 02721 060200 NID 0 ;AND IT ITERATES UNTIL
38 02722 060400 DIA 0,0 ;INTERRUPTED BY PI
39 02723 102025 ADC 0,0,SNR
40 02724 063077 HALT ;FILL INSTR.
41 02725 151404 TNC 2,2,SZR ;LOOP CTR
42 02726 000773 JMP TIMC+2
43 02727 063077 HALT ;DEVICE OR PI FAILED
44 02730 002731 ENTYM1 ,-1 ;TO HERE WHEN PI
45 02731 044406 STA 1,ORDINAL ;SAVE LOOP 1
46 02732 050406 STA 2,NUCAL ;AND LOOP 2
47 02733 020403 LDA 0,RVTHP
48 02734 040001 STA 0,1 ;RESTORE INTR. LINK
49 02735 001402 JMP 2,3 ;RETURN TO CALL

```

```

A 0044 .MAIN
01 ;
02 ;CONSTANTS SUBR. LINKS AND TEMP STORES
03 02736 000000 RVTHP1 0
04 02737 000000 ORDINAL1 0
05 02740 000000 NUCAL1 0
06 02741 000000 SVTIME1 0
07 02742 002255 TUMBLER1 MESS
08 02743 000000 LOCK1 0
09 02744 002471 KEYS1 TIND
10 02745 000256 MULT+1
11 02746 000270 IDIV+1
12 02747 000000 KS1 0
13 02750 000000 KN1 0
14 02751 000012 S.3011 10.
15 ;SESOUI1 .TXTE 1<15><12><12>
02752 005215
16 02753 152012 TTD BAUD RATE ?# 1
02754 147724
02755 041240
02756 052501
02757 120104
02760 040722
02761 142724
02762 037640
02763 120275
02764 000000
17
18 .TXT /COPYRIGHT (C) DGC,1969,70,72,73,74
02765 047503
02766 054520
02767 044522
02770 044107
02771 020124
02772 041450
02773 020051
02774 043504
02775 025103
02776 034401
02777 034466
03000 033454
03001 026060
03002 031067
03003 033454
03004 025063
03005 032067
19 03005 046101 ALL RIGHTS RESERVED/
03007 020114
03010 044522
03011 044107
03012 051524
03013 051040
03014 051505
03015 051105
03016 042526
03017 000104
20
21 03020 000000 PRGEND: 0 ;REST OF CORE IS BUFFER

```

```

A 0045 ,MAIN
01
02
03      003061      .LOC ,+40
04      .TXTE !PTP-R DIA6!
0501 152120
0302 026520
0303 120322
0304 144504
0305 033101
0306 000000
05 0307 000010      000010
06 03070 000002      000002
07 03071 000002      000002
08 03072 000060      000060
09 03073 000000      000000
10 03074 000000      000000
11 03075 000000      000000
12 03076 100012      100012
13
14      .END

```

```

0046 ,MAIN
A00 000753      5/19      6/51      7/30      16/02      21/38      29/32
A02 000760      16/00
A04 000766      16/15      16/33
A17 001013      16/37
A18 001015      17/02
A20 001023      17/09
A21 001030      17/15
A24 001035      17/21
A26 001043      17/28
A28 001052      17/36
A30 001061      17/44
A32 001071      18/02
A34 001101      18/11
A36 001110      18/19
A38 001120      18/28
A40 001127      18/36
A42 001137      18/45
A44 001145      18/52
A46 001153      19/02
A48 001165      19/13
A50 001175      19/23
A52 001204      19/30
A54 001214      19/39
A56 001231      20/02
A58 001244      20/14
A60 001253      20/22
A62 001263      20/31
A64 001274      20/41
A66 001304      20/50
A68 001313      21/02
A70 001330      21/16
ALTER 000422      18/22
ALTPU 000400      5/24      10/04
ALTR1 000503      11/44      11/54      11/55
ALTR2 000472      5/25      11/34      11/52
AUTOF 002220      31/32      31/40
BAD 000147      6/32      12/02      12/12
BADED 001360      12/11      21/41
C0774 000067      5/34      12/18      12/33
C1000 000141      6/26      13/30      14/31
C12 000160      6/41      6/53

```

```

C1420 020155      6/30      12/26
C16 000130      6/17
C1639 000127      6/16
C20 001534      22/26      23/58      25/38      25/50      26/03
C4 000077      5/43      19/15      20/03
C401 000104      5/48      10/06
C54 000125      6/14
C5405 000124      6/13
C6 000100      5/44      21/11      21/25
C60K 000115      6/06      7/07      7/13      7/20      7/24
C77X 000137      6/24      11/06      12/29
C7K 000126      6/15
CAL10 000071      5/36      8/41      13/28
CALIB 000070      5/35      8/26      8/45      13/09      13/25      21/00      21/22
CANNO 002547      13/16      37/57
CENIO 001533      23/40      23/57
CD100 000623      13/26      13/36
CD50 000543      8/44      8/57

```


0247 MAIN

CDVCD	000113	6/04	7/09	7/14	7/37				
CHAR	002370	32/13	33/38	33/41	34/31	35/01	35/27	37/25	
CHAR1	002400	35/09							
CHA3	002405	35/11	35/14	35/19					
CHORZ	002425	35/16	35/20	35/32	36/14	36/17			
CHRET	002424	35/01	35/13	35/21	35/31				
CIDIV	000062	5/29	13/27	21/10	21/24				
CIDT	000111	6/02	7/35						
CJPUN	000426	10/04	10/26						
CNIOC	001212	16/20	16/35						
CPTR.	000121	6/10	7/16	26/45					
CREAD	000110	5/52	10/08						
CRLF	002415	6/35	32/15	35/23					
CTABZ	000731	15/18	15/40						
CTRPB	000402	5/22	10/05						
CTRR1	000431	11/15	11/18	11/27	11/29	11/32			
CTRR2	000470	11/25	11/31						
CTRR3	000447	5/23	11/13						
CVCLE	002122	5/40	30/23						
CVCTS	002161	30/28	31/04						
DECOC	002317	34/13	34/32	34/34	34/42				
DECOT	002325	34/20	34/25						
DECP	002334	34/21	34/26						
DECTB	002353	34/07	34/42						
DEL	000063	5/30	11/09						
DEL1	000064	5/31	12/36						
DELAY	000344	5/32	9/02						
DEVCI	000241	7/41	7/52						
DEVCC	000246	7/39	7/46						
DEVCD	000231	7/12	7/18	7/32	7/44				
DEVRE	000122	6/11	7/32	7/45					
DLODP	000273	8/15	8/20						
DVCD	000114	6/05	7/11	7/17	7/22	7/49			
EGGS	002056	29/14	31/41	31/49					
EHALT	000076	5/42	18/05	16/10	16/29	17/06	17/12	17/18	17/24
		17/32	17/40	17/46	18/06	18/16	18/24	18/33	18/42
		18/49	18/56	19/10	19/20	19/27	19/36	19/50	20/11
		20/19	20/28	20/38	20/47	20/55	21/13	21/27	22/05
		22/10	22/16	22/22	22/32	22/40	22/48	22/54	23/05
		23/12	23/19	23/28	23/49	24/08	24/17	24/26	25/07
		25/20	25/34	25/46	25/57	26/09	26/20	26/27	26/38
		26/48	27/05	27/26	28/04	28/06	28/08	28/10	28/12
		28/14	28/16	28/18	28/21	28/23	28/25	28/27	28/29
		28/31	28/33	28/35	29/08				
ENDIT	002100	6/08	30/02						
ENTER	002101	5/39	30/04						
ENTYM	002730	43/10	43/44						
EPRIN	002234	31/37	31/48	32/02					
ER	000076	5/41	5/42						
ERET	002202	31/24	31/39						
ERR	002173	5/41	31/16						
ERR1	002210	31/23	31/31						
ERRCT	002114	30/09	30/16	30/42	31/03	31/27			
ESNIT	002113	29/09	30/08	30/15	30/33	31/07	31/21	31/33	32/02
		32/10							
FIRST	000116	6/07	7/33						
FLOT	000703	15/12	15/16	15/33					
FLTCT	000752	15/03	15/06	15/11	15/28	15/30	15/32	15/57	
GET	000645	6/20	11/14	11/16	11/37	11/45	14/20		

0248 MAIN

GET1	000654	14/22	14/27						
GET2	000657	14/26	14/31						
GOOD	000146	6/31	12/03	12/08					
GOOD1	001355	12/07	21/40						
HEAD1	002253	32/05	32/19						
HERE	000161	5/05	6/46						
HSPUN	000157	6/40	13/33	14/02	14/08				
HSREA	000156	6/39	13/07	14/20					
ICHAR	002246	30/45	32/13						
ICRLF	002250	30/41	32/03	32/15					
IDIV	000267	5/20	8/11	44/11					
IGET	000133	6/20	15/24	15/34					
IIINT	000123	6/12	15/01	15/23	22/02				
IMESS	002245	32/04	32/12						
INIT	000362	6/12	10/12	11/13	11/34	13/02	16/02		
INTEM	000622	13/08	13/34	13/35					
IPDEC	002252	30/43	32/17						
IPFLT	000135	6/22							
IPOCT	002251	32/09	32/16						
IPUT	000131	6/18	10/18	15/05	15/13				
IRDER	000134	6/21	15/38						
IRFLT	000136	6/23							
ITR	002111	8/38	30/05	30/13	30/30				
ITRCT	002112	30/06	30/14	30/27	30/31				
ITYPE	002247	32/14							
IZOCT	000154	6/37	12/09	12/13					
K17	000132	6/19	15/16						
K40	000176	7/03	16/13	23/33					
KEYS	002744	41/16	41/24	41/34	41/39	42/06	42/13	42/20	42/24
		42/27	44/09						
KN	002750	41/35	42/04	44/13					
KS	002747	41/36	42/05	42/07	42/23	44/12			
LAST	000117	6/08	7/42						
LEDR	000072	5/37							
LENGT	000142	6/27	10/19	10/34	11/26	11/53	12/32		
LOCK	002743	41/18	41/27	41/28	44/08				
LOOP	000075	5/40	16/06	16/11	16/30	17/07	17/13	17/19	17/26
		17/34	17/42	17/51	18/09	18/17	18/26	18/34	18/43
		18/50	18/57	19/11	19/21	19/28	19/37	19/51	20/12
		20/20	20/29	20/39	20/48	20/56	21/14	22/06	22/11
		22/17	22/23	22/33	22/41	22/49	22/57	23/06	23/13
		23/22	23/31	23/52	24/09	24/18	24/27	25/08	25/21
		25/35	25/47	25/58	26/10	26/21	26/28	26/39	26/40
		27/12	27/27	28/36					
LOOPR	002121	30/04	30/11	30/21	31/12				
M100	000322	7/19	7/47	8/25	8/39				
MDCTR	000302	8/03	8/13	8/22					
MESS	002255	6/36	32/12	33/26	33/39	44/07			
MESSR	002426	33/26	33/29	33/47	34/09	35/23	35/33	37/03	37/12
MLOOP	000260	8/04	8/08						
MSAV	000303	8/02	8/10	8/12	8/21	8/23			
MULT	000255	8/01	44/10						
NOEX	002152	30/37	30/50						
NUCAL	002740	40/35	42/17	42/22	42/28	43/46	44/05		
OCYAB	002343	34/04	34/34						
ORDIN	002737	41/22	41/40	41/41	42/30	43/45	44/04		
OUT	000611	13/13	13/25						
PASS	000107	5/51							
PASSE	001363	21/32	21/42	29/22					

