

**TE16,TU16/77**

UTILITY DRIVER  
CZTEFB0

AH-B159B-MC

COPYRIGHT 77-78

FICHE 1 OF 1

JAN 1979

**digital**

MADE IN USA

1  
.REM %

IDENTIFICATION

PRODUCT CODE: AC-3158B-MC  
PRODUCT NAME: CZTEFBO TU16/TE16/TU77 UTILITY DRIVER  
DATE CREATED: 21 MAY 78  
MAINTAINER: DIAGNOSTIC ENGINEERING  
AUTHOR: J. G. ADAMS

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.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED UNDER A LICENSE AND MAY ONLY BE USED OR COPIED IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1977,1978 BY DIGITAL EQUIPMEN CORPORATION

:TABLE OF CONTENTS

PARAGRAPH	SUBJECT	PAGE
1.	ABSTRACT	1
2.	REQUIREMENTS	1
3.	LOADING PROCEDURE	1
4.	STARTING PROCEDURE	1
5.	CONSOLE SWITCHES	1
6.	OPERATION	2
7.	PROGRAM DESCRIPTION	5
8.	LISTING	

1.      ABSTRACT

THIS PROGRAM IS INTENDED AS A BRUTE FORCE ROUTINE TO EXECUTE AN OPERATION OR SERIES OF OPERATIONS, CONTINUOUSLY REGARDLESS OF THE RESULTS OF THE OPERATION. BECAUSE OF THE COMPLEXITY OF THE TU16/TE16/TU77 MAG TAPE SYSTEM AS OPERATED ON THE MASSBUS, IT IS NOT ALWAYS POSSIBLE TO PROVIDE FOR EVERY CONTINGENCY IN THE NORMAL PROGRAMS. THEREFORE THIS UTILITY DRIVER WILL ALLOW AN OPERATOR TO EXECUTE ANYTHING DESIRED IN ANY ORDER. THERE ARE NO ERROR CHECKS OR PRINTOUTS MADE, AND ANY VARIATION FROM PRESET SEQUENCES AND VALUES ARE MADE BY CHANGING THE APPROPRIATE MEMORY LOCATIONS.

2.      REQUIREMENTS

2.1     HARDWARE:

- A. ANY PDP-11 PROCESSOR - WITH OR WITHOUT HARDWARE SWITCH REGISTER.
- B. RH MASSBUS CONTROLLER
- C. TMO2/TMO3 MAG TAPE CONTROLLER
- D. AT LEAST ONE (1) TU16/TE16/TU77 SLAVE

2.2     STORAGE:

THIS PROGRAM REQUIRES AT LEAST 3K OF CORE

3.      LOADING PROCEDURE:

USE STANDARD BINARY LOADING PROCEDURE

4.      STARTING PROCEDURE

THE PROGRAM IS ALWAYS STARTED AT LOCATION 200 (8)

\*\*\*LOC. 176 (SWREG) IS DEFINED AS THE SOFTWARE SWITCH REGISTER  
(REFER TO SECTION 5 FOR MORE DETAIL)

\*\*\*IF THE SOFTWARE SWITCH REGISTER IS USED THE DIAGNOSTIC TYPES OUT THE FOLLOWING  
MESSAGE: SWR=XXXXXX NEW= (REFER TO SECTION 5 FOR OPERATOR OPTIONS)  
AT THE START OF THE PROGRAM.

5. CONSOLE SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

CONTROL :

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G <^G>; THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE PROGRAM.
- 2) THE MACHINE WILL THEN TYPE: SWR=XXXXXXNEW= (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE "'NEW='" HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
  - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED AND ONLY 6 NUMBERS WILL BE ALLOWED)  
IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
  - B) IF A CONTROL U <^U> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.

SW15(100000): 1=STOP AFTER EACH OPERATION  
                  0=PROCEED  
SW14(040000): 1=STOP AT THE END OF THE OPERATION SEQUENCE  
                  0=PROCEED  
SW13(020000): 1=IGNORE END OF TAPE (EOT)  
                  0=REWIND AT END OF TAPE (EOT)

5.1 HALT

-----  
\*\*\*TO CHANGE THE CONTENTS OF SWREG TYPE <^G> BEFORE PRESSING CONTINUE AFTER A HALT.\*\*\*

6. OPERATION

THE PROGRAM OPERATION IS QUITE SIMPLE, BUT DOES REQUIRE THE OPERATOR TO HAVE KNOWLEDGE OF THE TU16/TE16/TU77 TAPE SYSTEM AS OPERATED ON THE RH MASSBUS CONTROLLER. THE OPERATOR MUST BE ABLE TO DECIDE WHICH SEQUENCE OF OPERATION IS REQUIRED, AND WHAT VALUES TO ASSIGN TO THE VARIOUS PARAMETERS REQUIRED TO EXECUTE THEM. THE OPERATION SEQUENCE IS SET UP BY LOADING A TABLE WITH THE FUNCTION CODES OF THE DESIRED OPERATIONS AND SETTING THE NUMBER OF OPERATIONS IN A COUNTER. THE PROGRAM IS SET UP TO DO A WRITE OF TEN (8) WORDS OF ALL ONES DATA TO SLAVE ZERO (0) ON DRIVE ZERO (0) IN PE (1600 BPI) WITH A NINE TRACK NORMAL DATA FORMAT. THE DATA ADDRESS IS 3000 (8). THE OPERATION SEQUENCE IS SET TO DO A SINGLE WRITE. IF LOADED AND STARTED AT 200 (8) WITH NO CHANGES MADE AND SWITCH 14 AND 15 SET TO A ZERO (0), THIS OPERATION WILL BE EXECUTED CONTINUOUSLY. THE FOLLOWING IS THE LIST OF PARAMETERS WHICH MAY BE VARIED AND A DESCRIPTION OF EACH ALONG WITH THEIR CORE LOCATION:

PARAMETER	LOCATION	DESCRIPTION
RH ADDRESS	600	ADDRESS OF RH (THE FIRST REGISTER ADDRESS: CS1)
DRIVE NUMBER	700	SET TO SELECT TMO2/TMO3 NUMBER ADDRESS 0-7
UNIT DESCRIPTION	702	SET SELECTED SLAVE NUMBER (0-7) IN BITS 0,1,2 SELECT PARITY IN BIT 3 (0=ODD 1=EVEN) SELECT DATA FORMAT IN BITS 4,5,6,7 SELECT DENSITY IN BITS 8,9,10
FRAME COUNT	704	SET NUMBER OF FRAMES TO WRITE PER WORD COUNT AND FORMAT IN TWOS' COMPLIMENT
WORD COUNT	706	SET NUMBER OF WORDS TO BE TRANSFERRED IN TWOS' COMPLIMENT
READ ADDRESS	710	SET DESIRED ADDRESS FOR START OF READ BUFFER.
WRITE ADDRESS	712	SET DESIRED ADDRESS FOR START OF WRITE BUFFER.
READY DELAY	714	THIS DELAY VALUE IS USED BY THE PROGRAM TO ESTABLISH A MAXIMUM TIME TO AWAIT THE COMPLETION OF AN OPERATION BEFORE PROCEEDING TO THE NEXT. ** (DEFAULT IS APPROX 435 MS FOR PDP-11/20)**
READY MULTIPLIER	716	IF THE VALUE SET INTO 714 DOES NOT ALLOW ENOUGH TIME, INCREASE THE SIZE OF THE MULTIPLIER. EACH INCREMENT OF THE MULTIPLIER WILL CAUSE THE 714 DELAY TO BE EXECUTED THAT MANY MORE TIMES.

OPERATION DELAY	720	THIS DELAY IS USED TO ALLOW FOR SOME AMOUNT OF TIME BETWEEN THE EXECUTION OF EACH OPERATION. IT IS LOADED AND USED JUST AS IN THE READY DELAY(714) **(DEFAULT IS APPROX 54 MS FOR PDP-11/20)**
OPER MULTIPLIER	722	THIS IS USED JUST AS THE READY DELAY MULTIPLIER(716)
OPERATION NUMBER	724	THIS IS THE NUMBER OF OPERATIONS TO BE PERFORMED IN A SEQUENCE AND SHOULD REFLECT THE NUMBERS OF OPERATIONS SET INTO THE OPERATION TABLE.
OPERATION TABLE	740-770	THIS TABLE (CONSISTING OF 15 LOCATIONS) IS TO BE LOADED WITH THE FUNCTION CODES FOR EACH OPERATION TO BE PERFORMED IN SEQUENCE. THE NUMBER OF ENTIRES MAY BE FROM ONE (1) TO FIFTEEN (15). MAKE SURE THAT THE NUMBER OF FUNCTION CODES SET IN THE TABLE IS REFLECTED BY THE NUMBER IN LOCATION 724 (OPNUM)

6.1 FUNCTION CODES

20=READ IN PRESET  
02=REWIND-OFF LINE  
06=REWIND  
10=DRIVE CLEAR  
26=WRITE TAPE MARK  
24=ERASE  
30=SPACE FORWARD  
32=SPACE REVERSE  
50=WRITE CHECK FORWARD  
56=WRITE CHECK REVERSE  
60=WRITE FORWARD  
70=READ FORWARD  
76=READ REVERSE

6.2 DATA FORMATS (BIT 7,6,5,4 OF UNIT DESCRIPTION)

14=NINE TRACK NORMAL: 2 FRAMES PER WORD  
15=CORE DUMP: 4 FRAMES PER WORD

6.3 DENSITY (BITS 10,9,8 OF UNIT DESCRIPTION)

4=1600 BPI:PE (PE USES ONLY ODD PARITY)  
3=800 BPI:NRZI  
2=800 BPI:NRZI (TU16 ONLY)  
1=556 BPI:NRZI (TU16 ONLY)  
0=200 BPI:NRZI (TU16 ONLY)

6.4 PARITY (BIT 3 OF UNIT DESCRIPTION)

1=EVEN PARITY  
0=ODD PARITY

6.5 SLAVE SELECT (BITS 2,1,0 OF UNIT DESCRIPTIONS)

SET TO DEVICE SLAVE ADDRESS (0-7)



7. PROGRAM DESCRIPTION

IN ORDER TO MAINTAIN THE CONTINUOUS EXECUTION OF THE OPERATIONS DESCRIBED THE PROGRAM IS ORGANIZED AS FOLLOWS:

START  
INITIALIZE THE RH  
SET UP TAPE PARAMETERS (DENSITY, PARITY, FORMAT: WORD COUNT, FRAME COUNT, BUS ADDRESS)  
SELECT DEVICE TO TEST (DRIVE NUMBER, SLAVE NUMBER)  
EXECUTE OPERATION (SET FUNCTION AND FROM OP TABLE AND SET GO=1)  
AWAIT END OF OPERATION (READY DELAY)  
STOP IF SWITCH 15=1  
DO OPERATION DELAY (OP DELAY)  
STOP IF LAST OPERATION IN SEQUENCE AND SWITCH 14=1  
POINT TO NEXT FUNCTION CODE IN OP TABLE  
JUMP BACK TO START

7.1 FLOW:    START:            HOUSEKEEPING  
          INIT:            CLEAR MASSBUS AND TMO2/TMO3  
          SET UP:          SET UP REQUIRED REGISTERS  
          EXECUTE:         SET FUNCTION AND GO=1  
          AWAIT END:       LOOP ON DRY=1 AS LONG AS ALLOWED BY READY DELAY  
          STOP:            IF SWITCH 15=1  
          DELAY:           PER OP DELAY  
          END OF RSEQUENCE? IF NOT JUMP TO START  
          STOP:            IF SWITCH 14=1  
          JUMP TO START    RESTART SEQUENCE

7.2 VARIATIONS: THERE ARE TWO VARIATIONS MADE FROM THIS FLOW.  
BOTH ARE CAUSED BY A PARTICULAR FUNCTION CODE.  
IF A READ REVERSE IS TO BE EXECUTED, THEN THE BUS ADDRESS IS INCREMENTED BY THE SIZE OF THE RECORD BECAUSE THE DATA IS LOADED INTO MEMORY IN REVERSE (I.E: HIGH ADDRESS TO LOW ADDRESS)  
THE SECOND VARIATION IS CAUSED BY A SPACE (FORWARD OR REVERSE) OPERATION AND IT IS THAT THE FRAME COUNTER IS SET TO A -1 SO THAT ONLY ONE (1) RECORD IS SPACED OVER. IF YOU WISH TO SPACE OVER MORE THAN ONE (1) RECORD, SET LOCATION 1100 (8) TO THE TWOS' COMPLIMENT OF THE NUMBER OF RECORDS DESIRED.

8. LISTING

310

%

.LIST BIN,LOC,SEQ

311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
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

```
.TITLE CZTEFBO TU16/TE16/TU77 UTIL  
:UTILITY DRIVER  
:AC-B158B-MC  
:15 APR 77  
:J. G. ADAMS  
:REVISED APRIL ,1976 BY S. CARPENTER  
:      1) SUPPORTS SOFTWARE SWITCH REGISTER  
:      2) SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER  
:REVISED APRIL 1978 BY J. G. ADAMS  
:      1)DOCUMENTATION CHANGES TO REFLECT TMO3/TE16 CAPABILITY  
:REVISED JUNE 1977 BY J. G. ADAMS  
:      2)DOCUMENTATION CHANGES TO REFLECT ADDED TMO3-TU77 CAPABILITY
```

.ABS

:CONSOLE SWITCHES

```
:SW 15=1(100000) STOP ON EACH OPERATION  
:      0 CONTINUE  
:SW 14=1(040000) STOP AT END OF SEQUENCE  
:      0 CONTINUE  
:SW 13=1(020000) IGNORE END OF TAPE (EOT)  
:      0 REWIND AT END OF TAPE (EOT)
```

:REGISTER EQUIVES

```
R0=%0  
R1=%1  
R2=%2  
R3=%3  
R4=%4  
R5=%5  
SP=%6  
PC=%7
```

```
.=46  
RESTART:      170      ;ALLOW RESTART WHEN <LF> IS PRESSED  
                ;DURING CHANGING OF SWREG IF SOFTWARE SWITCH  
                ;REGISTER IS USED.
```

:SOFTWARE SWITCH REGISTER\*\*\*\*\*

```
.=176  
SWREG: 0      ;SOFTWARE SWITCH REGISTER
```

:\*\*\*\*\*~\*\*\*\*\*

```
:THIS PROGRAM SUPPORTS THE SOFTWARE SWITCH REGISTER LOC.176.  
:REFER TO SECTION 5 OF DOCUMENT FOR DESCRIPTION
```

:\*\*\*\*\*~\*\*\*\*\*

```
000046 000046  
000176 000176  
000176 000000
```

```
366                                     ;STARTING ADDRESS
367
368                                     . =200
369 000200 000200 001110             JMP      SETUP
370 000200 000167 001110             . =600
371 000200 000600
372
373                                     ;TM02/TM03 REGISTERS
374
375 000600 172440                     C1:      172440
376 000602 172442                     WC:      172442
377 000604 172444                     BA:      172444
378 000606 172446                     FC:      172446
379 000610 172450                     CS:      172450
380 000612 172452                     DS:      172452
381 000614 172454                     ER:      172454
382 000616 172456                     AS:      172456
383 000620 172460                     CC:      172460
384 000622 172462                     DB:      172462
385 000624 172464                     MR:      172464
386 000626 172466                     DT:      172466
387 000630 172470                     SN:      172470
388 000632 172472                     C2:      172472
389
390                                     ;PROCESSOR ADDRESSES
391
392 000634 177776                     PSW:     177776           ;PROCESSOR STATUS
393 000636 177570                     SWR:     177570           ;SWITCH REGISTER
394
395                                     ;TTY REGISTERS
396
397 000640 177560                     TKS:     177560           ;TTY READER STATUS
398 000642 177562                     TKB:     177562           ;TTY READ BUFFER
399 000644 177564                     TPS:     177564           ;TTY PUNCH STATUS
400 000646 177566                     TPB:     177566           ;TTY PUNCH BUFFER
```

401 000700  
402  
403  
404 000700 000000  
405 000702 002300  
406 000704 177760  
407 000706 177770  
408 000710 004000  
409 000712 005000  
410 000714 100000  
411 000716 000001  
412 000720 010000  
413 000722 000001  
414 000724 000001  
415 000726 000000  
416 000730 000000  
417 000732 000000  
418 000734 000000  
419 000736 000000

:=700  
:SET PARAMETERS DESIRED FOR UNIT UNDER TEST\*\*\*\*\*  
DRVN: 0 :DRIVE NUMBER  
UDES: 2300 :SLAVE DESCRIPTION  
FCNT: -20 :FRAME COUNT  
WCNT: -10 :WORD COUNT  
RADDR: 4000 :READ ADDRESS  
WADDR: 5000 :WRITE ADDRESS  
RDYDLY: 100000 :READY DELAY  
RDYDX: 1 :READY DELAY MULTIPLIER  
OPDLY: 10000 :OPERATION DELAY  
OPDX: 1 :OPERATION DELAY MULTIPLIER  
OPNUM: 1 :NUMBER OF OPERATION (1 TO 15)  
TIB: 0  
TOB: 0  
COUNT: 0  
RDSW: 0  
TEMPST: 0

420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442

:OPERATION TABLE\*\*\*\*\*  
:ENTER OPERATION SEQUENCE DESIRED.  
:MUST HAVE AT LEAST 1 OPERATION, AND  
:MAY HAVE UP TO 15(8).  
:SET THE OPERATION COUNTER EQUAL  
:TO THE NUMBER OF OPERATIONS IN  
:THE SEQUENCE.  
:  
:20 = READ IN PRESET  
:02 = REWIND-OFFLINE  
:06 = REWIND  
:10 = DRIVE CLEAR  
:26 = WRITE TAPE MARK  
:24 = ERASE  
:30 = SPACE FORWARD  
:32 = SPACE REVERSE  
:50 = WRITE CHECK FORWARD  
:56 = WRITE CHECK REVERSE  
:60 = WRITE FORWARD  
:70 = READ FORWARD  
:76 = READ REVERSE

443 000740 000060  
444 000742 000000  
445 000744 000000  
446 000746 000000  
447 000750 000000  
448 000752 000000  
449 000754 000000  
450 000756 000000  
451 000760 000000  
452 000762 000000  
453 000764 000000  
454 000766 000000  
455 000770 000000

OPTBL: 60  
0  
0  
0  
0  
0  
0 :FILL WITH OPERATION SEQUENCE  
0  
0  
0  
0

```

456          001000          .=1000
457          :START OF PROGRAM*****
458
459 001000 012706 000500 START: MOV #500,SP
460 001004 012777 000340 177622 MOV #340,@PSW
461
462 001012 016700 177706 MOV OPNUM,R0 ;SET COUNTER
463 001016 012701 000740 MOV #OPTBL,R1 ;SET POINTER
464 001022 012777 000040 177560 A: MOV #40,@CS ;INIT
465 001030 016777 177644 177552 MOV DRVN,@CS ;DRIVE NUMBER
466 001036 016777 177640 177566 MOV UDES,@C2 ;UNIT DESCRIPTION
467 001044 016777 177636 177530 MOV WCNT,@WC ;WORD COUNT
468 001052 016777 177626 177526 MOV FCNT,@FC ;FRAME COUNT
469 001060 012102 MOV (R1)+,R2 ;SET OP CODE
470 001062 022702 000030 CMP #30,R2 ;SEE IF SPACE FORWARD
471 001066 001403 BEQ AA ;IF SO: BR
472 001070 022702 000032 CMP #32,R2 ;SEE IF SPACE REVERSE
473 001074 001003 BNE A0 ;IF NOT: BR
474 001076 012777 177777 177502 AA: MOV #-1,@FC ;SET TO SPACE ONE RECORD
475 001104 022702 000060 A0: CMP #60,R2 ;SEE IF READ OP
476 001110 103404 BLO A1 ;IF SO: BR
477 001112 016777 177574 177464 MOV WADDR,@BA ;SET WRITE ADDRESS
478 001120 000413 BR A3
479 001122 016777 177562 177454 A1: MOV RADDR,@BA ;SET READ ADDRESS
480 001130 022702 000070 CMP #70,R2 ;SEE IF READ OPERATION
481 001134 001405 BEQ A3 ;IF SO: BR
482 001136 016703 177542 MOV FCNT,R3 ;GET FRAME COUNT
483 001142 005403 NEG R3
484 001144 060377 177434 ADD R3,@BA ;SET BUS ADDRESS FOR READ REVERSE
485 001150 052702 000001 A3: BIS #1,R2 ;SET GO BIT
486 001154 000240 NOP
487 001156 000240 NOP
488 001160 010277 177414 MOV R2,@C1 ;START OPERATION
489 001164 000240 NOP
490 001166 000240 NOP
491 001170 016704 177522 MOV RDYDX,R4 ;SET DELAY MULTIPLIER
492 001174 016703 177514 MOV RDYDLY,R3 ;SET READY DELAY
493 001200 032777 000200 177404 B: BIT #200,@DS
494 001206 001005 BNE C ;IF DRY: BR
495 001210 005303 DEC R3
496 001212 001372 BNE B
497 001214 005304 DEC R4
498 001216 001366 BNE B0 ;DELAY FOR DRIVE READY
499 001220 000240 NOP
500 001222 005777 177410 C: TST @SWR ;SEE IF STOP ON OPERATION
501 001226 100001 BPL D ;IF NOT: BR
502 001230 000000 HALT
503 001232 004767 000302 D: JSR PC,CKSWR ;CHECK FOR CNTL G
504 001236 000240 NOP
505 001240 000240 NOP
506 001242 016704 177454 MOV OPDX,R4 ;SET DELAY MULTIPLIER
507 001246 016703 177446 E0: MOV OPDLY,R3 ;SET OPERATION DELAY
508 001252 005303 E: DEC R3
509 001254 001376 BNE E
510 001256 005304 DEC R4
511 001260 001372 BNE E0 ;DELAY BETWEEN OPERATIONS
  
```

```

512 001262 004767 000152      JSR    PC,RWND      ;GO SEE IF REWIND
513 001266 005300              DEC    R0           ;IF SEQUENCE NOT DONE: BR
514 001270 001254              BNE    A           ;IF SEQUENCE NOT DONE: BR
515
516 001272 032777 040000 177336  BIT    #40000,@SWR  ;SEE IF HALT ON SEQUENCE
517 001300 001401              BEQ    1$          ;SEE IF HALT ON SEQUENCE
518 001302 000000              HALT
519 001304 004767 000230      1$:   JSR    PC,CKSWR  ;CHECK FOR CNTL G
520 001310 000167 177464      JMP    START
521
522                               ;RH REGISTER SETUP*****
523
524 001314 000240              SETUP: NOP
525 001316 016701 177256      MOV    C1,R1       ;GET ADDRESS OF CS1
526 001322 012700 000015      MOV    #15,R0      ;SET NUMBER OF REGISTERS
527 001326 012702 000602      MOV    #WC,R2      ;GET FIRST ADDRESS
528 001332 062701 000002      SETA:  ADD    #2,R1 ;INCREMENT
529 001336 010122              MOV    R1,(R2)+    ;LOAD ADDRESS
530 001340 005300              DEC    R0           ;SEE IF DONE
531 001342 001373              BNE    SETA        ;IF NOT: BR
532 001344 012706 000500      MOV    #500,SP     ;SAVE VECTORS
533 001350 013746 000006      SUSWR: MOV    @#6,-(SP)
534 001354 013746 000004      MOV    @#4,-(SP)
535 001360 012737 001400 000004  MOV    #1$,@#4     ;SET UP FOR TIMEOUT
536 001366 022777 177777 177242  CMP    #-1,@SWR    ;REFERENCE HARDWARE SWITCH REGISTER
537 001374 001402              BEQ    2$          ;REFERENCE HARDWARE SWITCH REGISTER
538 001376 000404              BR     3$          ;REFERENCE HARDWARE SWITCH REGISTER
539 001400 022626      1$:   CMP    (SP)+,(SP)+ ;ADJUST STACK
540 001402 012767 000176 177226  2$:   MOV    #SWREG,SWR ;POINT TO SOFTWARE SWITCH REG
541 001410 012637 000004      3$:   MOV    (SP)+,@#4   ;RESTORE VECTORS
542 001414 012637 000006      MOV    (SP)+,@#6
543 001420 023727 000636 000176  CMP    @#SWR,#SWREG ;IS SOFTWARE REG USED
544 001426 001002              BNE    GO          ;BRANCH IF NO
545 001430 004767 000156      JSR    PC,CNTLU   ;ALLOW SOFTWARE SWITCH REGISTER TO BE CHANGED
546 001434 000167 177340      GO:   JMP    START     ;ELSE GO START EXECUTION
547
548                               ;REWIND FROM EOT (PER SW13)
549
550 001440 032777 020000 177170  RWND:  BIT    #20000,@SWR  ;SEE IF IGNORE EOT
551 001446 001033              BNE    RWNDX       ;IF SO: BR
552 001450 032777 002000 177134  BIT    #2000,@DS   ;SEE IF AT EOT
553 001456 001427              BEQ    RWNDX       ;IF NOT: BR
554 001460 012777 000040 177122  MOV    #40,@CS     ;INIT
555 001466 016777 177206 177114  MOV    DRVN,@CS    ;SET DRIVE NUMBER
556 001474 016777 177202 177130  MOV    UDES,@C2    ;SET SLAVE NUMBER
557 001502 012777 000007 177070  MOV    #7,@C1      ;START REWIND
558 001510 032777 000200 177074  RWNDA: BIT    #200,@DS ;SEE IF DRY
559 001516 001774              BEQ    RWNDA       ;IF NOT: BR
560 001520 032777 020000 177064  RWNDB: BIT    #20000,@DS ;SEE IF PIP RESET
561 001526 001374              BNE    RWNDB       ;IF NOT: BR
562 001530 005726              TST    (SP)+       ;RESET STACK
563 001532 000167 177242      JMP    START       ;RESTART SEQUENCE
564 001536 000207      RWNDX: RTS    PC   ;RETURN

```

```

565
566
567
568
569 001540 022767 000176 177070 CKSWR:  CMP    #SWREG,SWR      ;SOFTWARE SWITCH REG PRESENT
570 001546 001041                BNE    OUT          ;NO, GET OUT
571 001550 105777 177064                TSTB   @TKS         ;YES, WAIT FOR
572 001554 100036                BPL    OUT          ;READY, GET CHARACTER
573 001556 017767 177060 177142    MOV    @TKB,TIB     ;AND STRIP OFF
574 001564 042767 177600 177134    BIC    #177600,TIB  ;THE GARBAGE
575 001572 022767 000007 177126    CMP    #7,TIB       ;IS IT A <^G>
576 001600 001024                BNE    OUT
577 001602 012704 002512                MOV    #SCNTG,R4
578 001606 004767 000242                JSR    PC,TTOUT
579 001612 012704 002516                CNTLU:  MOV    #SMSWR,R4
580 001616 004767 000232                JSR    PC,TTOUT
581 001622 017703 177010                MOV    @SWR,R3
582 001626 004767 000354                JSR    PC,OCPE
583 001632 012704 002525                MOV    #SMNEW,R4
584 001636 004767 000212                JSR    PC,TTOUT
585 001642 005037 000736                CLR    @TEMPST
586 001646 004767 000002                JSR    PC,$READ
587 001652 000207                OUT:    RTS          ;GC READ A LINE
588
589 001654 005067 177056                $READ:  CLR    TEMPST
590 001660 012767 000007 177044    MOV    #7,COUNT
591 001666 004767 000546                1$:    JSR    PC,TTIN   ;GO READ A CHARACTER
592 001672 042767 177600 177026    BIC    #177600,TIB  ;STRIP OFF GARBAGE
593 001700 122767 000025 177020    CMPB   #25,TIB      ;IS IT A ^U?
594 001706 001002                BNE    2$           ;BRANCH IF NOT
595 001710 005726                3$:    TST    (SP)+    ;POP THE STACK
596 001712 000737                BR     CNTLU        ;START OVER
597 001714 122767 000015 177004    2$:    CMPB   #15,TIB
598 001722 001013                BNE    4$           ;IS IT A <CR>?
599 001724 012767 000200 177002    MOV    #200,RDSW    ;BRANCH IF NOT
600 001732 004767 000150                JSR    PC,TCRLF
601 001736 022767 000007 176766    CMP    #7,COUNT
602 001744 001037                BNE    7$           ;ECHO IT WITH <LF>
603 001746 005726                8$:    TST    (SP)+    ;WAS IT FIRST CHARACTER
604 001750 000740                BR     OUT          ;CHANGE SWR IF NOT FIRST ONE
605 001752 122767 000060 176746    4$:    CMPB   #60,TIB
606 001760 003004                BGT    5$           ;POP THE STACK
607 001762 122767 000067 176736    CMPB   #67,TIB
608 001770 003005                BGT    6$           ;GET OUT
609 001772 012704 002535                5$:    MOV    #SQUEST,R4
610 001776 004767 000052                JSR    PC,TTOUT
611 002002 000742                BR     3$           ;START OVER IF NOT LEGAL CHARACTER
612 002004 006367 176726                6$:    ASL    TEMPST
613 002010 006367 176722                ASL    TEMPST
614 002014 006367 176716                ASL    TEMPST
615 002020 142767 000060 176700    BICB   #60,TIB      ;GET NITTY-GRITTY
616 002026 156767 176674 176702    BISB   TIB,TEMPST
617 002034 005367 176672                DEC    COUNT
618 002040 001754                BEQ    5$
619 002042 000711                BR     1$
620 002044 016777 176666 176564    7$:    MOV    TEMPST,@SWR ;CHANGE SWITCH REGISTER CONTENTS

```

```

621 002052 000735          BR      8$
622
623
624          ;TTY OUTPUT SUBROUTINE*****
625
626 002054 112467 176650          TTOUT:  MOVB   (R4)+,TOB
627 002060 122767 000043 176642    CMPB   #43,TOB
628 002066 001446          BEQ    TEX
629 002070 122767 000045 176632    CMPB   #45,TOB
630 002076 001403          BEQ    TCRLF
631 002100 004767 000064          JSR    PC,TOG
632 002104 000763          BR     TTOUT
633 002106 112767 000015 176614    TCRLF:  MOVB   #15,TOB
634 002114 004767 000050          JSR    PC,TOG
635 002120 012703 000004          MOV    #4,R3
636 002124 005067 176600          TCRLFA: CLR    TOB
637 002130 004767 000034          JSR    PC,TOG
638 002134 005303          DEC    R3
639 002136 001372          BNE    TCRLFA          ;DO FILLERS
640 002140 112767 000012 176562    MOVB   #12,TOB
641 002146 004767 000016          JSR    PC,TOG
642 002152 105767 176556          TSTB   RDSW
643 002156 100401          BMI    1$
644 002160 000735          BR     TTOUT
645 002162 005067 176546          1$:   CLR    RDSW
646 002166 000406          BR     TEX
647 002170 105777 176450          TOG:   TSTB   @TPS
648 002174 100375          BPL    TOG
649 002176 116777 176526 176442    MOVB   TOB,@TPB
650 002204 000207          TEX:   RTS    PC
651
652          ;OCTAL OUTPUT SUBROUTINE*****
653
654 002206 012767 000001 000222    OCTPE:  MOV    #1,OFL
655 002214 010304          MOV    R3,R4
656 002216 000410          BR     OCTP0
657 002220 005067 000212          CCTP:  CLR    OFL          ;CLEAR FLAG FOR LEADING ZERO
658 002224 010304          OCTPE1: MOV    R3,R4          ;SEE IF NUMBER IS ZERO
659 002226 001004          BNE    OCTP0          ;IF NOT ZERO: BR
660 002230 004767 000162          JSR    PC,OCTPG1      ;ELSE PRINT ZERO
661 002234 000167 000120          JMP    OCTP3          ;SPACE AND EXIT
662 002240 032704 100000          OCTP0: BIT    #100000,R4 ;SEE IF MSD = 1
663 002244 001406          BEQ    OCTP1          ;IF NOT: BR
664 002246 012704 000001          MOV    #1,R4
665 002252 004767 000116          JSR    PC,OCTPG      ;PRINT 1
666 002256 000167 000006          JMP    OCTP2
667 002262 005004          OCTP1: CLR    R4
668 002264 004767 000104          JSR    PC,OCTPG      ;PRINT 0
669 002270 010304          OCTP2: MOV    R3,R4
670 002272 006004          ROR    R4
671 002274 006004          ROR    R4
672 002276 006004          ROR    R4          ;POSITION DIGIT
673 002300 006004          ROR    R4
674 002302 000304          SWAB   R4
675 002304 004767 000064          JSR    PC,OCTPG      ;PRINT DIGIT 2
676 002310 010304          MOV    R3,R4

```



677	002312	006004			ROR	R4	
678	002314	000304			SWAB	R4	
679	002316	004767	000052		JSR	PC,OCTPG	;PRINT DIGIT 3
680	002322	010304			MOV	R3,R4	
681	002324	006104			ROL	R4	
682	002326	006104			ROL	R4	
683	002330	000304			SWAB	R4	
684	002332	004767	000036		JSR	PC,OCTPG	;PRINT DIGIT 4
685	002336	010304			MOV	R3,R4	
686	002340	006004			ROR	R4	
687	002342	006004			ROR	R4	
688	002344	006004			ROR	R4	
689	002346	004767	000022		JSR	PC,OCTPG	
690	002352	010304			MOV	R3,R4	
691	002354	004767	000014		JSR	PC,OCTPG	;PRINT DIGIT 5
692	002360	012767	000240	176342	MOV	#240,TOB	
693	002366	004767	177576		JSR	PC,TOG	;PRINT SPACE
694	002372	000207			RTS	PC	;EXIT
695	002374	042704	177770		OCTPG:	BIC #177770,R4	
696	002400	001004			BNE	OCTPG0	
697	002402	005767	000030		TST	OFL	
698	002406	001001			BNE	OCTPG0	
699	002410	000207			RTS	PC	
700	002412	005267	000020		OCTPG0:	INC OFL	
701	002416	052704	000260		OCTPG1:	BIS #260,R4	
702	002422	010467	176302		MOV	R4,TOB	
703	002426	004767	177536		JSR	PC,TOG	
704	002432	010304			MOV	R3,R4	
705	002434	000207			RTS	PC	
706	002436	000000			OFL:	0	;FIRST CHAR FLAG
707							
708							;TTY READ SUBROUTINE*****
709							
710	002440	005077	176174		TTIN:	CLR @TKS	
711	002444	005077	176172			CLR @TKB	
712	002450	005067	176252			CLR TIB	
713	002454	005277	176160			INC @TKS	
714	002460	105777	176154		TTIN1:	TSTB @TKS	
715	002464	100375				BPL TTIN1	
716	002466	017767	176150	176232		MOV @TKB,TIB	
717	002474	105777	176144		TTIN2:	TSTB @TPS	
718	002500	100375				BPL TTIN2	
719	002502	116777	176220	176136		MOVB TIB,@TPB	
720	002510	000207				RTS PC	
721							
722	002512	057045	021507		\$CNTG:	.ASCII /%*G#/	
723	002516	051445	051127	020075	\$MSWR:	.ASCII /%SWR= #/	
724	002524	043					
725	002525	040	047040	053505	\$MNEW:	.ASCII / NEW= #/	
726	002532	020075	043				
727	002535	077	021445		\$QUEST:	.ASCII /?%#/	
728		004000				.=4000	
729		000100				.REPT 100	
730						0	
731						.ENDR	
732							

733	005000	.=5000	
734	000100	.REPT	100
735		177777	
736		.ENDR	
737			
738	000001	.END	

A	001022	DB	000622	OCTP1	002262	RWDX	001536	TPS	000644
AA	001076	DRVN	000700	OCTP2	002270	SETA	001332	TTIN	002440
AS	000616	DS	000612	OCTP3	002360	SETUP	001314	TTIN1	002460
AO	001104	DT	000626	OFL	002436	SN	000630	TTIN2	002474
A1	001122	E	001252	OPDLY	000720	START	001000	TTOUT	002054
A3	001150	ER	000614	OPDX	000722	SUSWR	001350	UDES	000702
B	001200	EO	001246	OPNUM	000724	SWR	000636	WADDR	000712
BA	000604	FC	000606	OPTBL	000740	SWREG	000176	WC	000602
BO	001174	FCNT	000704	OUT	001652	TCRLF	002106	WCNT	000706
C	001222	GO	001434	PSW	000634	TCRLFA	002124	\$CNTG	002512
CC	000620	MR	000624	RADDR	000710	TEMPST	000736	\$MNEW	002525
CKSWR	001540	OCTP	002220	RDSW	000734	TEX	002204	\$MSWR	002516
CNTLU	001612	OCTPE	002206	RDYDLY	000714	TIB	000726	\$QUEST	002535
COUNT	000732	OCTPE1	002224	RDYDX	000716	TKB	000642	\$READ	001654
CS	000610	OCTPG	002374	RESTAR	000046	TKS	000640	.	= 005200
C1	000600	OCTPG0	002412	RWDX	001440	TOB	000730		
C2	000632	OCTPG1	002416	RWDA	001510	TOG	002170		
D	001232	OCTP0	002240	RWDB	001520	TPB	000646		

. ABS. 005200 000

ERRORS DETECTED: 0

CZTEFB,CZTEFB/SOL=CZTEFB.P11  
RUN-TIME: .7 1 0 SECONDS  
RUN-TIME RATIO: 33/2=13.4  
CORE USED: 5K (9 PAGES)