TE16,TU16/77

UTILITY DRIVER
CZTEFB0

AH-B159B-MC
COPYRIGHT 77-78
FICHE 1 OF 1

JAN 1979
digital
MADE IN USA

```
.REM    %
```

IDENTIFICATION

PRODUCT CODE: AC-3158B-MC

PRODUCT NAME: CZTEFB0 TU16/TE16/TU77 UTILITY DRIVER

DATE CREATED: 21 MAY 78

MAINTAINER: DIAGNOSTIC ENGINEERING

AUTHOR: J. G. ADAMS

;TABLE OF CONTENTS

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 APPROPIATE MEMORY LOCATIONS.

2. REQUIREMENTS

2.1 HARDWARE:

A. ANY PDP-11 PROCESSOR - WITH OR WITHOUT HARDWARE SWITCH REGISTER.
B. RH MASSBUS CONTROLLER
C. TM02/TM03 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 LOACATION:

PARAMETER	LOCATION	DESCRIPTION
RH ADDRESS	600	ADDRESS OF RH (THE FIRST REGISTER ADDRESS: CS1)
DRIVE NUMBER	700	SET TO SELECT TM02/TM03 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. 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 TM02/TM03
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 PARITCULAR 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 THAT ONE (1) RECORD, SET LOCATION 1100 (8)
TO THE TWOS' COMPLIMENT OF THE NUMBER OF RECORDS DESIRED.

8. LISTING

```
.LIST   BIN,LOC,SEQ
```

```
                                        .TITLE CZTEFB0 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 TM03/TE16 CAPABILITY
                                        ;REVISED JUNE 1977 BY J. G. ADAMS
                                        ;       2)DOCUMENTATION CHANGES TO REFLECT ADDED TM03-TU77 CAPABILITY

                                        .ABS

                                    ;CONSOLF 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

000000                              R0-%0
000001                              R1=%1
000002                              R2=%2
000003                              R3=%3
000004                              R4=%4
000005                              R5=%5
000006                              SP=%6
000007                              PC=%7

000046                              .=46
000046  000170                      RESTART:        170             ;ALLOW RESTART WHEN <LF> IS PRESSED
                                                                    ;DURING CHANGING OF SWREG IF SOFTWARE SWITCH
                                                                    ;REGISTER IS USED.


                                    ;SOFTWARE SWITCH REGISTER**********

000176                              .=176
000176  000000                      SWREG:  0               ;SOFTWARE SWITCH REGISTER

                                    ;*****************************************************************************

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

                                    ;*****************************************************************************
```

```
                                                        K 1
366
367                                     ;STARTING ADDRESS
368
369             000200                  .=200
370   000200    000167   001110         JMP     SETUP
371             000600                  .=600
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           D1:    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                          .=700
402                                            ;SET PARAMETERS DESIRED FOR UNIT UNDER TEST******
403
404   000700   000000          DRVN:   0        ;DRIVE NUMBER
405   000702   002300          UDES:   2300     ;SLAVE DESCRIPTION
406   000704   177760          FCNT:   -20      ;FRAME COUNT
407   000706   177770          WCNT:   -10      ;WORD COUNT
408   000710   004000          RADDR:  4000     ;READ ADDRESS
409   000712   005000          WADDR:  5000     ;WRITE ADDRESS
410   000714   100000          RDYDLY: 100000   ;READY DELAY
411   000716   000001          RDYDX:  1        ;READY DELAY MULTIPLIER
412   000720   010000          OPDLY:  10000    ;OPERATION DELAY
413   000722   000001          OPDX:   1        ;OPERATION DELAY MULTIPLIER
414   000724   000001          OPNUM:  1        ;NUMBER OF OPERATION (1 TO 15)
415   000726   000000          TIB:    0
416   000730   000000          TOB:    0
417   000732   000000          COUNT:  0
418   000734   000000          RDSW:   0
419   000736   000000          TEMPST: 0
420
421                                            ;OPERATION TABLE**********
422                                            ;ENTER OPERATION SEQUENCE DESIRED.
423                                            ;MUST HAVE AT LEAST 1 OPERATION, AND
424                                            ;MAY HAVE UP TO 15(8).
425                                            ;SET THE OPERATION COUNTER EQUAL
426                                            ;TO THE NUMBER OF OPERATIONS IN
427                                            ;THE SEQUENCE.
428                                            ;
429                                            ;20 = READ IN PRESET
430                                            ;02 = REWIND-OFFLINE
431                                            ;06 = REWIND
432                                            ;10 = DRIVE CLEAR
433                                            ;26 = WRITE TAPE MARK
434                                            ;24 = ERASE
435                                            ;30 = SPACE FORWARD
436                                            ;32 = SPACE REVERSE
437                                            ;50 = WRITE CHECK FORWARD
438                                            ;56 = WRITE CHECK REVERSE
439                                            ;60 = WRITE FORWARD
440                                            ;70 = READ FORWARD
441                                            ;76 = READ REVERSE
442
443   000740   000060          OPTBL:  60
444   000742   000000                  0
445   000744   000000                  0
446   000746   000000                  0
447   000750   000000                  0
448   000752   000000                  0
449   000754   000000                  0        ;FILL WITH OPERATION SEQUENCE
450   000756   000000                  0
451   000760   000000                  0
452   000762   000000                  0
453   000764   000000                  0
454   000766   000000                  0
455   000770   000000                  0
```

```
                        001000                          .=1000
                                                        ;START OF PROGRAM**********

001000  012706  000500          START:  MOV     #500,SP
001004  012777  000340  177622          MOV     #340,@PSW

001012  016700  177706                  MOV     OPNUM,R0        ;SET COUNTER
001016  012701  000740                  MOV     #OPTBL,R1       ;SET POINTER
001022  012777  000040  177560  A:      MOV     #40,@CS         ;INIT
001030  016777  177644  177552          MOV     DRVN,@CS        ;DRIVE NUMBER
001036  016777  177640  177566          MOV     UDES,@C2        ;UNIT DESCRIPTION
001044  016777  177636  177530          MOV     WCNT,@WC        ;WORD COUNT
001052  016777  177626  177526          MOV     FCNT,@FC        ;FRAME COUNT
001060  012102                          MOV     (R1)+,R2        ;SET OP CODE
001062  022702  000030                  CMP     #30,R2          ;SEE IF SPACE FORWARD
001066  001403                          BEQ     AA              ;IF SO: BR
001070  022702  000032                  CMP     #32,R2          ;SEE IF SPACE REVERSE
001074  001003                          BNE     A0              ;IF NOT: BR
001076  012777  177777  177502  AA:     MOV     #-1,@FC         ;SET TO SPACE ONE RECORD
001104  022702  000060          A0:     CMP     #60,R2          ;SEE IF READ OP
001110  103404                          BLO     A1              ;IF SO: BR
001112  016777  177574  177464          MOV     WADDR,@BA       ;SET WRITE ADDRESS
001120  000413                          BR      A3
001122  016777  177562  177454  A1:     MOV     RADDR,@BA       ;SET READ ADDRESS
001130  022702  000070                  CMP     #70,R2          ;SEE IF READ OPERATION
001134  001405                          BEQ     A3              ;IF SO: BR
001136  016703  177542                  MOV     FCNT,R3         ;GET FRAME COUNT
001142  005403                          NEG     R3
001144  060377  177434                  ADD     R3,@BA          ;SET BUS ADDRESS FOR READ REVERSE
001150  052702  000001          A3:     BIS     #1,R2           ;SET GO BIT
001154  000240                          NOP
001156  000240                          NOP
001160  010277  177414                  MOV     R2,@C1          ;START OPERATION
001164  000240                          NOP
001166  000240                          NOP
001170  016704  177522                  MOV     RDYDX,R4        ;SET DELAY MULTIPLIER
001174  016703  177514          B0:     MOV     RDYDLY,R3       ;SET READY DELAY
001200  032777  000200  177404  B:      BIT     #200,@DS
001206  001005                          BNE     C               ;IF DRY: BR
001210  005303                          DEC     R3
001212  001372                          BNE     B
001214  005304                          DEC     R4
001216  001366                          BNE     B0              ;DELAY FOR DRIVE READY
001220  000240                          NOP
001222  005777  177410          C:      TST     @SWR            ;SEE IF STOP ON OPERATION
001226  100001                          BPL     D               ;IF NOT: BR
001230  000000                          HALT
001232  004767  000302          D:      JSR     PC,CKSWR        ;CHECK FOR CNTL G
001236  000240                          NOP
001240  000240                          NOP
001242  016704  177454                  MOV     OPDX,R4         ;SET DELAY MULTIPLIER
001246  016703  177446          E0:     MOV     OPDLY,R3        ;SET OPERATION DELAY
001252  005303                  E:      DEC     R3
001254  001376                          BNE     E
001256  005304                          DEC     R4
001260  001372                          BNE     E0              ;DELAY BETWEEN OPERATIONS
```

```
                                        JSR     PC,RWND         ;GO SEE IF REWIND
                                        DEC     R0
                                        BNE     A               ;IF SEQUENCE NOT DONE: BR

                                        BIT     #40000,@SWR     ;SEE IF HALT ON SEQUENCE
                                        BEQ     1$
                                        HALT
                                1$:     JSR     PC,CKSWR        ;CHECK FOR CNTL G
                                        JMP     START

                                        ;RH REGISTER SETUP**********

                                SETUP:  NOP
                                        MOV     C1,R1           ;GET ADDRESS OF CS1
                                        MOV     #15,R0          ;SET NUMBER OF REGISTERS
                                        MOV     #WC,R2          ;GET FIRST ADDRESS
                                SETA:   ADD     #2,R1           ;INCREMENT
                                        MOV     R1,(R2)+        ;LOAD ADDRESS
                                        DEC     R0              ;SEE IF DONE
                                        BNE     SETA            ;IF NOT: BR
                                        MOV     #500,SP
                                SUSWR:  MOV     @#6,-(SP)               ;SAVE VECTORS
                                        MOV     @#4,-(SP)
                                        MOV     #1$,@#4                 ;SET UP FOR TIMEOUT
                                        CMP     #-1,@SWR                ;REFERENCE HARDWARE SWITCH REGISTER
                                        BEQ     2$
                                        BR      3$
                                1$:     CMP     (SP)+,(SP)+             ;ADJUST STACK
                                2$:     MOV     #SWREG,SWR              ;POINT TO SOFTWARE SWITCH REG
                                3$:     MOV     (SP)+,@#4               ;RESTORE VECTORS
                                        MOV     (SP)+,@#6
                                        CMP     @#SWR,#SWREG            ;IS SOFTWARE REG USED
                                        BNE     GO                      ;BRANCH IF NO
                                        JSR     PC,CNTLU                ;ALLOW SOFTARE SWITCH REGISTER TO BE CHANGED
                                GO:     JMP     START           ;ELSE GO START EXECUTION

                                        ;REWIND FROM EOT (PER SW13)

                                RWND:   BIT     #20000,@SWR     ;SEE IF IGNORE EOT
                                        BNE     RWNDX           ;IF SO: BR
                                        BIT     #2000,@DS       ;SEE IF AT EOT
                                        BEQ     RWNDX           ;IF NOT: BR
                                        MOV     #40,@CS         ;INIT
                                        MOV     DRVN,@CS        ;SET DRIVE NUMBER
                                        MOV     UDES,@C2        ;SET SLAVE NUMBER
                                        MOV     #7,@C1          ;START REWIND
                                RWNDA:  BIT     #200,@DS        ;SEE IF DRY
                                        BEQ     RWNDA           ;IF NOT: BR
                                RWNDB:  BIT     #20000,@DS      ;SEE IF PIP RESET
                                        BNE     RWNDB           ;IF NOT: BR
                                        TST     (SP)+           ;RESET STACK
                                        JMP     START           ;RESTART SEQUENCE
                                RWNDX:  RTS     PC              ;RETURN
```

```
                                        ;CKSWR ROUTINE THAT ALLOWS THE LOADING OF LOC.176, SWREG**********
                                        ;FROM THE TTY AT SELECTED POINTS WITHIN THE PROGRAM**********

001540  022767  000176  177070  CKSWR:  CMP     #SWREG,SWR              ;SOFTWARE SWITCH REG PRESENT
001546  001041                          BNE     OUT                     ;NO, GET OUT
001550  105777  177064                  TSTB    @TKS                    ;YES,WAIT FOR
001554  100036                          BPL     OUT                     ;READY, GET CHARACTER
001556  017767  177060  177142          MOV     @TKB,TIB                ;AND STRIP OFF
001564  042767  177600  177134          BIC     #177600,TIB             ;THE GARBAGE
001572  022767  000007  177126          CMP     #7,TIB          ;IS IT A <^G>
001600  001024                          BNE     OUT
001602  012704  002512                  MOV     #$CNTG,R4
001606  004767  000242                  JSR     PC,TTOUT
001612  012704  002516          CNTLU:  MOV     #$MSWR,R4
001616  004767  000232                  JSR     PC,TTOUT
001622  017703  177010                  MOV     @SWR,R3
001626  004767  000354                  JSR     PC,OCTPE
001632  012704  002525                  MOV     #$MNEW,R4
001636  004767  000212                  JSR     PC,TTOUT
001642  005037  000736                  CLR     @#TEMPST
001646  004767  000002                  JSR     PC,$READ                ;GO READ A LINE
001652  000207                  OUT:    RTS     PC                      ;RETURN TO MAIN BODY OF PROGRAM

001654  005067  177056          $READ:  CLR     TEMPST
001660  012767  000007  177044          MOV     #7,COUNT
001666  004767  000546          1$:     JSR     PC,TTIN         ;GO  READ A CHARACTER
001672  042767  177600  177026          BIC     #177600,TIB     ;STRIP OFF GARBAGE
001700  122767  000025  177020          CMPB    #25,TIB         ;IS IT A ^U?
001706  001002                          BNE     2$              ;BRANCH IF NOT
001710  005726                  3$:     TST     (SP)+           ;POP THE STACK
001712  000737                          BR CNTLU                ;START OVER
001714  122767  000015  177004  2$:     CMPB    #15,TIB         ;IS IT A <CR>?
001722  001013                          BNE     4$              ;BRANCH IF NOT
001724  012767  000200  177002          MOV     #200,RDSW
001732  004767  000150                  JSR     PC,TCRLF        ;ECHO IT WITH <LF>
001736  022767  000007  176766          CMP     #7,COUNT        ;WAS IT FIRST CHARACTER
001744  001037                          BNE     7$              ;CHANGE SWR IF NOT FIRST ONE
001746  005726                  8$:     TST     (SP)+           ;POP THE STACK
001750  000740                          BR      OUT             ;GET OUT
001752  122767  000060  176746  4$:     CMPB    #60,TIB
001760  003004                          BGT     5$
001762  122767  000067  176736          CMPB    #67,TIB
001770  003005                          BGT     6$
001772  012704  002535          5$:     MOV     #$QUEST,R4
001776  004767  000052                  JSR     PC,TTOUT
002002  000742                          BR      3$              ;START OVER IF NOT LEGAL CHARACTER
002004  006367  176726          6$:     ASL     TEMPST
002010  006367  176722                  ASL     TEMPST
002014  006367  176716                  ASL     TEMPST
002020  142767  000060  176700          BICB    #60,TIB         ;GET NITTY-GRITTY
002026  156767  176674  176702          BISB    TIB,TEMPST
002034  005367  176672                  DEC     COUNT           ;ONLY WANT 6   DIGITS
002040  001754                          BEQ     5$
002042  000711                          BR      1$
002044  016777  176666  176564  7$:     MOV     TEMPST,@SWR     ;CHANGE SWITCH REGISTER CONTENTS
```

```
                                        BR      8$

                                        ;TTY OUTPUT SUBROUTINE************

002054  112467  176650        TTOUT:  MOVB    (R4)+,TOB
002060  122767  000043  176642        CMPB    #43,TOB
002066  001446                        BEQ     TEX
002070  122767  000045  176632        CMPB    #45,TOB
002076  001403                        BEQ     TCRLF
002100  004767  000064                JSR     PC,TOG
002104  000763                        BR      TTOUT
002106  112767  000015  176614  TCRLF:  MOVB    #15,TOB
002114  004767  000050                JSR     PC,TOG
002120  012703  000004                MOV     #4,R3
002124  005067  176600        TCRLFA: CLR     TOB
002130  004767  000034                JSR     PC,TOG
002134  005303                        DEC     R3
002136  001372                        BNE     TCRLFA          ;DO FILLERS
002140  112767  000012  176562        MOVB    #12,TOB
002146  004767  000016                JSR     PC,TOG
002152  105767  176556                TSTB    RDSW
002156  100401                        BMI     1$
002160  000735                        BR      TTOUT
002162  005067  176546        1$:     CLR     RDSW
002166  000406                        BR      TEX
002170  105777  176450        TOG:    TSTB    @TPS
002174  100375                        BPL     TOG
002176  116777  176526  176442        MOVB    TOB,@TPB
002204  000207                TEX:    RTS     PC

                                        ;OCTAL OUTPUT SUBROUTINE************

002206  012767  000001  000222  OCTPE:  MOV     #1,OFL
002214  010304                        MOV     R3,R4
002216  000410                        BR      OCTP0
002220  005067  000212        OCTP:   CLR     OFL             ;CLEAR FLAG FOR LEADING ZERO
002224  010304                OCTPE1: MOV     R3,R4           ;SEE IF NUMBER IS ZERO
002226  001004                        BNE     OCTP0           ;IF NOT ZERO: BR
002230  004767  000162                JSR     PC,OCTPG1       ;ELSE PRINT ZERO
002234  000167  000120                JMP     OCTP3           ;SPACE AND EXIT
002240  032704  100000        OCTP0:  BIT     #100000,R4      ;SEE IF MSD   1
002244  001406                        BEQ     OCTP1           ;IF NOT: BR
002246  012704  000001                MOV     #1,R4
002252  004767  000116                JSR     PC,OCTPG        ;PRINT 1
002256  000167  000006                JMP     OCTP2
002262  005004        OCTP1:  CLR     R4
002264  004767  000104                JSR     PC,OCTPG        ,PRINT 0
002270  010304        OCTP2:  MOV     R3,R4
002272  006004                        ROR     R4
002274  006004                        ROR     R4
002276  006004                        ROR     R4              ;POSITION DIGIT
002300  006004                        ROR     R4
002302  000304                        SWAB    R4
002304  004767  000064                JSR     PC,OCTPG        ;PRINT DIGIT 2
002310  010304                        MOV     R3,R4
```

```
                                                    D 2
  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 OCTP3: 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
```

```
                                        .=5000
                                        .REPT   100
                                        177777
                                        .ENDR

                                        .END
```

```
A       001022      DB      000622      OCTP1   002262      RWNDX   001536      TPS     000644
AA      001076      DRVN    000700      OCTP2   002270      SETA    001332      TTIN    002440
AS      000616      DS      000612      OCTP3   002360      SETUP   001314      TTIN1   002460
A0      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      E0      001246      OPNUM   000724      SWR     000636      WADDR   000712
BA      000604      FC      000606      OPTBL   000740      SWREG   000176      WC      000602
B0      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      RWND    001440      TOB     000730
C2      000632      OCTPG1  002416      RWNDA   001510      TOG     002170
D       001232      OCTP0   002240      RWNDB   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)
```