

TU16

UTILITY DRIVER
MD-11-DZTUE-D

EP-DZTUE-D-DL-A

OCT 1976

COPYRIGHT ©1976

digital

FICHE 1 OF 1

Made in U.S.A.

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

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 HA

149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204

(PAGE 2)

6. OPERATION

THE PROGRAM OPERATION IS QUITE SIMPLE, BUT DOES REQUIRE THE OPERATOR TO HAVE KNOWLEDGE OF THE TU16 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 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 IN TWOS' COMPLIMENT
WORD COUNT	706	SET NUMBER OF WORDS TO BE TRANSFERRED IN TWOS' C
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 ESTAB A MAXIMUM TIME TO AWAIT THE COMPLETION OF AN OPE BEFORE PROCEEDING TO THE NEXT. *(DEFAULT IS APP
READY MULTIPLIER	716	IF THE VALUE SET INTO 714 DOES NOT ALLOW ENOUGH INCREASE THE SIZE OF THE MULTIPLIER. EACH INCRE MULTIPLIER WILL CAUSE THE 714 DELAY TO BE EXECUT THAT MANY MORE TIMES.

(PAGE 3)

206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241

OPERATION DELAY 720

THIS DELAY IS USED TO ALLOW FOR SOME AMOUNT OF TIME BETWEEN THE EXECUTION OF EACH OPERATION. I 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(

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 OPERATIO TO BE PERFORMED IN SEQUENCE. THE NUMBER OF ENTI MAY BE FROM ONE (1) TO FIFTEEN (15). MAKE SURE THE NUMBER OF FUNCTION CODES SET IN THE TABLE IS 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

268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323

(PAGE 5)

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 AD
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
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

%
.TITLE TU16 UTILITY DRIVER
:MAINDEC-11-DZTUE-D-D
:15 FEB 75
:R. BARNES
:REVISED APRIL 1976 BY S. CARPENTER
: 1) SUPPORTS SOFTWARE SWITCH REGISTER
: 2) SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER
:ABS

```

324
325
326
327
328
329
330
331
332
333
334
335
336      000000
337      000001
338      000002
339      000003
340      000004
341      000005
342      000006
343      000007
344
345      000046
346 000046 000170
347
348
349
350
351
352
353      000176
354 000176 000000
355
356
357
358
359
360
361
362
363
364
365
366      000200
367 000200 000167 001110
368      000600
369
370
371
372 000600 172440
373 000602 172442
374 000604 172444
375 000606 172446
376 000610 172450
377 000612 172452
378 000614 172454
379 000616 172456

```

```

;CONSOLE SWITCHES
;SW 15=1(10000) 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

```

;STARTING ADDRESS

```

.=200
JMP     SETUP
.=600

```

;TMO2 REGISTERS

```

C1:  172440
WC:  172442
BA:  172444
FC:  172446
CS:  172450
DS:  172452
ER:  172454
AS:  172456

```

380 000620 172460
381 000622 172462
382 000624 172464
383 000626 172466
384 000630 172470
385 000632 172472

CC: 172460
DB: 172462
MR: 172464
DT: 172466
SN: 172470
C2: 172472

;PROCESSOR ADDRESSES

388 000634 177776
389 000636 177570

PSW: 177776 ;PROCESSOR STATUS
SWR: 177570 ;SWITCH REGISTER

;TTY REGISTERS

394 000640 177560
395 000642 177562
396 000644 177564
397 000646 177566

TKS: 177560 ;TTY READER STATUS
TKB: 177562 ;TTY READ BUFFER
TPS: 177564 ;TTY PUNCH STATUS
TPB: 177566 ;TTY PUNCH BUFFER

MO1

TU16 UTILITY DRIVER MACY11 27(1006) 16-SEP-76 18:13 PAGE 13
DZTUED.P11 16-SEP-76 15:24

454

```

455          001000          .:=1000
456                                     ;START OF PROGRAM*****
457
458 001000 012706 000500 START: MOV #500,SP
459 001004 012777 000340 177622 MOV #340,@PSW
460
461 001012 016700 177706 MOV OPNUM,R0 ;SET COUNTER
462 001016 012701 000740 MOV #OPTBL,R1 ;SET POINTER
463 001022 012777 000040 177560 A: MOV #40,@CS ;INIT
464 001030 016777 177644 177552 MOV DRVN,@CS ;DRIVE NUMBER
465 001036 016777 177640 177556 MOV UDES,@C2 ;UNIT DESCRIPTION
466 001044 016777 177636 177530 MOV WCNT,@WC ;WORD COUNT
467 001052 016777 177626 177526 MOV FCNT,@FC ;FRAME COUNT
468 001060 012102 MOV (R1)+,R2 ;SET OP CODE
469 001062 022702 000030 CMP #30,R2 ;SEE IF SPACE FORWARD
470 001066 001403 BEQ AA ;IF SO: BR
471 001070 022702 000032 CMP #32,R2 ;SEE IF SPACE REVERSE
472 001074 001003 BNE A0 ;IF NOT: BR
473 001076 012777 177777 177502 AA: MOV #-1,@FC ;SET TO SPACE ONE RECORD
474 001104 022702 000060 A0: CMP #60,R2 ;SEE IF READ OP
475 001110 103404 BLO A1 ;IF SO: BR
476 001112 016777 177574 177464 MOV WADDR,@BA ;SET WRITE ADDRESS
477 001120 000413 BR A3
478 001122 016777 177562 177454 A1: MOV RADDR,@BA ;SET READ ADDRESS
479 001130 022702 000070 CMP #70,R2 ;SEE IF READ OPERATION
480 001134 001405 BEQ A3 ;IF SO: BR
481 001136 016703 177542 MOV FCNT,R3 ;GET FRAME COUNT
482 001142 005403 NEG R3
483 001144 060377 177434 ADD R3,@BA ;SET BUS ADDRESS FOR READ REVERSE
484 001150 052702 000001 A3: BIS #1,R2 ;SET GO BIT
485 001154 000240 NOP
486 001156 000240 NOP
487 001160 010277 177414 MOV R2,@C1 ;START OPERATION
488 001164 000240 NOP
489 001166 000240 NOP
490 001170 016704 177522 MOV RDYDX,R4 ;SET DELAY MULTIPLIER
491 001174 016703 177514 B0: MOV RDYDLY,R3 ;SET READY DELAY
492 001200 032777 000200 177404 B: BIT #200,@DS
493 001206 001005 BNE C ;IF DRY: BR
494 001210 005303 DEC R3
495 001212 001372 BNE B
496 001214 005304 DEC R4
497 001216 001366 BNE B0 ;DELAY FOR DRIVE READY
498 001220 000240 NOP
499 001222 005777 177410 C: TST @SWR ;SEE IF STOP ON OPERATION
500 001226 100001 BPL D ;IF NOT: BR
501 001230 000000 HALT
502 001232 004767 000302 D: JSR PC,CKSWR ;CHECK FOR CNTL G
503 001236 000240 NOP
504 001240 000240 NOP
505 001242 016704 177454 MOV OPDX,R4 ;SET DELAY MULTIPLIER
506 001246 016703 177446 E0: MOV OPDLY,R3 ;SET OPERATION DELAY
507 001252 005303 E: DEC R3
508 001254 001376 BNE E
509 001256 005304 DEC R4
510 001260 001372 BNE E0 ;DELAY BETWEEN OPERATIONS

```

```

511 001262 004767 000152 JSR PC,RWND ;GO SEE IF REWIND
512 001266 005300 DEC RO ;
513 001270 001254 BNE A ;IF SEQUENCE NOT DONE: BR
514
515 001272 032777 040000 177336 BIT #40000,ASWR ;SEE IF HALT ON SEQUENCE
516 001300 001401 BEQ 15 ;
517 001322 000000 HALT ;
518 001304 004767 000230 15: JSR PC,CKSWR ;CHECK FOR CNTL G
519 001310 000167 177454 JMP START ;
520
521 ;RH REGISTER SETUP*****
522
523 001314 000240 SETUP: NOP ;
524 001316 016701 177256 MOV C1,R1 ;GET ADDRESS OF CS1
525 001322 012700 000015 MOV #15,RO ;SET NUMBER OF REGISTERS
526 001326 012702 000602 MOV #WC,R2 ;GET FIRST ADDRESS
527 001332 062701 000002 SETA: ADD #2,R1 ;INCREMENT
528 001336 010122 MOV R1,(R2)+ ;LOAD ADDRESS
529 001340 005300 DEC RO ;SEE IF DONE
530 001342 001373 BNE SETA ;IF NOT: BR
531 001344 012706 000500 MOV #500,SP ;
532 001350 013746 000006 SUSWR: MOV @#6,-(SP) ;SAVE VECTORS
533 001354 013746 000004 MOV @#4,-(SP) ;
534 001360 012737 001400 000004 MOV #15,@#4 ;SET UP FOR TIMEOUT
535 001366 022777 177777 177242 CMP #-1,ASWR ;REFERENCE HARDWARE SWITCH REGISTER
536 001374 001402 BEQ 25 ;
537 001376 000404 BR 35 ;
538 001400 022626 15: CMP (SP)+,(SP)+ ;ADJUST STACK
539 001402 012767 000176 177226 25: MOV #SWREG,SWR ;POINT TO SOFTWARE SWITCH REG
540 001410 012637 000004 35: MOV (SP)+,@#4 ;RESTORE VECTORS
541 001414 012637 000006 MOV (SP)+,@#6 ;
542 001420 023727 000636 000176 CMP @#SWR,#SWREG ;IS SOFTWARE REG USED
543 001426 001002 BNE GO ;BRANCH IF NO
544 001430 004767 000156 JSR PC,CNTLU ;ALLOW SOFTWARE SWITCH REGISTER TO BE CHANGED
545 001434 000167 177340 GO: JMP START ;ELSE GO START EXECUTION
546
547 ;REWIND FROM EOT (PER SW13)
548
549 001440 032777 020000 177170 RWND: BIT #20000,ASWR ;SEE IF IGNORE EOT
550 001446 001033 BNE RWNOX ;IF SO: BR
551 001450 032777 002000 177134 BIT #2000,ADS ;SEE IF AT EOT
552 001456 001427 BEQ RWNOX ;IF NOT: BR
553 001460 012777 000040 177122 MOV #40,ACS ;INIT
554 001466 016777 177206 177114 MOV DRVN,ACS ;SET DRIVE NUMBER
555 001474 016777 177202 177130 MOV UDES,ACS ;SET SLAVE NUMBER
556 001502 012777 000007 177070 MOV #7,AC1 ;START REWIND
557 001510 032777 000200 177074 RWNDA: BIT #200,ADS ;SEE IF DRY
558 001516 001774 BEQ RWNDA ;IF NOT: BR
559 001520 032777 020000 177064 RWNDB: BIT #20000,ADS ;SEE IF PIP RESET
560 001526 001374 BNE RWNDB ;IF NOT: BR
561 001530 005726 TST (SP)+ ;RESET STACK
562 001532 000167 177242 JMP START ;RESTART SEQUENCE
563 001536 000207 RWNDX: RTS PC ;RETURN
564
565
566

```

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

```

567
568 001540 022767 000176 177070 CKSWR: CMP #SWREG,SWR ;SOFTWARE SWITCH REG PPRESENT
569 001546 001041 BNE OUT ;NO GET OUT
570 001550 105777 177064 TSTB JTKS ;YES WAIT FOR
571 001554 100036 BPL OUT ;READY GET CHARACTER
572 001556 017767 177060 177142 MOV JTKB,TIB ;AND STRIP OFF
573 001564 042767 177600 177134 BIC #177600,TIB ;THE GARBAGE
574 001572 022767 000007 177126 CMP #7,TIB ;IS IT A <IG>
575 001600 001024 BNE OUT
576 001602 012704 002512 MOV #SCNTG,R4
577 001606 004767 000242 JSR PC,TTOUT
578 001612 012704 002516 CNTLU: MOV #MSWR,R4
579 001616 004767 000232 JSR PC,TTOUT
580 001622 017703 177010 MOV JSWR,R3
581 001626 004767 000354 JSR PC,OCPE
582 001632 012704 002525 MOV #SMNEW,R4
583 001636 004767 000212 JSR PC,TTOUT
584 001642 005037 000736 CLR J#TEMPST
585 001646 004767 000002 JSR PC,$READ ;GO READ A LINE
586 001652 000207 OUT: RTS ;RETURN TO MAIN BODY OF PROGRAM
587
588 001654 005067 177056 $READ: CLR TEMPST
589 001660 012767 000007 177044 MOV #7,COUNT
590 001666 004767 000546 1$: JSR PC,TTIN ;GO READ A CHARACTER
591 001672 042767 177600 177026 BIC #177600,TIB ;STRIP OFF GARBAGE
592 001700 122767 000025 177020 CMPB #25,TIB ;IS IT A ↑U?
593 001706 001002 BNE 2$ ;BRANCH IF NOT
594 001710 005726 3$: TST (SP)+ ;POP THE STACK
595 001712 000737 BR CNTLU ;START OVER
596 001714 122767 000015 177004 2$: CMPB #15,TIB ;IS IT A <CR>?
597 001722 001013 BNE 4$ ;BRANCH IF NOT
598 001724 012767 000200 177002 MOV #200,RDSW
599 001732 004767 000150 JSR PC,TCRLF ;ECHO IT WITH <LF>
600 001736 022767 000007 176766 CMP #7,COUNT ;WAS IT FIRST CHARACTER
601 001744 001037 BNE 7$ ;CHANGE SWR IF NOT FIRST ONE
602 001746 005726 8$: TST (SP)+ ;POP THE STACK
603 001750 000740 BR OUT ;GET OUT
604 001752 122767 000060 176746 4$: CMPB #60,TIB
605 001760 003004 BGT 5$
606 001762 122767 000067 176736 CMPB #67,TIB
607 001770 003005 BGT 6$
608 001772 012704 002535 5$: MOV #SQUEST,R4
609 001776 004767 000052 JSR PC,TTOUT
610 002002 000742 BR 3$ ;START OVER IF NOT LEGAL CHARACTER
611 002004 006367 176726 6$: ASL TEMPST
612 002010 006367 176722 ASL TEMPST
613 002014 006367 176716 ASL TEMPST
614 002020 142767 000060 176700 BICB #60,TIB ;GET NITTY-GRITTY
615 002026 156767 176674 176702 BICB TIB,TEMPST
616 002034 005367 176672 DEC COUNT ;ONLY WANT 6 DIGITS
617 002040 001754 BEQ 5$
618 002042 000711 BR 1$
619 002044 016777 176666 176564 7$: MOV TEMPST,JSWR ;CHANGE SWITCH REGISTER CONTENTS
620 002052 000735 BR 8$
621
622

```

```

623                                     :TTY OUTPUT SUBROUTINE*****
624
625 002054 112467 176650 TTYOUT: MOVB (R4)+,TOB
626 002060 122767 000043 176642 CMPB #43,TOB
627 002066 001446 BEQ TEX
628 002070 122767 000045 176632 CMPB #45,TOB
629 002076 001403 BEQ TCRLF
630 002100 004767 000064 JSR PC,TOG
631 002104 000763 BR TTYOUT
632 002106 112767 000015 176614 TCRLF: MOVB #15,TOB
633 002114 004767 000050 JSR PC,TOG
634 002120 012703 000004 MOV #4,R3
635 002124 005067 176600 TCRLFA: CLR TOB
636 002130 004767 000034 JSR PC,TOG
637 002134 005303 DEC R3
638 002136 001372 BNE TCRLFA ;DO FILLERS
639 002140 112767 000012 176562 MOVB #12,TOB
640 002146 004767 000016 JSR PC,TOG
641 002152 105767 176556 TSTB RDSW
642 002156 109401 BMI IS
643 002160 000735 BR TTYOUT
644 002162 005067 176546 IS: CLR RDSW
645 002166 000406 BR TEX
646 002170 105777 176450 TOG: TSTB @TPS
647 002174 100375 BPL TOG
648 002176 116777 176526 176442 MOVB TOB,@TPB
649 002204 000207 TEX: RTS PC

```

```

650                                     :OCTAL OUTPUT SUBROUTINE*****
651
652 002206 012767 000001 000222 OCTPE: MOV #1,OFL
653 002214 010304 MOV R3,R4
654 002216 000410 BR OCTPO
655 002220 005067 000212 OCTP: CLR OFL ;CLEAR FLAG FOR LEADING ZERO
656 002224 010304 OCTPE1: MOV R3,R4 ;SEE IF NUMBER IS ZERO
657 002226 001004 BNE OCTPO ;IF NOT ZERO: BR
658 002230 004767 000162 JSR PC,OCTPG1 ;ELSE PRINT ZERO
659 002234 000167 000120 JMP OCTP3 ;SPACE AND EXIT
660 002240 032704 100000 OCTPO: BIT #100000,R4 ;SEE IF MSD = 1
661 002244 001406 BEQ OCTP1 ;IF NOT: BR
662 002246 012704 000001 MOV #1,R4
663 002252 004767 000116 JSR PC,OCTPG ;PRINT 1
664 002256 000167 000006 JMP OCTP2
665 002262 005004 OCTP1: CLR R4
666 002264 004767 000104 JSR PC,OCTPG ;PRINT 0
667 002270 010304 OCTP2: MOV R3,R4
668 002272 006004 ROR R4
669 002274 006004 ROR R4
670 002276 006004 ROR R4 ;POSITION DIGIT
671 002300 006004 ROR R4
672 002302 000304 SWAB R4
673 002304 004767 000064 JSR PC,OCTPG ;PRINT DIGIT 2
674 002310 010304 MOV R3,R4
675 002312 006004 ROR R4
676 002314 000304 SWAB R4
677 002316 004767 000052 JSR PC,OCTPG ;PRINT DIGIT 3

```

```

679 002322 010304      MOV      R3,R4
680 002324 006104      ROL      R4
681 002326 006104      ROL      R4
682 002330 000304      SWAB     R4
683 002332 004767 000036      JSR      PC,OCTPG      ;PRINT DIGIT 4
684 002336 010304      MOV      R3,R4
685 002340 006004      ROR      R4
686 002342 006004      ROR      R4
687 002344 006004      ROR      R4
688 002346 004767 000022      JSR      PC,OCTPG
689 002352 010304      MOV      R3,R4
690 002354 004767 000014      JSR      PC,OCTPG      ;PRINT DIGIT 5
691 002360 012767 000240 176342 OCTPG3: MOV      #240,T0B
692 002366 004767 177576      JSR      PC,T0G      ;PRINT SPACE
693 002372 000207      RTS      PC      ;EXIT
694 002374 042704 177770      OCTPG:  BIC      #177770,R4
695 002400 001004      BNE      OCTPG0
696 002402 005767 000030      TST      OFL
697 002406 001001      BNE      OCTPG0
698 002410 000207      RTS      PC
699 002412 005267 000020      OCTPG0: INC      OFL
700 002416 052704 000260      OCTPG1: BIS      #260,R4
701 002422 010467 176302      MOV      R4,T0B
702 002426 004767 177536      JSR      PC,T0G
703 002432 010304      MOV      R3,R4
704 002434 000207      RTS      PC
705 002436 000000      OFL:    0      ;FIRST CHAR FLAG
706
707      ;TTY READ SUBROUTINE*****
708
709 002440 005077 176174      TTIN:   CLR      @TKS
710 002444 005077 176172      CLR      @TKB
711 002450 005067 176252      CLR      TIB
712 002454 005277 176160      INC      @TKS
713 002460 105777 176154      TTIN1:  TSTB   @TKS
714 002464 100375      BPL      TTIN1
715 002466 017767 176150 176232      MOV      @TKB,TIB
716 002474 105777 176144      TTIN2:  TSTB   @TPS
717 002500 100375      BPL      TTIN2
718 002502 116777 176220 176136      MOVB    TIB,@TPB
719 002510 000207      RTS      PC
720
721 002512 057045 021507      $CNTG:  .ASCII  /%#G#/
722 002516 051445 051127 020075      $MSWR:  .ASCII  /%SWR= #/
723 002524      043
724 002525      040 047040 053505      $MNEW:  .ASCII  / NEW= #/
725 002532 020075      043
726 002535      077 021445      $QUEST: .ASCII  /?%#/
727      004000      .=4000
728      000100      .REPT  100
729      0
730      .ENDR
731
732      005000      .=5000
733      000100      .REPT  100
734      177777

```

F02

TU16 UTILITY DRIVER MACY11 27(1006) 16-SEP-76 18:13 PAGE 19
DZTUED.P11 16-SEP-76 15:24

735
736
737

000001

.ENDR

.END

A	001022	463#	513						
AA	001076	470	473#						
AS	000616	379#							
AO	001104	472	474#						
AI	001122	475	478#						
A3	001150	477	480	484#					
B	001200	492#	495						
BA	000604	374#	476*	478*	483*				
BO	001174	491#	497						
C	001222	493	499#						
CC	000620	380#							
CKSWR	001540	502	518	568#					
CNTLU	001612	544	578#	595					
COUNT	000732	414#	589#	600	616*				
CS	000610	376#	463*	464*	553*	554*			
CI	000600	372#	487*	524	556*				
C2	000632	385#	465*	555*					
D	001232	500	502#						
DB	000622	381#							
DRVN	000700	401#	464	554					
DS	000612	377#	492	551	557	559			
DT	000626	383#							
F	001252	507#	508						
ER	000614	378#							
EO	001246	506#	510						
FC	000606	375#	467*	473*					
FCNT	000704	403#	467	481					
GO	001434	543	545#						
MR	000624	382#							
OCTP	002220	656#							
OCTPE	002206	581	653#						
OCTPE1	002224	657#							
OCTPG	002374	664	667	674	678	683	688	690	694#
OCTPG0	002412	695	697	699#					
OCTPG1	002416	659	700#						
OCTP0	002240	655	658	661#					
OCTP1	002262	662	666#						
OCTP2	002270	665	668#						
OCTP3	002360	660	631#						
OFL	002436	653*	656*	696	699*	705#			
OPDLY	000720	409#	506						
OPDX	000722	410#	505						
OPNUM	000724	411#	461						
OPTBL	000740	440#	462						
OUT	001652	569	571	575	586#	603			
PSW	000634	389#	459*						
RADDR	000710	405#	478						
RDSW	000734	415#	598*	641	644*				
RDYDLY	000714	407#	491						
RDYDX	000716	408#	490						
RESTAR	000046	346#							
RWNO	001440	511	549#						
RWNOA	001510	557#	558						
RWNOB	001520	559#	560						
RWNOX	001536	550	552	562#					
SETA	001332	527#	530						

SETUP	001314	367	523*																	
SN	000630	394*																		
START	001000	458*	519	545	562															
SUSWR	001350	532*																		
SWR	000636	390*	499	515	535	539*	542	549	568	580	619*									
SWREG	000176	354*	539	542	568															
TCRLF	002106	599	629	632*																
TCRLFA	002124	635*	638																	
TEMPST	000736	416*	584*	588*	611*	612*	613*	615*	619											
TEX	002204	627	645	649*																
TIB	000726	412*	572*	573*	574	591*	592	596	604	606	614*	615	711*	715*						
		718																		
TKB	000642	395*	572	710*	715															
TKS	000640	394*	570	709*	712*	713														
TJB	000730	413*	625*	626	629	632*	635*	639*	648	691*	701*									
TOG	002170	630	633	636	640	646*	647	692	702											
TP9	000646	397*	648*	718*																
TFS	000644	396*	646	716																
TTIN	002440	590	709*																	
TTIN1	002460	713*	714																	
TTIN2	002474	716*	717																	
TTOUT	002054	577	579	583	609	625*	631	643												
UDES	000702	402*	465	555																
WADDR	000712	406*	476																	
WC	000602	373*	466*	526																
WCNT	000706	404*	466																	
\$CNTG	002512	576	721*																	
\$MNEW	002525	582	724*																	
\$MSWR	002516	578	722*																	
\$QUEST	002535	608	726*																	
\$READ	001654	585	588*																	
.	= 005200	345*	353*	366*	368*	398*	455*	727*	732*											

. ABS. 005200 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DZTUED, DZTUED.SEG/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DZTUED.P11
RUN-TIME: 24.4 SECONDS
RUN-TIME RATIO: 22/7=2.8
CORE USED: 6K (11 PAGES)

