

DUP11

CONFIDENCE TEST
MD-11-DZDPE-A

EP-DZDPE-A-DL-A

COPYRIGHT © 1976

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA


```

.REM      2

```

PRODUCT CODE: MAINDEC-11-DZDPE-A-D
PRODUCT NAME: DUPII CONFIDENCE TEST
DATE: DEC 75
MAINTAINER: DIAGNOSTICS

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) 1975 BY DIGITAL EQUIPMENT CORPORATION

പ്രതികരണം

1.0 ABSTRACT

THE FUNCTION OF THE MAINDEC IS TO PROVIDE A LEVEL OF CONFIDENCE IN THE OPERATION OF THE DUP-11 WITHOUT CHANGING JUMPERS OR SWITCHES FROM CUSTOMER CONFIGURATION.

THE OPTION IS TESTED IN SOLC MODE (BIT-STUFFING), THEN IN DEC MODE USING A SIMULATED DDMP-LINE PROTOCOL WITH AN IMBEDDED CRC CHARACTER. BOTH OF THESE MODES WILL BE TESTED OVER A CABLE IF A TURNAROUND IS POSSIBLE.

THE MODEM CONTROL LEADS WILL ALSO BE TESTED, IF THE DUP-11 IS CONFIGURED CORRECTLY. THE DETERMINATION OF WHAT WILL BE TESTED IS DONE BY ANSWERING A "PARAMETER DIALOG" (LOAD ADDRESS 200, START ADDRESS 0 OR 1.) ALL QUESTIONS MUST BE ANSWERED. IF AN ERROR SHOULD OCCUR, A TIMEOUT WILL EXPLAIN THE FUNCTIONAL AREA OF THE DEVICE WHICH FAILED. TO REPAIR THE OPTION, THE DIAGNOSTICS WILL HAVE TO BE RUN.

CURRENTLY THERE ARE THREE OFF-LINE DIAGNOSTICS THAT ARE TO BE RUN IN SEQUENCE TO ENSURE THAT IF AN ERROR SHOULD OCCUR IT WILL BE DETECTED AT AN EARLY STAGE AND ESTABLISH THAT DIAGNOSIS OF THE ERROR WILL BE IMMEDIATE TO DISCOVERING THE PROBLEM.

NOTE: ADDITIONAL DIAGNOSTICS MAY BE ADDED IN THE FUTURE.

THE THREE DIAGNOSTICS ARE:

1. DZDPB (REV) BASIC AND OFFLINE TRANSMITTER TESTS.
2. DZDPC (REV) OFFLINE RECEIVER TESTS, MODEM CONTROL AND INTERRUPT TESTS
3. DZDPD (REV) OFFLINE SOLC DATA AND FUNCTION OFFLINE DECMODE DATA AND FUNCTION TESTS

NOTE: THE FOURTH TAPE IS:

4. DZDPE (REV) THIS CONFIDENCE TEST.

45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85

2.0 REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 4K MEMORY)
ASR 33 (OR EQUIVALENT)
DUP11

2.2 STORAGE

PROGRAM WILL USE ALL 4K OF MEMORY EXCEPT WHERE ABS AND
BOOTSTRAP LOADER RESIDE. LOCATION 1500 THRU 1560 ARE
ESPECIALLY TO BE NOTED AND LEFT UNTOUCHED BY THE OPERATOR
AFTER THE DUP11 PARAMETER DIALOG HAS BEEN EXECUTED OR AFTER
THE DEFAULT SETUP HAS BEEN DONE.

3.0 LOADING PROCEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND ARE LOADED USING THE
ABSOLUTE LOADER. NOTE: IF THE DIAGNOSTICS ARE ON A MEDIA
SUCH AS DISK, MAGTAPE, DECTAPE, OR CASSETTE FOLLOW
INSTRUCTIONS FOR THE MONITOR WHICH HAS BEEN PROVIDED ON THAT
SPECIFIC MEDIA.

ABSOLUTE LOADER STARTING ADDRESS = *+500

MEMORY SIZE

	(*)=
4K	17
8K	37
12K	57
16K	77
20K	117
24K	137
28K	157

3.1.1 PLACE ADDRESS OF ABS LOADER INTO SWITCH REGISTER. (ALSO PLACE
'HALT' SW UP)

3.1.2 DEPRESS 'LOAD ADDRESS' KEY ON CONSOLE AND RELEASE.

3.1.3 DEPRESS 'START KEY' ON CONSOLE AND RELEASE (PROGRAM SHOULD NOW
BE LOADING INTO CPU)

4.0 STARTING PROCEEDURE

- A. SET SWITCH REGISTER TO 000200
- B. DEPRESS 'LOAD ADDRESS' KEY AND RELEASE
- C. LEAVE SWR BIT 7=1 TO USE EXISTING PARAMETERS PREVIOUSLY SET UP BY THE DUP11 PARAMETER DIALOG. SET SWR 0 OR 1 TO GO THROUGH THE PARAMETER DIALOG. (*)
- D. DEPRESS 'START KEY' AND RELEASE. THE PROGRAM WILL TYPE MAINDEC NAME AND PROGRAM NAME (IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO THE FOLLOWING:

'EXAMPLE'

'MAP OF DUP11 STATUS'

1500	160050	CSR OF DUP11
1502	000300	VECTOR OF DUP11

THE ABOVE IS ONLY AN EXAMPLE! 'THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADDRESS 1500 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE USER.

IT IS POSSIBLE FOR THE OPERATOR TO MANUALLY CHANGE (TOGGLE IN) THE INFORMATION IN THE MAP TO SUIT A SPECIFIC CONFIGURATION, BUT THE RESPONSIBILITY FOR VERIFYING THAT INFORMATION RESTS WITH THE OPERATOR.

THE PROGRAM WILL TYPE 'R' AND PROCEED TO RUN THE MAINDEC

4.1 CONTROL SWITCH SETTINGS

SW 15	SET:	HALT ON ERROR
SW 14	SET:	LOOP ON CURRENT TEST
SW 13	SET:	INHIBIT ERROR PRINT OUT
SW 12	SET:	INHIBIT TYPE OUT/BELL ON ERROR.
SW 11	SET:	INHIBIT ITERATIONS. (QUICK PASS)
SW 10	SET:	ESCAPE TO NEXT TEST ON ERROR
SW 09	SET:	RESERVED
SW 08	SET:	CATCH ERROR AND LOOP ON IT
SW 07	SET:	'USE PREVIOUS STATUS TABLE.
SW 06	SET:	RESERVED
SW 05	SET:	RESERVED
SW 04	SET:	RESERVED
SW 03	SET:	RESERVED
SW 02	SET:	LOCK ON SELECTED TEST
SW 01	SET:	RESTART PROGRAM AT SELECTED TEST
SW 00	SET:	ENTER PARAMETERS USING MANUAL DIALOG

(*) THIS MUST BE DONE EVERY TIME A NEW DEVICE IS TESTED.

F01

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 6
DZDPER.CMB

192

PARAMETERS MUST BE CHANGED TO TEST MORE THAN ONE DEVICE.

SWITCHES 8 THROUGH 15 ARE DYNAMIC AND SHOULD BE USED AS NEEDED IN THE DIAGNOSTIC. SWITCHES 0 THROUGH 2 ARE STATIC (ONLY ARE OPERABLE WHEN THE MONITOR PORTION OF THE TAPE IS RUNNING) AND SHOULD BE SET UP PRIOR TO STARTING OR RESTARTING THE DIAGNOSTIC.

4.1.2 SWITCH REGISTER RESTRICTIONS

SW 01 RESTART PROGRAM AT SELECTED TEST. IT IS STRONGLY SUGGESTED THAT AT LEAST ONE PASS HAS BEEN MADE BEFORE TRYING TO SELECT A TEST THAT IS NOT IN THE ORDER OF SEQUENCE. THE REASON FOR THIS IS THAT THE PROGRAM HAS TO CLEAR AREAS AND SET UP PARAMETERS IN THE MONITOR PORTION OF THE PROGRAM. IT IS POSSIBLE TO LD200, AND RAISE SW01, THEN START, PROVIDED PARAMETERS HAVE BEEN PREVIOUSLY SET UP AS DESCRIBED IN SECTION 4.0. ALSO, WHEN A TEST IS SELECTED, ALWAYS START AT THE VERY BEGINNING OF THAT TEST.

4.1.3 SWITCH REGISTER PRIORITIES

A) ERROR SWITCHES

1. SW 12 DELETE PRINT OUT/BELL ON ERROR.
2. SW 13 DELETE ERROR PRINTOUT.
3. SW 15 HALT ON THE ERROR.
4. SW 08 GOTO BEGINNING OF THE TEST(ON ERROR).
5. SW 10 GOTO NEXT TEST(ON ERROR).

B) SCOPE SWITCHES

1. SW 14 - LOOP ON TEST. WILL LOOP ON TEST UNTIL SWITCH IS LOWERED.
2. SW 11 - INHIBIT ITERATIONS (QUICK PASS). ALLOWS ONLY ONE PASS THROUGH A TEST.

4.2 STARTING ADDRESS

STARTING ADDRESS IS AT 000200. THERE ARE NO OTHER STARTING ADDRESSES FOR THE DUP11 DIAGNOSTICS OR THIS EXERCISER.

NOTE: IF ADDRESS 000042 IS NON-ZERO THE PROGRAM ASSUMES IT IS UNDER ACT11 OR XXDP CONTROL AND WILL ACT ACCORDINGLY. AFTER DUP11 IS TESTED, THE PROGRAM WILL RETURN TO 'XXDP' OR 'ACT-11'.

193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241

5.0 OPERATING PROCEDURE

WHEN THE PROGRAM IS INITIALLY STARTED MESSAGES AS DESCRIBED IN SECTION FOUR WILL BE PRINTED AND PROGRAM WILL BEGIN RUNNING THE DIAGNOSTIC.

5.1 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

1. HALT ON ERROR (VIA SW 15=1) WHENEVER AN ERROR OCCURS.
2. CLEAR SW 15.

THE TEST NUMBER AND PC WILL BE TYPED OUT AND POSSIBLY AN ERROR MESSAGE (THIS DEPENDS ON THE TEST), TO GIVE THE OPERATOR AN IDEA AS TO THE SOURCE OF THE PROBLEM. IF IT IS NECESSARY TO KNOW MORE INFORMATION CONCERNING THE ERROR REPORT, LOOK IN THE LISTING FOR THAT TEST NUMBER WHICH WAS TYPED OUT AND THEN NOTE THE PC OF THE ERROR REPORT. IN THIS WAY THE EXACT FUNCTIONING OF THE TEST CAN BE INTERPRETED SINCE THE ERROR PC IS THE HLT+2 LOCATION.

AT THIS POINT IT IS RECOMMENDED THAT THE NORMAL DIAGNOSTICS BE RUN TO ISOLATE THE ERROR CONDITION.

6.0 ERRORS

AS DESCRIBED PREVIOUSLY THERE WILL ALWAYS BE A TEST NUMBER AND PC TYPED OUT AT THE TIME OF AN ERROR (PROVIDING SW 13=0 AND SW 12=0). ADDITIONAL INFORMATION WILL BE SUPPLIED TO THE ERROR MESSAGE WHICH IS TO GIVE THE OPERATOR AN INDICATION OF THE ERROR.

6.1 ERROR RECOVERY

IF FOR SOME REASON THE DUP11 SHOULD 'HANG THE BUS' (GAIN CONTROL OF BUS SO THAT CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT OR POWER DOWN/UP IS NECESSARY FOR OPERATOR TO REGAIN CONTROL OF CPU. IF THIS SHOULD HAPPEN LOOK IN LOCATION 'TSTNO' FOR THE NUMBER OF THE TEST THAT WAS RUNNING AT THE TIME OF THE CATASTROPHIC ERROR. THIS GIVES THE OPERATOR SOME IDEA AS TO WHAT THE DUP11 WAS DOING AT THE TIME OF THE ERROR.

242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286

7.0 RESTRICTIONS

7.1 STARTING RESTRICTIONS

SEE SECTION 4 (PLEASE). STATUS TABLE SHOULD BE VERIFIED REGARDLESS OF HOW THE PROGRAM WAS STARTED.

7.2 OPERATING RESTRICTIONS

DUP11 PARAMETER DIALOG MUST BE ANSWERED BEFORE RUNNING THIS MAINDEX. THERE ARE NO DEFAULT PARAMETERS FOR THIS MAINDEX. IF MORE THAN ONE DEVICE IS TO BE TESTED THERE ARE TWO OPTIONS FOR THE USER:

1. ANSWER THE COMPLETE PARAMETER DIALOG AGAIN.
2. TOGGLE IN THE NEW CSR AT 1500 AND THE VECTOR AT 1502, THEN RESTART THE PROGRAM WITH SW07=1.

8.0 MISCELLANEOUS

8.1 EXECUTION TIME

ALL DUP11 DEVICE DIAGNOSTICS WILL GIVE AN 'END PASS' MESSAGE (PROVIDING NO ERRORS AND SW12=0) WITHIN 4 MINS. THIS IS ASSUMING SW11=1 (DELETE ITERATIONS) IS SET TO GIVE THE FASTEST POSSIBLE EXECUTION. THE ACTUAL EXECUTION TIME DEPENDS GREATLY ON THE PDPI1 CPU CONFIGURATION.

8.2 PASS COMPLETE

NOTE: *EVERY* TIME THE PROGRAM IS STARTED, THE TESTS WILL RUN AS IF SW11 (DELETE ITERATIONS) WAS UP (=1). THIS IS TO VERIFY NO *HARD* ERRORS AS SOON AS POSSIBLE. THEREFORE THE FIRST PASS--EACH TIME PROGRAM IS STARTED--WILL BE A 'QUICK PASS' UNTIL ALL DUP11'S IN SYSTEM ARE TESTED. WHEN THE DIAGNOSTIC HAS COMPLETED A PASS WITH THE NORMAL ITERATION COUNT (ICOUNT=50), THE FOLLOWING IS AN EXAMPLE OF THE PRINT OUT TO BE EXPECTED.

END PASS DZDPBA CSR:160050 VEC:300 PASSES:000001 ERRORS:000000

NOTE: THE NUMBERS FOR CSR AND VEC ARE NOT NECESSARILY THE VALUES FOR THE DEVICE. THEY ARE ONLY FOR THIS EXAMPLE.

8.3 KEY LOCATIONS

RETURN CONTAINS THE ADDRESS WHERE PROGRAM WILL RETURN WHEN ITERATION COUNT IS REACHED OR IF LOOP ON TEST IS ASSERTED.

NEXT CONTAINS THE ADDRESS OF THE NEXT TEST TO BE PERFORMED.

TSTNO CONTAINS THE NUMBER OF THE TEST NOW BEING PERFORMED.

RUN THE BIT IN 'RUN' ALWAYS POINTS ONE PAST THE DUP11 CURRENTLY BEING TESTED.

DUPCR00-DUPCR07
(1500)-(1560) THESE LOCATIONS CONTAIN THE INFORMATION NEEDED TO TEST UP TO 8 (DECIMAL) DUP11S SEQUENTIALY (USED ONLY IN THE DIAGNOSTICS, NOT IN THIS CONFIDENCE TEST). THEY CONTAIN THE CSR, VECTOR AND STATUS CONCERNING THE CONFIGURATION OF EACH DUP11.

RXCSR CONTAINS THE RECEIVER CSR OF THE DUP11 UNDER TEST.

8.4 MORE ON THAT 'STATUS TABLE' (1500-1560)

'MAP OF DUP11 STATUS'

	1500	160050	
	1502	000300	
1500	160050	THIS IS THE SYSTEM CONTROL REGISTER FOR THE 1ST DUP11 IN THE SYSTEM.	
1502	000300	THIS IS VECTOR 'A' FOR THE FIRST DUP11 IN THE SYSTEM.	

THE ABOVE IS REPEATED FOR EACH DUP11 IN THE SYSTEM. THE TABLE IS FILLED BY THE MANUAL PARAMETER INPUT AS DESCRIBED PREVIOUSLY. ALSO, IF DESIRED BY THE USER - THE LOCATIONS MAY BE ALTERED BY HAND (TOGGLED IN) TO SUIT THE SPECIFIC CONFIGURATION, THUS MAKING EACH DEVICE MAP DIFFERENT. IT IS THE RESPONSIBILITY OF THE OPERATOR TO VERIFY THE DATA IN THE MAP.

K01

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 11
DZDPEA.CMB INTRODUCTION TO DUP11 DIAGNOSTIC

```

385      ;*MAINDEC-11-DZDPE-A /(<377>)/DUP11 QUICK-VERIFY EXERCISER
386      ;*COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
387      ;*-----
388
389      ;STARTING PROCEDURE
390      ;LOAD PROGRAM
391      ;LOAD ADDRESS 000200
392      ;PRESS START
393      ;PROGRAM WILL TYPE "MAINDEC-11-DZDPE-A /(<377>)/DUP11 QUICK-VERIFY EXERCISER "
394      ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
395      ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
396      ;AND THEN RESUME TESTING
397
398
399      ;SWITCH REGISTER OPTIONS
400      ;-----
401
402      100000      SW15=100000      ;=1, HALT ON ERROR
403      040000      SW14=40000       ;=1, LOOP ON CURRENT TEST
404      020000      SW13=20000       ;=1, INHIBIT ERROR TYPEOUT
405      010000      SW12=10000       ;=1, DELETE TYPEOUT/BELL ON ERROR.
406      004000      SW11=4000        ;=1, INHIBIT ITERATIONS
407      002000      SW10=2000        ;=1, ESCAPE TO NEXT TEST ON ERROR
408      001000      SW09=1000        ;=1, LOOP WITH CURRENT DATA
409      000400      SW08=400          ;=1, LOOP ON ERROR
410      000200      SW07=200
411      000100      SW06=100
412      000040      SW05=40
413      000020      SW04=20
414      000010      SW03=10
415
416      000004      SW02=4
417      000002      SW01=2
418      000001      SW00=1

```

```

;SELECT DUP'S DESIRED ACTIVE
;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT
;LOCK ON TEST SELECT
;RESTART PROGRAM AT SELECTED TEST
;ENTER PARAMETERS

```

419			
420			
421			
422			
423			
424	000000	R0=%0	; GENERAL REGISTER
425	000001	R1=%1	; GENERAL REGISTER
426	000002	R2=%2	; GENERAL REGISTER
427	000003	R3=%3	; GENERAL REGISTER
428	000004	R4=%4	; GENERAL REGISTER
429	000005	R5=%5	; GENERAL REGISTER
430	000006	SP=%6	; PROCESSOR STACK POINTER
431	000007	PC=%7	; PROGRAM COUNTER
432			
433			
434			
435			
436	177776	PS=177776	; PROCESSOR STATUS WORD
437	001150	STACK=1150	; START OF PROCESSOR STACK
438			
439			
440			
441			
442	005746	PUSH1SP=5746	; DECREMENT PROCESSOR STACK 1 WORD
443	005726	POP1SP=5726	; INCREMENT PROCESSOR STACK 1 WORD
444	010046	PUSHR0=10046	; SAVE R0 ON STACK
445	012600	POPR0=12600	; RESTORE R0 FROM STACK
446	024646	PUSH2SP=24646	; DECREMENT STACK TWICE
447	022626	POP2SP=22626	; INCREMENT STACK TWICE
448		.EQUIV EMT,HLT	; BASIC DEFINITION OF ERROR CALL
449			
450			
451	100000	BIT15=100000	
452	040000	BIT14=40000	
453	020000	BIT13=20000	
454	010000	BIT12=10000	
455	004000	BIT11=4000	
456	002000	BIT10=2000	
457	001000	BIT9=1000	
458	000400	BIT8=400	
459	000200	BIT7=200	
460	000100	BIT6=100	
461	000040	BIT5=40	
462	000020	BIT4=20	
463	000010	BIT3=10	
464	000004	BIT2=4	
465	000002	BIT1=2	
466	000001	BIT0=1	
467			
468			

MO1

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 13
DZDPEA.CMB TRAPCATCHER FOR UNEXPECTED INTERRUPTS

```

469 ;*****
470 ;-----
471 ; TRAPCATCHER FOR ILLEGAL INTERRUPTS
472 ; THE STANDARD "TRAP CATCHER" IS PLACED
473 ; BETWEEN ADDRESS 0 TO ADDRESS 776.
474 ; IT LOOKS LIKE "PC+2 HALT".
475 ;-----
476 ;*****
477
478      000000      .=0
479      ; STANDARD INTERRUPT VECTORS
480      ;-----
481
482      .=24
483      000024      004614      .PFAIL      ; POWER FAIL HANDLER
484      000026      000340      340          ; SERVICE AT LEVEL 7
485      000030      004232      .HLT          ; ERROR HANDLER
486      000032      000340      340          ; SERVICE AT LEVEL 7
487      000034      004200      .TRPSRV       ; GENERAL HANDLER DISPATCH SERVICE
488      000036      000340      340          ; SERVICE AT LEVEL 7
489
490      .=40
491      000040      000000      0            ; SAVE FOR ACT-11 OR DDP2
492      000042      000000      0            ; RETURN ADDRESS IF UNDER ACT-11 OR DDP2
493      000044      000000      0            ; SAVE FOR ACT-11 OR DDP2
494      000046      002766      $ENDAD        ; FOR USE WITH ACT-11 OR DDP2
495      000052      000000      0            ; ACT-11 PROGRAM CHARACTERISTICS
496
497      .=174
498      000174      000000      DISPRG:      0            ; SOFTWARE DISPLAY REGISTER
499      000176      000000      SWREG:      0            ; SOFTWARE SWITCH REGISTER
500      .=200
501      000200      000137      001562      JMP      .START      ; GO TO START OF PROGRAM
502
503
504      001000      001000      040515      047111      .=1000
505      (2) 001000      005377      040515      047111      MTITLE: .ASCIZ <377><12>/MAINDEC-11-DZDPE-A /<377>/DUP11 QUICK-VERIFY EXERCISER /<377>
506      001200      001200      .=1200
507      ; SWR AND LIGHTS
508      ;-----
509
510      001200      177570      DISPLAY:      177570      ; 11/45 CONSOLE LIGHTS
511      001202      177570      SWR:          177570      ; INDIRECT POINTER TO SWITCH REGISTER
512
513      ; INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
514      ;-----
515
516      001204      177560      TKCSR:      177560      ; TELETYPE KEYBOARD CONTROL REGISTER
517      001206      177562      TKDBR:      177562      ; TELETYPE KEYBOARD DATA BUFFER
518      001210      177564      TPCSR:      177564      ; TELEPRINTER CONTROL REGISTER
519      001212      177566      TPDBR:      177566      ; TELEPRINTER DATA BUFFER
520
521      ; PROGRAM CONTROL PARAMETERS
522      ;-----
523

```

NO1

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 14
 DZDPEA.CMB PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

524	001214	000000	RETURN:	0	; SCOPE ADDRESS FOR LOOP ON TEST
525	001216	000000	NEXT:	0	; ADDRESS OF NEXT TEST TO BE EXECUTED
526	001220	000000	LOCK:	0	; ADDRESS FOR LOCK ON CURRENT DATA
527	001222	000001	ICOUNT:	1	; NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
528	001224	000000	LPCNT:	0	; NUMBER OF ITERATIONS COMPLETED
529	001226	000000	TSTNO:	0	; NUMBER OF TEST IN PROGRESS
530	001230	000000	PASCNT:	0	; NUMBER OF PASSES COMPLETED
531	001232	000000	ERRCNT:	0	; TOTAL NUMBER OF ERRORS
532	001234	000000	LSTERR:	0	; PC OF LAST ERROR CALL
533					
534					
535			; PROGRAM CONTROL FLAGS		
536			; -----		
537					
538	001346	000	INIFLG:	.BYTE 0	; PROGRAM INITIALIZATION FLAG
539	001347	000	ERRFLG:	.BYTE 0	; ERROR OCCURED FLAG
540	001350	000	LOKFLG:	.BYTE 0	; LOCK ON CURRENT TEST FLAG
541	001351	000	QV.FLG:	.BYTE 0	; QUICK VERIFY FLAG.
542					; ON FIRST PASS OF EACH DUPL1 ITERATIONS
543					; WILL BE SUPPRESSED
544			.EVEN		

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 15
 DZDPEA.CMB PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

545
 546
 547
 548
 549
 550
 551
 552
 553
 554
 555
 556
 557
 558
 559
 560
 561
 562
 563
 564
 565
 566
 567
 568
 569
 570
 571
 572
 573
 574
 575
 576

;DEFINITIONS FOR TRAP SUBROUTINE CALLS
 ;POINTERS TO SUBROUTINES CAN BE FOUND
 ;IN THE TABLE IMMEDIATELY FOLLOWING THE DEFINITIONS

```

;*****
;-----
;TRPTAB:
SCOPE=TRAP+0      ;CALL TO SCOPE LOOP AND ITERATION HANDLER
      .SCOPE
SCOPI=TRAP+1      ;CALL TO LOOP ON CURRENT DATA HANDLER
      .SCOPI
TYPE=TRAP+2       ;CALL TO TELETYPE OUTPUT ROUTINE
      .TYPE
INSTR=TRAP+3      ;CALL TO ASCII STRING INPUT ROUTINE
      .INSTR
INSTER=TRAP+4     ;CALL TO INPUT ERROR HANDLER
      .INSTER
PARAM=TRAP+5      ;CALL TO NUMERICAL DATA INPUT ROUTINE
      .PARAM
SAVOS=TRAP+6      ;CALL TO REGISTER SAVE ROUTINE
      .SAVOS
RESOS=TRAP+7      ;CALL TO REGISTER RESTORE ROUTINE
      .RESOS
CONVRT=TRAP+10    ;CALL TO DATA OUTPUT ROUTINE
      .CONVRT
CNVRT=TRAP+11     ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
      .CNVRT
SETFLG=TRAP+12    ;CALL TO TELETYPE INPUT ROUTINE
      .SETFLG
;-----
;*****

```

DZLPE-A MACY11 27(732) 21-OCT-75 16:11 PAGE 16
 DZLPEA.CMB PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

```

577                                     ;DUP11 VECTOR AND REGISTER INDIRECT POINTERS
578
579 001400 000000 DUPRVC: 0 ; POINTER TO DUP11 RECEIVER INTERRUPT VECTOR
580 001402 000000 DUPRPS: 0 ; POINTER TO DUP11 RECEIVER INTERRUPT SERVICE PS
581 001404 000000 DUPTVC: 0 ; POINTER TO DUP11 TRANSMITTER INTERRUPT VECTOR
582 001406 000000 DUPTPS: 0 ; POINTER TO DUP11 TRANSMITTER INTERRUPT SERVICE PS
583 001410 000000 RXCSR: 0 ; POINTER TO DUP11 RECEIVER STATUS REGISTER
584 001412 000000 RXDBUF: 0 ; POINTER TO DUP11 RECEIVER DATA BUFFER
585 001414 000000 PARCSR: 0 ; POINTER TO DUP11 PARAMETER STATUS REGISTER
586 001416 000000 TXCSR: 0 ; POINTER TO DUP11 TRANSMITTER STATUS REGISTER
587 001420 000000 TXDBUF: 0 ; POINTER TO DUP11 TRANSMITTER DATA BUFFER
588 001422 000000 DUPSEC: 0 ; POINTER TO DUP11 SECONDARY REGISTER SELECT REGISTER
589 001424 000000 HUPPSR: 0 ; POINTER TO PARAMETER STATUS HIGH BYTE
590 001426 000000 HUPRBF: 0 ; POINTER TO RECEIVER BUFFER HIGH BYTE
591 001430 000000 HUPRCR: 0 ; POINTER TO RECEIVER CONTROL REG HIGH BYTE
592 001432 000000 HUPTBF: 0 ; POINTER TO TRANSMITTER BUFFER HIGH BYTE
593 001434 000000 HUPTCR: 0 ; POINTER TO TRANSMITTER CONTROL REG HIGH BYTE
594
595                                     ;DUP11 CONTROL INDICATORS FOR CURRENT DUP11 UNDER TEST
596                                     ;-----
597
598
599 001436 000 MASK.A: .BYTE 000 ; LAST CHAR TO TEST AND PARITY MASK
600
601 001437 010 CLK.A: .BYTE 8. ; NUMBER OF CLOCKS NEEDED FOR ONE CHAR
602
603 001440 000000 L00.00: 000000 ; PARAMETERS
604
605
606
607

```

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 17
 DZDPEA.CMB PROGRAM INITIALIZATION AND START UP.

```

608
609
610
611
612
613
614
615
        :PROGRAM INITIALIZATION
        :LOCK OUT INTERRUPTS
        :SET UP PROCESSOR STACK
        :SET UP POWER FAIL VECTOR
        :CLEAR PROGRAM CONTROL FLAGS AND COUNTS
        :TYPE TITLE MESSAGE
616 001562 012737 000340 177776 .START: MOV      #340,PS          ;LOCK OUT INTERRUPTS
617 001570 012706 001150          MOV      #STACK,SP        ;SET UP STACK
618 001574 012737 004614 000024 MOV      #.PFAIL,2#24    ;SET UP POWER FAIL VECTOR
619 001602 113737 001311 001313 MOV8     DUPNUM,SAVNUM    ;SAVE NUMBER OF DEVICES IN SYSTEM
620 001610 005037 001230          CLR      PASCNT          ;CLEAR PASS COUNT
621 001614 105037 001347          CLR8     ERRFLG          ;CLEAR ERROR FLAG
622 001620 105037 001351          CLR8     QV.FLG          ;ZERO QUICK VERIFY FLAG
623 001624 012737 001500 001316 MOV      #DUP.MAP,CREAM    ;GET MAP POINTER.
624 001632 112737 000001 001314 MOV8     #1,RUN          ;POINT POINTER TO FIRST DEVICE.
625 001640 005037 001232          CLR      ERRCNT          ;CLEAR ERROR COUNT
626 001644 005037 001234          CLR      LSTERR          ;CLEAR LAST ERROR POINTER
627 001650 012737 000001 001226 MOV      #1,TSTNO        ;SET UP FOR TEST 1
628 001656 012737 001562 001214 MOV      #.START,RETURN    ;SET UP FOR POWER FAIL BEFORE
629                                     ;TESTING STARTS
630 001664 013746 000006          MOV      2#6,-(SP)          ;SAVE CURRENT VECTORS
631 001670 013746 000004          MOV      2#4,-(SP)
632 001674 012737 001710 000004 MOV      #12,2#4          ;SETUP FOR TIMEOUT
633 001702 005777 177274          TST      2SWR              ;REFERENCE HARDWARE SWITCH REG
634 001706 000407          BR          13$                   ;BR IF IT EXISTS
635 001710 012737 000176 001202 12$: MOV      #SWREG,SWR      ;POINT TO SOFT SWR
636 001716 012737 000174 001200 MOV      #DISPREG,DISPLAY ;POINT TO SOFT DISPLAY REG
637 001724 022626          CMP      (SP)+,(SP)+              ;ADJUST STACK
638 001726 012637 000004          MOV      (SP)+,2#4          ;RESTORE VECTORS
639 001732 012637 000006          MOV      (SP)+,2#6
640 001736 105737 001346          TSTB     INIFLG              ;HAS INITIALIZATION BEEN PERFORMED
641 001742 001010          BNE         6$
642 001744 022737 002766 000042 11$: CMP      #SENDAD,2#42    ;IF ACT-11 AUTO MODE,
643 001752 001404          BEQ         6$                     ;DON'T TYPE ID
644 001754 104402 001000          TYPE     #TITLE              ;TYPE TITLE MESSAGE
645 001760 105137 001346          COMB     INIFLG              ;IF NOT SET FLAG AND DO
646 001764 105777 177212          TSTB     2SWR              ;BIT7=1??
647 001770 100002          BPL         10$
648 001772 000137 002402          JMP      1$
649 001776          10$:
650 001776 105137 001332          COMB     FLAG
651 001782 112737 000001 001346 MOV8     #1,INIFLG          ;SET TO MANUAL ENTRY
652 002010 012700 001500          MOV      #DUP.MAP,RO        ;CLR MAP
653 002014 000020          CLR      (RO)+
654 002016 000027 001560          CMP      RO,DUP.END        ;DONE WITH MAP?
655 002022 001374          BNE         68$                   ;BR IF NO
656 002024 105037 001340          CLR8     MONFLG
657 002030 100037 001342          CLR8     ALJMF
658 002034 104403          INSTR     ;OUTPUT MESSAGE & GET INPUT STRING
659 002036 005237          MCSR      ;MESSAGE
660 002040 104405          PARAM     ;CONVERT STRING
661 002042 160000          160000    ;LOW LIMIT
662 002044 175500          175500    ;HIGH LIMIT
663 002046 001500          DUCRO     ;STORE AT THIS LOCATION

```


E02

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 18
 DZDPEA.CMB PROGRAM INITIALIZATION AND START UP.

664	002050	001		.BYTE	1		:MASK
665	002051	001		.BYTE	1		:HOW MANY TIMES + 2
666	002052	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
667	002054	005256			MVEC		:MESSAGE
668	002056	104405			PARAM		:CONVERT STRING
669	002060	000300			300		:LOW LIMIT
670	002062	000770			770		:HIGH LIMIT
671	002064	001502			DUPTRO		:STORE AT THIS LOCATION
672	002066	001		.BYTE	1		:MASK
673	002067	001		.BYTE	1		:HOW MANY TIMES + 2
674	002070	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
675	002072	005606			MMODEM		:MESSAGE
676	002074	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
677	002076	001340			MDMFLG		:THIS FLAG
678	002100	105737	001340		TSTB		:MDMFLG
679	002104	001405			BEQ		:71\$
680	002106	105037	001322		CLRB		:TCNFLG
681	002112	105037	001342		CLRB		:ALJMF
682	002116	104441			BR		:70\$
683	002120	012737	000001	001236	71\$:	MOV	:#1,TEMP1
684	002126	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
685	002130	005344			MTCH		:MESSAGE
686	002132	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
687	002134	001322			TCNFLG		:THIS FLAG
688	002136	105737	001322		TSTB		:TCNFLG
689	002142	001427			BEQ		:70\$
690	002144	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
691	002146	005263			MALLJM		:MESSAGE
692	002150	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
693	002152	001342			ALJMF		:THIS FLAG
694	002154	105737	001342		TSTB		:ALJMF
695	002160	001020			BNE		:70\$
696	002162	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
697	002164	005271			MJMPR		:MESSAGE
698	002166	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
699	002170	001323			CPCLRJ		:THIS FLAG
700	002172	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
701	002174	005472			MSTJM		:MESSAGE
702	002176	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
703	002178	001334			STJMF		:THIS FLAG
704	002180	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
705	002182	005525			MSRJM		:MESSAGE
706	002184	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
707	002186	001336			SRJMF		:THIS FLAG
708	002188	104403			INSTR		:OUTPUT MESSAGE & GET INPUT STRING
709	002190	005737			MSC		:MESSAGE
710	002192	104412			SETFLG		:SET FLAG BASED UPON INPUT STRING
711	002194	001344			DSCFLG		:THIS FLAG
712	002196	112737	000001	001312	70\$:	MOVB	:#1,SAVACT
713	002198	113737	001236	001311		MOVB	:TEMP1,DUPNUM
714	002200	113737	001236	001313		MOVB	:TEMP1,SAVNUM
715	002202	005337	001236		65\$:	DEC	:TEMP1
716	002204	001404			BEQ		:64\$
717	002206	000261			SEC		
718	002208	106137	001312		ROLB		:SAVACT
719	002210	000771			BR		:65\$

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 19
 DZDPER.CMB PROGRAM INITIALIZATION AND START UP.

720	002262	113737	001312	001240	64\$:	MOV	SAVACT,TEMP2	:# OF TIMES
721	002270	113737	001312	001310		MOV	SAVACT,DUPACTV	
722	002276	000241				CLC		
723	002300	106037	001240			RORB	TEMP2	
724	002304	012700	0C1500			MOV	#DUPCRO,RO	
725	002310	012701	001506			MOV	#DUPCR1,R1	
726	002314	000241			67\$:	CLC		
727	002316	106037	001240			RORB	TEMP2	
728	002322	103010				BCC	66\$	
729	002324	012011				MOV	(R0)+,(R1)	
730	002326	062721	000010			ADD	#10,(R1)+	;CSR
731	002332	012011				MOV	(R0)+,(R1)	
732	002334	062721	000010			ADD	#10,(R1)+	;VECTOR
733	002340	012021				MOV	(R0)+,(R1)+	;PARAMETERS
734	002342	000764				BR	67\$	
735	002344				66\$:			
736	002344	104402	005560		16\$:	TYPE	XHEAD	;TYPE HEADER
737	002350	012737	001500	001236		MOV	#DUP.MAP,TEMP1	;SET POINTER
738	002356	017737	176654	001240	5\$:	MOV	#TEMP1,TEMP2	;SET DATA
739	002354	001406				BEQ	1\$	;ALL DONE WITH DATA
740	002356	104410				CONVRT		
741	002370	005004				XSTATQ		
742	002372	002737	000002	001236		ADD	#2,TEMP1	;UPDATE POINTER
743	002370	001766				BR	5\$	
744	002372	002777	000001	176572	1\$:	BIT	#SW00,#SWR	
745	002410	001405				BEQ	7\$	
746	002412	001737	001332			TST	FLAG	
747	002416	001002				BNE	7\$	
748	002420	001137	001776			JMP	10\$	
749	002424	001037	001332		7\$:	CLR	FLAG	
750	002430	005737	000042			TST	#42	;IS PROGRAM RUNNING UNDER MONITOR
751	002434	001030				BNE	3\$	;BR IF YES
752	002436	002777	000010	176536		BIT	#SW03,#SWR	;SELECT SPECIFIC DEVICES??
753	002444	001424				BEQ	3\$	;BR IF NO.
754	002446	104402	005157			TYPE	MNEW	;TYPE THE MESSAGE.
755	002452	001000				CLR	RO	;ZERO DATA LIGHTS
756	002454	001000				HALT		;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
757	002456	127737	176520	001312		CMPB	#SWR,SAVACT	;IS THE NUMBER VALID?
758	002464	101404				BLOS	2\$	;BR IF NUMBER IS OK.
759	002466	104402	005020			TYPE	,MERR3	;TELL USER OF INVALID NUMBER.
760	002472	001000				HALT		;STOP EVERY THING.
761	002474	001776				BR	-2	;RESTART THE PROGRAM AGAIN.
762	002476	117737	176500	001310	2\$:	MOV	#SWR,DUPACTV	;GET NEW DEVICE PATTERN
763	002504	113700	001310			MOV	DUPACTV,RO	;SHOW THE USER WHAT HE SELECTED.
764	002510	042700	177400			BIC	#1C<377>,RO	;USE ONLY LOW BYTE.
765	002514	000000				HALT		;CONTINUE DYNAMIC SWITCHES.
766	002516	012700	000300		3\$:	MOV	#300,RO	;PREPARE TO CLEAR THE FLOATING
767	002522	012701	000302			MOV	#302,R1	;VECTOR AREA. 300-776
768	002526	010120			4\$:	MOV	R1,(R0)+	;START PUTTING "PC+2 - HALT"
769	002530	001021				CLR	(R1)+	;IN VECTOR AREA.
770	002532	002021				CMP	(R0)+,(R1)+	;POP POINTERS
771	002534	002700	001000			CMP	#1000,RO	;ALL DONE??
772	002540	001372				BNE	4\$	;BR IF NO.
773								
774								
775								

;TEST START AND RESTART

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 20
 DZDPEA.CMB PROGRAM INITIALIZATION AND START UP.

776									
777	002542	012737	000340	177776	.BEGIN:	MOV	#340,PS		;LOCK OUT INTERRUPTS
778	002550	012706	001150			MOV	#STACK,SP		;SET UP STACK
779	002554	005737	000042			TST	#42		;IS PROGRAM UNDER MONITOR CONTROL
780	002560	001023				BNE	2\$		;BR IF YES
781	002562	032777	000004	176412		BIT	#BIT2,JSWR		;CHECK FOR LOCK ON TEST
782	002570	001411				BEQ	1\$		;BR IF NO LOCK DESIRED.
783	002572	104402	005056			TYPE	,MLOCK		;TYPE LOCK SELECTED.
784	002576	012737	000240	003056		MOV	#NOP,TTST		;ADJUST SCOPE ROUTINE.
785	002604	012737	000240	003060		MOV	#NOP,TTST+2		;SET UP TO LOCK
786	002612	000406				BR	2\$		;CONTINUE ALONG.
787	002614	013737	003170	003056	1\$:	MOV	BRW,TTST		;PREPARE NORMAL SCOPE ROUTINE
788	002622	013737	003172	003060		MOV	BRX,TTST+2		;LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
789	002630	012737	006166	001214	2\$:	MOV	#CYCLE,RETURN		;START AT "CYCLE" FIND WHICH DEVICE TO TEST
790	002636	104402	004746			TYPE	,MR		;TYPE R
791	002642	003177	176346			JMP	RETURN		;START TESTING

```

792                                     ;END OF PASS
793                                     ;TYPE NAME OF TEST
794                                     ;UPDATE PASS COUNT
795                                     ;CHECK FOR EXIT TO ACT-11
796                                     ;RESTART TEST
797
798 002646 005037 001234      .EOP: CLR      LSTERR      ;CLEAR LAST ERROR PC
799 001652 105037 001347      CLR      ERRFLG      ;CLEAR ERROR FLAG
800 000056 005237 001230      INC      PASCNT      ;UPDATE PASS COUNT
801 000002 013777 001230      MOV      PASCNT, @DISPLAY ;DISPLAY PASS COUNT
802 000070 104402 004723      TYPE     ,MEPASS      ;TYPE END PASS
803 000074 104402 005105      TYPE     ,MCSR      ;TYPE CSR
804 000070 104411 003012      CNVRT    ,XCSR      ;SHOW IT
805 000070 104402 005113      TYPE     ,MVECX      ;TYPE VECTOR
806 000071 104411 003020      CNVRT    ,XVEC      ;SHOW IT
807 000071 104402 005121      TYPE     ,MPASSX      ;TYPE PASSES
808 000072 104411 003026      CNVRT    ,XPASS      ;SHOW IT
809 000072 104402 005132      TYPE     ,MERRX      ;TYPE ERRORS
810 000073 104411 003034      CNVRT    ,XERR      ;SHOW IT
811 000073 105337 001313      DECB     SAVNUM      ;ARE ALL DEVICES TESTED?
812 000074 001017 000377 001351 BNE     RESTR      ;BR IF NO.
813 000074 112737 000377 001351 MOV     #377, @.FLG ;SET THE QUICK VERIFY FLAG.
814 000075 113737 001311 001313 MOV     DUPNUM, SAVNUM ;RESTORE THE COUNT
815 000075 013701 000042      MOV     @#42, R1 ;CHECK FOR ACT-11 OR DDP
816 000076 001406 000005      BEQ     RESTR      ;IF NOT, CONTINUE TESTING
817 000076 000005 000005      RESET     ;STOP THE SHOW--CLEAR THE WORLD
818
819 000076 004711      SENDAO: JSR      PC, (R1)
820 000077 000240      NOP
821 000077 000240      NOP
822 000077 000240      NOP
823 000077 000240      NOP
824 000077 000240      NOP
825 003000 012737 006166 001214 RESTR: MOV     #CYCLE, RETURN
826 003006 000137 006166      JMP      CYCLE
827 003014 000006 000002      XCSR: 1 ;.BYTE 6,2
828 003016 001410      RXCSR
829 003020 000001      XVEC: 1 ;.BYTE 3,2
830 003022 000003 000002      DUPRVC
831 003024 001400      XPASS: 1 ;.BYTE 6,2
832 003026 000001 000002      PASCNT
833 000030 000006 000002      XERR: 1 ;.BYTE 6,2
834 000032 001230      ERRCNT
835 000034 000001 000002      .BYTE 6,2
836 000036 000006 000002      ERRCNT
837 000040 001232
838
839                                     ;SCOPE LOOP AND INTERATION HANDLER
840
841 003042 005037 001234      .SCOPE: CLR      LSTERR      ;CLEAR LAST ERROR PC
842 003046 010016 000000 176124 MOV     R0, (SP) ;SAVE R0 ON STACK
843 003050 032777 040000      BIT     #BIT14, @SWR ;LOOP ON TEST?
844 003056 001407 000000      BEQ     1$ ;BR IF NO (IF LOCK SW01 = 1; THIS LOCATION = 240)
845 003060 000437 000000      BR      3$ ;GO TO 3$ (DITTO)
846 003062 105777 176116      TSTB    @TKCSR ;KYBD DONE?
847 003066 100034 000000      BPL     3$ ;BR IF NO (LOCK: HIT A KEY ON TTY TO GO TO NEXT TEST)

```

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 22
 DZDPEA.CMB END OF PASS ROUTINE

```

848 003070 017700 176112      MOV      JTKDBR,R0      ;CLR DONE BIT
849 003074 000415      BR        2$      ;CONTINUE
850 003076 032777 004000 176076 1$: BIT      #SW11,2SWR      ;DELETE ITERATION (QUICK PASS)?
851 003104 001011      BNE      2$      ;BR IF YES
852 003106 105737 001351      TSTB     QV.FLG      ;HAS FIRST PASS BEEN COMPLETED?
853 003112 001406      BEQ       2$      ;BR IF QUICK VERIFY
854 003114 005237 001224      INC      LPCNT      ;UPDATE ITERATION COUNTER
855 003120 023737 001224 001222      CMP     LPCNT,ICOUNT ;ALL ITERATIONS DONE?
856 003126 001014      BNE      3$      ;BR IF NOT YET
857 003130 105037 001347      CLRB     ERRFLG      ;PREPARE FOR NEW TEST
858 003134 005037 001224      CLR      LPCNT      ;START ICOUNT AT ZERO
859 003140 005037 001220      CLR      LOCK
860 003144 012737 000050 001222      MOV     #50,ICOUNT ;RESET ITERATIONS
861 003152 013737 001216 001214      MOV     NEXT,RETURN ;GET NEXT TEST
862 003160 011600      MOV     (SP),R0 ;POP R0 OFF STACK
863 003162 022626      POP2SP ;FAKE AN RTI
864 003164 000177 176024      JMP      JRETURN ;GO DO THE TEST
865 003170 001407      BRW:    1407
866 003172 000437      BRX:    437
867                                     ;TELETYPE OUTPUT ROUTINE
868                                     -----
869
870 003220 010546      .TYPE:  MOV     R5,-(SP) ;SAVE R5 ON THE STACK.
871 003222 017605 000002      MOV     @2(SP),R5 ;GET ADDRESS OF MESSAGE.
872 003226 062766 000002 000002      ADD     #2,2(SP) ;POP OVER ADDRESS.
873 003234 032777 010000 175740 1$: BIT     #SW12,2SWR ;INHIBIT ALL PRINT OUT??
874 003242 001012      BNE      3$      ;BR IF NO PRINT OUT WANTED (SW12=1)
875 003244 105715      TSTB     (R5) ;IS NUMBER MINUS? (MSB=1(BIT7))
876 003246 100002      BPL      2$      ;BR IF NUMBER IS PLUS
877 003250 104402 004702      TYPE     MCRLF ;TYPE A CR/LF!
878 003254 105777 175730 2$: TSTB     JTPCSR ;TTY READY?
879 003260 100375      BPL      2$      ;BR IF NO.
880 003262 112577 175724      MOVB     (R5)+,JTPDBR ;PRINT CURRENT CHAR.
881 003266 001362      BNE      1$      ;IF NOT ZERO KEEP PRINTING!
882 003270 012605 3$: MOV      (SP)+,R5 ;END OF OUTPUT. RESTORE R5
883 003272 000002      RTI ;GO HOME
884                                     -----
885
886 003274 010346      .INSTR: MOV     R3,-(SP) ;SAVE R3 ON STACK
887 003276 010446      MOV     R4,-(SP) ;SAVE R4 ON STACK
888 003300 017637 000004 003316      MOV     @4(SP),.MSG
889 003306 062766 000002 000004      ADD     #2,4(SP)
890 003314 104402      .INST1: TYPE
891 003316 000000      .MSG:    0
892 003320 012704 006122      MOV     #INBUF,R4
893 003324 012703 000007      MOV     #7,R3
894 003330 105777 175650 1$: TSTB     JTKCSR
895 003334 100375      BPL      1$
896 003336 117714 175644      MOVB     JTKDBR,(R4)
897 003342 142714 000200      BICB     #200,(R4)
898 003346 122427 000015      CMPB     (R4),#15
899 003352 001417      BEQ      INSTR2
900 003354 105777 175630 2$: TSTB     JTPCSR
901 003360 100375      BPL      2$
902 003362 017777 175620 175622      MOV     JTKDBR,JTPDBR
903 003370 005303      DEC      R3

```

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 23
 DZDPER.CMB END OF PASS ROUTINE

904	003372	001356		BNE	1\$	
905	003374	012604		MOV	(SP)+,R4	
906	003376	012603		MOV	(SP)+,R3	
907	003400	010346		.INSTE: MOV	R3,-(SP)	
908	003402	010446		MOV	R4,-(SP)	
909	003404	004402	004676	TYPE	,MM	
910	003410	000741		BR	.INST1	
911	003412	012604		INSTR2: MOV	(SP)+,R4	;RESTORE R4
912	003414	012603		MOV	(SP)+,R3	;RESTORE R3
913	003416	000002		RTI		
914						
915					;CONVERT ASCII STRING TO OCTAL	
916					-----	
917						
918	003420	010546		.PARAM: MOV	R5,-(SP)	
919	003422	010446		MOV	R4,-(SP)	
920	003424	016605	000004	MOV	4(SP),R5	
921	003430	012537	003610	MOV	(R5)+,LOLIM	
922	003434	012537	003612	MOV	(R5)+,HILIM	
923	003440	012537	003614	MOV	(R5)+,DEVADR	
924	003444	112537	003616	MOV8	(R5)+,LOBITS	
925	003450	112537	003617	MOV8	(R5)+,ADRCNT	
926	003454	010566	000004	MOV	R5,4(SP)	
927	003460	005005		PARAM1: CLR	R5	
928	003462	012704	006122	MOV	#INBUF,R4	
929	003466	122714	000015	CMPS	#15,(R4)	
930	003472	001420		BEQ	PARERR	
931	003474	121427	000060	1\$: CMPS	(R4),#60	
932	003500	002415		BLT	PARERR	
933	003502	121427	000067	CMPS	(R4),#67	
934	003506	003012		BGT	PARERR	
935	003510	142714	000060	BICB	#60,(R4)	
936	003514	152405		BISB	(R4)+,R5	
937	003516	122714	000015	CMPS	#15,(R4)	
938	003522	001406		BEQ	LIMITS	
939	003524	006305		ASL	R5	
940	003526	006305		ASL	R5	
941	003530	006305		ASL	R5	
942	003532	000760		BR	1\$	
943	003534	104404		PARERR: INSTER		
944	003536	000750		BR	PARAM1	
945						
946					;TEST TO SEE IF NUMBER IS WITHIN LIMITS	
947					-----	
948						
949	003540	020537	003612	LIMITS: CMP	R5,HILIM	
950	003544	101373		BHI	PARERR	
951	003546	020537	003610	CMP	R5,LOLIM	
952	003552	103770		BLO	PARERR	
953	003554	133705	003616	BITB	LOBITS,R5	
954	003560	001365		BNE	PARERR	
955						
956					;STORE NUMBER AT SPECIFIED ADDRESS	
957						
958	003562	013704	003614	1\$: MOV	DEVADR,R4	
959	003566	010524		MOV	R5,(R4)+	

K02

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 24
 DZDPEA.CMB END OF PASS ROUTINE

960	003570	062705	000002		ADD	#2,R5
961	003574	105337	003617		DECB	ADRCNT
962	003600	001372			BNE	1\$
963	003602	012604			MOV	(SP)+,R4
964	003604	012605			MOV	(SP)+,R5
965	003606	000002			RTI	
966	003610	000000				
967	003612	000000				
968	003614	000000				
969	003616	000000				
970		003617				
971						
972						
973						
974	003712	104402	004702			
975	003716	010046				
976	003720	010146				
977	003722	010346				
978	003724	010446				
979	003726	010546				
980	003730	017601	000012			
981	003734	062766	000002	000012		
982	003742	012137	004116			
983	003746	112137	004120			
984	003752	112137	004121			
985	003756	013137	004122			
986	003762	013704	004122			
987	003766	113705	004120			
988	003772	012700	006016			
989	003776	010403				
990	004000	042703	177770			
991	004004	062703	000060			
992	004010	110320				
993	004012	000241				
994	004014	000004				
995	004016	000241				
996	004020	006004				
997	004022	000241				
998	004024	000004				
999	004026	000305				
1000	004030	001362				
1001	004032	012703	006060			
1002	004036	114023				
1003	004040	105337	004120			
1004	004044	001374				
1005	004046	105737	004121			
1006	004052	001405				
1007	004054	112723	000040			
1008	004060	105337	004121			
1009	004064	001373				
1010	004066	105013				
1011	004070	104402	006060			
1012	004074	005337	004116			
1013	004100	001322				
1014	004102	012605				
1015	004104	012604				

LOLIM: 0
 HILIM: 0
 DEVADR: 0
 LOBITS: 0
 ADRCNT=LOBITS+1
 ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER

 .CONVR: TYPE MCRLF
 .CNVRT: MOV R0,-(SP)
 MOV R1,-(SP)
 MOV R3,-(SP)
 MOV R4,-(SP)
 MOV R5,-(SP)
 MOV 212(SP),R1
 ADD #2,12(SP)
 MOV (R1)+,WRDCNT
 1\$: MOV (R1)+,CHRCNT
 MOV (R1)+,SPACNT
 MOV 2(R1)+,BINWRD
 2\$: MOV BINWRD,R4
 MOV CHRCNT,R5
 MOV #TEMP,R0
 3\$: MOV R4,R3
 BIC #177770,R3
 ADD #060,R3
 MOV R3,(R0)+
 CLC
 ROR R4
 CLC
 ROR R4
 CLC
 ROR R4
 DEC R5
 BNE 3\$
 MOV #MDATA,R3
 4\$: MOV -(R0),R3
 DECB CHRCNT
 BNE 4\$
 TSTB SPACNT
 BEQ 6\$
 5\$: MOV #040,(R3)+
 DECB SPACNT
 BNE 5\$
 6\$: CLRB (R3)
 TYPE ,MDATA
 DEC WRDCNT
 BNE 1\$
 MOV (SP)+,R5
 MOV (SP)+,R4

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 25
 DZDPEA.CMB END OF PASS ROUTINE

```

1016 004106 012603      MOV      (SP)+,R3
1017 004110 012601      MOV      (SP)+,R1
1018 004112 012600      MOV      (SP)+,R0
1019 004114 000002      RTI
1020 004116 000000      WRDCNT: 0
1021 004120 000000      CHRCNT: 0
1022      004121 004121      SPACNT=CHRCNT+1
1023 004122 000000      BINWRD: 0
1024
1025
1026      ;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
1027      ;BUFFER TO THE CHARACTERS "N" AND "Y".
1028      ;IF THE CHARACTER IS "N" CLEAR THE FLAG
1029      ;IF THE CHARACTER IS "Y" SET THE FLAG
1030
1031 004124 017605 000000 .SETFLG:MOV      2(SP),R5
1032 004130 042737 000040 006122      BIC      #40,INBUF
1033 004136 122737 000116 006122      CMPB     #'N',INBUF      ;IS IT "N" ?
1034 004144 001002      BNE      1$
1035 004146 105015      CLRB     (R5)      ;000
1036 004150 000406      BR       2$
1037 004152 122737 000131 006122 1$:      CMPB     #'Y',INBUF      ;IS IT "Y" ?
1038 004160 001005      BNE      3$
1039 004162 112715 177777      MOVB     #-1,(R5)      ;377
1040 004166 062716 000002      2$:      ADD      #2,(SP)
1041 004172 000002      RTI
1042 004174 104404      3$:      INSTER   ;RETRY
1043 004176 000752      BR       .SETFLG
1044
1045
1046      ;TRAP DISPATCH SERVICE
1047      ;ARGUMENT OF TRAP IS EXTRACTED
1048      ;AND USED AS OFFSET TO OBTAIN POINTER
1049      ;TO SELECTED SUBROUTINE
1050
1051 004200 011646      .TRPSR:MOV      (SP),-(SP)      ;GET PC OF RETURN
1052 004202 162716 000002      SUB      #2,(SP)      ;=PC OF TRAP
1053 004206 017616 000000      MOV      2(SP),(SP)      ;GET TRP
1054 004212 006316      TRPOK:ASL      (SP)      ;MULTIPLY TRAP ARG BY 2
1055 004214 042716 177001      BIC      #177001,(SP)      ;CLEAR UNWANTED BITS
1056 004220 062716 001352      ADD      #.TRPTAB,(SP)      ;POINTER TO SUBROUTINE ADDRESS
1057 004224 017616 000000      MOV      2(SP),(SP)      ;SUBROUTINE ADDRESS
1058 004230 000136      JMP      2(SP)      ;GO TO SUBROUTINE
1059
1060      ;ERROR HANDLER
1061      ;-----
1062
1063 004232 032777 010000 174742 .HLT:      BIT      #SW12,2SWR      ;BELL ON ERROR?
1064 004240 001406      BEQ      XBX      ;BR IF NO BELL
1065 004242 105777 174742      TSTB     2TPCSR      ;TTY READY.
1066 004246 100003      BPL      XBX      ;DON'T WAIT IF TTY NOT READY.
1067 004250 112777 000207 174734      MOVB     #207,2TPDBR      ;PUSH A BELL AT THE TTY.
1068 004256 032777 020000 174716 XBX:      BIT      #SW13,2SWR      ;DELETE ERROR PRINT OUT?
1069 004264 001105      BNE      HALTS      ;BR IF NO PRINT OUT WANTED.
1070 004266 021637 001234      CMP      (SP),LSTERR      ;WAS THIS ERROR FOUND LAST TIME?
1071 004272 001404      BEQ      1$      ;BR IF YES

```

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 26
 DZDPEA.CMB END OF PASS ROUTINE

1072	004274	011637	001234		MOV	(SP), LSTERR	: RECORD BEING HERE
1073	004300	105037	001347		CLRB	ERRFLG	: PREPARE HEADER
1074	004304	104406		15:	SAV05		: SAVE ALL PROC REGISTERS
1075	004306	011605			MOV	(SP), R5	: GET THE PC OF ERROR
1076	004310	162705	000002		SUB	#2, R5	: GET ADDRESS OF TRAP CALL
1077	004314	011504			MOV	(R5), R4	: GET HLT INSTRUCTION
1078	004316	006304			ASL	R4	: MULT BY TWO
1079	004320	061504			ADD	(R5), R4	: DOUBLE IT
1080	004322	006304			ASL	R4	: MULT AGAIN
1081	004324	042704	177001		BIC	#177001, R4	: CLEAR JUNK
1082	004330	062704	011674		ADD	#.ERRTAB, R4	: GET POINTER
1083	004334	012437	004450		MOV	(R4)+, ERRMSG	: GET ERROR MESSAGE
1084	004340	012437	004462		MOV	(R4)+, DATAHD	: GET DATA HEADRER
1085	004344	011437	004474		MOV	(R4), DATABP	: GET DATA TABLE
1086	004350	105737	001347		TSTB	ERRFLG	: TYPE HEADREER
1087	004354	001403			BEQ	TYPMSG	: BR IF YES
1088	004356	005737	004474		TST	DATABP	: DOES DATA TABLE EXIST?
1089	004362	001040			BNE	TYPDAT	: BR IF YES.
1090	004364	104402	004702		TYPMSG:	TYPE	
1091	004370	104402	004702			, MCRLF	
1092	004374	005737	001220			, MCRLF	
1093	004400	001402			TST	LOCK	
1094	004402	104402	005155		BEQ	15	
1095	004406	104402	005143		TYPE	, MASTEK	
1096	004412	104411	004602		TYPE	, MTSTN	
1097	004416	104402	005232		CNVRT	, XTSTN	: SHOW IT
1098	004422	104411	004574		TYPE	, MERRPC	: TYPE PC.
1099	004426	104402	004702		CNVRT	, ERTABO	: SHOW IT
1100	004432	112737	177777	001347	TYPE	, MCRLF	: GIVE A CR/LF
1101	004440	005737	004450		MOVB	#-1, ERRFLG	: NO MORE HEADER UNLESS NO DATA TABLE.
1102	004444	001402			TST	ERRMSG	: IS THERE AN ERROR MESSAGE?
1103	004446	104402			BEQ	WRKO.FM	: BR IF NO.
1104	004450	000000			TYPE		: TYPE
1105	004452				ERRMSG:	0	: ERROR MESSAGE
1106	004452	005737	004462		WRKO.FM:		
1107	004456	001402			TST	DATAHD	: DATA HEADER?
1108	004460	104402			BEQ	TYPDAT	: BR IF NO
1109	004462	000000			TYPE		: TYPE
1110	004464	005737	004474		DATAHD:	0	: DATA HEADER
1111	004470	001402			TYPDAT:	TST	: DATA TABLE?
1112	004472	104410				DATABP	: BR IF NO.
1113	004474	000000			BEQ	RESREG	: SHOW
1114	004476	104407			CONVRT		: DATA TABLE
1115	004500	022737	002766	000042	RESREG:	RES05	: RESTORE PROC REGISTERS
1116	004506	001403			HALTS:	CMP	: IF ACT-11 AUTO MODE--HALT!!
1117	004510	005777	174466			#SENDAD, #42	
1118	004514	160005			BEQ	15	
1119	004516	010046			TST	2SWR	: HALT ON ERROR?
1120	004520	016600	000002		BPL	EXITER	: BR IF NO HALT ON ERROR
1121	004524	000000			15:	PUSHRO	: SAVE R0
1122	004526	012600			MOV	2(SP), R0	: SHOW ERROR PC IN DATA LIGHTS
1123	004530	005237	001232		HALT		: HALT
1124	004534	032777	000400	174440	POPPO		: GET R0
1125	004542	001007			EXITER:	INC	: UPDATE ERROR COUNT
1126	004544	032777	002000	174430		ERRCNT	: GOTO TOP OF TEST?
1127	004552	001407			BIT	#SW08, 2SWR	: BR IF YES
					BNE	15	: GOTO NEXT TEST?
					BIT	#SW10, 2SWR	: BR IF NO
					BEQ	25	

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 27
DZDPEA.CMB END OF PASS ROUTINE

ADDRESS	HEX	ASCII	DISPATCH	OPERATION	OPERANDS	REMARKS
1128	004554	013737	001216	MOV	NEXT_RETURN	;SET FOR NEXT TEST
1129	004552	012706	001150	MOV	*STACK,SP	;RESET SP
1130	004566	000177	174422	JMP	RETURN	;GOTO SPECIFIED TEST
1131	004572	000002		RTI		;RETURN
1132	004574	000001		ERTABO: 1		
1133	004576	006	002	.BYTE	6,2	
1134	004600	001266		SAVPC		
1135	004602	000001		XTSTN: 1		
1136	004604	003	002	.BYTE	3,2	
1137	004606	001226		TSTNO		
1138						

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 23
 DZDPEA.CMB END OF PASS ROUTINE

```

1139      ;WAIT ROUTINE
1140      004610 000240      SMALL: NOP      ;STALL
1141      004612 000207      RTS      PC      ;RETURN
1142
1143      ;POWER FAIL ROUTINE
1144
1145      .PFAIL: MOV      #PWRUP,24      ;LOAD PFAIL VECTOR FOR POWER UP
1146      004614 012737 004624 000024      HALT
1147      004622 000000      PWRUP: RESET      ;WAIT TTY TO COME UP
1148      004624 000005      MOV      #STACK,SP      ;REINIT STACK POINTER
1149      004626 012706 001150      MOV      #.PFAIL,24      ;LOAD PFAIL VECTOR FOR POWER DOWN
1150      004632 012737 004614 000024      TYPE
1151      004640 104402      MPOWER
1152      004642 004705      JMP      @RETURN
1153      004644 000177 174344
1154
1155      ;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2,HALT
1156
1157      CLRVEC: MOV      #300,R2      ;R2 COMM VECTOR AREA ADRS
1158      004650 012702 000300      MOV      #302,R1      ;INIT R1 WITH ADRS OF HALT
1159      004654 012701 000302      IS: MOV      R1,(R2)+      ;MOV .+2 TO PC
1160      004660 010122      CLR      (R2)+      ;MOV HALT TO PC
1161      004662 005022      CMP      (R1)+,(R1)+      ;INC TO NEXT VECTOR AREA
1162      004664 022121      CMP      #776,R1      ;END OF VECTOR AREA
1163      004666 022701 000776      BNE      IS      ;NO
1164      004672 001372      RTS      PC      ;RETURN
1165      004674 000207
1166
1167
1168      XSTAT0: 2
1169      006004 000002      .BYTE      6,3
1170      006006 006      TEMP1
1171      006010 001236      .BYTE      6,2
1172      006012 006      TEMP2
1173      006014 001240
1174
1175      .EVEN
1176      TEMP: 0
1177      .+.40
1178      MDATA: 0
1179      .+.40
1180      INBUF: 0
1181      .+.40
1182      TRP.PC: .BLKW 1

```

```

1183
1184
1185
1186
1187
1188
1189
1190
1191
1192 006166 105737 001310      CYCLE: TSTB   DUPACTV   ;ARE ANY DUP11'S TO BE TESTED?
1193 006172 001004              BNE     1$      ;BR IF OK.
1194 006174 104402 004751      TYPE    ,MERR2    ;NO DUP11'S SELECTED!!
1195 006200 000000              HALT                    ;STOP THE SHOW.
1196 006202 000776              BR      -2          ;DISQUALIFY CONT. SW.
1197 006204 133737 001314 001310 1$: BITB   RUN,DUPACTV ;IS THIS ONE "ACTIVE"
1198 006212 001020              BNE     2$      ;BR IF GOOD ONE FOUND.
1199 006214 000241              CLC                    ;CLEAR PROC. CARRY BIT.
1200 006216 106137 001314      ROLB   RUN          ;UPDATE POINTER
1201 006222 105537 001314      ADCB   RUN          ;CATCH CARRY FROM RUN
1202 006226 062737 000006 001316      ADD   #6,CREAM ;UPDATE ADDRESS POINTER.
1203 006234 022737 001560 001316      CMP   #DUP.END,CREAM
1204 006242 001360              BNE     1$      ;KEEP GOING; NOT ALL TESTED FOR.
1205 006244 012737 001500 001316      MOV   #DUP.MAP,CREAM ;RESET ADDRESS POINTER.
1206 006252 000754              BR      1$      ;KEEP LOOKING FOR ACTIVE DUP11
1207 006254 000241              CLC                    ;CLEAR PROC. CARRY.
1208 006256 106137 001314      ROLB   RUN          ;UPDATE POINTER.
1209 006262 105537 001314      ADCB   RUN          ;CATCH CARRY.
1210 006266 013700 001316      MOV   CREAM,RO      ;GET ADDRESS POINTER.
1211 006272 062737 000006 001316      ADD   #6,CREAM ;UPDATE.
1212 006300 022737 001560 001316      CMP   #DUP.END,CREAM
1213
1214 006306 001003              BNE     3$      ;ALL DONE?
1215 006310 012737 001500 001316      MOV   #DUP.MAP,CREAM ;BR IF NO.
1216 006316 012037 001410              MOV   (RO)+,RXCSR ;RESTORE POINTER.
1217 006322 012037 001400              MOV   (RO)+,DUPRVC ;LOAD SYSTEM CTRL. REG
1218 006326 012037 001440              MOV   (RO)+,LOO.00 ;LOAD VECTOR
1219 006332 012700 000002              MOV   #2,RO      ;GET PARAMETERS
1220 006336 013737 001410 001430      MOV   RXCSR,HUPRCR ;SAVE CORE THIS WAY!
1221 006344 005237 001430              INC    HUPRCR   ;GET CONTROL REG HIGH BYTE
1222 006350 013737 001430 001412      MOV   HUPRCR,RXDBUF ;GOT IT
1223 006356 005237 001412              INC    RXDBUF   ;GET RX CONTROL REG BUFFER
1224 006362 013737 001412 001422      MOV   RXD.F,DUPSEC ;GOT IT
1225 006370 013737 001412 001414      MOV   RXD.F,PARCSR ;GOT SECONDARY REG SELECT REG
1226 006376 013737 001412 001426      MOV   RXD.F,HUPRBF ;GOT PARAMETER STATUS REGISTER
1227 006404 005237 001426              INC    HUPRBF   ;GET RX BUFFER HIGH BYTE
1228 006410 013737 001426 001424      MOV   HUPRBF,HUPPSR ;GOT IT
1229 006416 013737 001424 001416      MOV   HUPPSR,TXCSR  ;GOT PAR STATUS REG HIGH BYTE
1230 006424 005237 001416              INC    TXCSR    ;GET TX CONTROL REGISTER
1231 006430 013737 001416 001434      MOV   TXCSR,HUPTCR ;GOT IT
1232 006436 005237 001434              INC    HUPTCR   ;GET TX CONTROL REG HIGH BYTE
1233 006442 013737 001434 001420      MOV   HUPTCR,TXDBUF ;GOT IT
1234 006450 005237 001420              INC    TXDBUF   ;GET TX BUFFER
1235 006454 013737 001420 001432      MOV   TXDBUF,HUPTBF ;GOT IT
1236 006462 005237 001432              INC    HUPTBF   ;GET TX BUFFER HIGH BYTE
1237
1238 006466 013737 001400 001402      MOV   DUPRVC,DUPRPS ;RX VECTOR

```


1239	006474	060037	001402		ADD	RO,DUPRPS	;RX PRIORITY LEVEL
1240	006500	013737	001402	001404	MOV	DUPRPS,DUPTVC	
1241	006506	060037	001404		ADD	RO,DUPTVC	;TX VECTOR
1242	006512	013737	001404	001406	MOV	DUPTVC,DUPTPS	
1243	006520	060037	001406		ADD	RO,DUPTPS	;TX PRIORITY LEVEL
1244							
1245							
1246	006524	012700	001440		MOV	#L00.00,RO	;LOAD STATUS 00-00
1247	006530	012701	001436		MOV	#MASK.A,R1	;PREPARE MASK.
1248	006534	012702	001437		MOV	#CLK.A,R2	;PREPARE CLOCKS
1249	006540	004737	006704		JSR	PC,FIX.00	;GO AND CALCULATE CONFIGURATION.
1250	006544	005737	000042		TST	#42	
1251	006550	001050			BNE	4\$	
1252	006552	032777	000002	172422	BIT	#SW01,#SWR	;IF SW01=1,GET STARTING TEST #
1253	006560	001444			BEQ	4\$	
1254	006562	104402	004702	7\$:	TYPE	,MCRLF	
1255	006566	104403			INSTR	,OUTPUT MESSAGE & GET INPUT STRING	
1256	006570	005143			MTSTN	,MESSAGE	
1257	006572	104405			PARAM	,CONVERT STRING	
1258	006574	000001			I	,LOW LIMIT	
1259	006576	001000			1000	,HIGH LIMIT	
1260	006580	001226			TSTNO	,STORE AT THIS LOCATION	
1261	006582	000		.BYTE	0	,MASK	
1262	006603	001		.BYTE	1	,HOW MANY TIMES + 2	
1263	006604	012700	007034		MOV	#TST1,RO	
1264	006610	022710	012737	5\$:	CMP	#12737,(RO)	
1265	006614	001017			BNE	6\$	
1266	006616	023760	001226	000002	CMP	TSTNO,2(RO)	
1267	006624	001013			BNE	6\$	
1268	006626	022760	001226	000004	CMP	#TSTNO,4(RO)	
1269	006634	001007			BNE	6\$	
1270	006636	010037	001214		MOV	RO,RETURN	;SAVE PC
1271	006642	104402	004702		TYPE	,MCRLF	
1272	006646	104402	004746		TYPE	,MR	
1273	006652	000412			BR	8\$	
1274	006654	005720		6\$:	TST	(RO)+	
1275	006656	000027	010370		CMP	RO,#TLAST+10	
1276	006662	000122			BNE	5\$	
1277	006664	104402	004676		TYPE	,MOM	
1278	006670	000734			BR	7\$	
1279							
1280	006672	012737	007034	001214	4\$: MOV	#TST1,RETURN	;PREPARE RETURN ADDRESS
1281	006700	000177	172310		8\$: JMP	#RETURN	;GO START TESTING.
1282							
1283	006704	011003			FIX.00: MOV	(RO),R3	;GET PARAMETERS.
1284	006706	000207			5\$: RTS	PC	


```

1285                                     : THIS ROUTINE PICKS UP THE ADDRESS OF
1286                                     : THE JUMPER TABLE AND LOADS R5 WITH
1287                                     : THE CORRECT DATA BASED ON THE STATE
1288                                     : OF THE JUMPER AND CONNECTOR FLAGS.
1289                                     :-----
1290
1291 006710 012100                                     JUMPER: MOV (R1)+,R0      ;GET THE TABLE ADDRESS
1292 006712 105737 001322                             TSTB TCNFLG      ;TEST THE TURN AROUND CONNECTOR FLAG
1293 006716 001406                                     BEQ 2$              ;BRANCH IF CONNECTOR IS MISSING
1294 006720 105737 001323                             TSTB OPCLRJ      ;TEST CLEAR JUMPER FLAG
1295 006724 001403                                     BEQ 2$              ;BRANCH IF JUMPER IS MISSING
1296 006726 011005                                     MOV (R0),R5      ;MOVE THE DATA TO R5, BOTH JUMPER
1297                                           ;AND CONNECTOR ARE THERE
1298 006730 000137 006754                                     JMP 5$
1299 006734 000202 2$: CMP (R0)+,(R0)+      ;POP POINTER
1300 006736 105737 001323                             TSTB OPCLRJ      ;TEST CLEAR JUMPER FLAG
1301 006742 001403                                     BEQ 3$              ;BRANCH IF MISSING
1302 006744 011005                                     MOV (R0),R5      ;MOVE DATA- JUMPER IN, CONNECTOR OFF
1303 006746 000137 006754                                     JMP 5$
1304 006752 012005 3$: MOV (R0)+,R5      ;NO CONNECTOR OR JUMPER
1305 006754 000201 5$: RTS R1              ;RETURN
1306
1307 006756 012100 0JUMPER: MOV (R1)+,R0      ;GET THE POINTER ADDRESS
1308 006760 105737 001322                             TSTB TCNFLG      ;CHECK FOR TURNAROUND CONNECTOR
1309 006764 001403                                     BEQ 4$              ;BR IF MISSING
1310 006766 011005                                     MOV (R0),R5      ;MOVE THE INFO TO R5
1311 006770 000137 007000                                     JMP 6$              ;GO BACK
1312 006774 022020 4$: CMP (R0)+,(R0)+      ;POP POINTER
1313 006776 011005                                     MOV (R0),R5      ;LOAD DATA TO R5
1314 007000 000201 6$: RTS R1              ;RETURN
1315
1316                                     :ROUTINE TO SET UP INTERRUPT VECTORS
1317
1318 007002 012577 172372 SETVEC: MOV (R5)+,20UPRVC
1319 007006 012577 172372                             MOV (R5)+,20UPTVC
1320 007012 112577 172364                             MOV8 (R5)+,20UPRPS
1321 007016 112577 172364                             MOV8 (R5)+,20UPTPS
1322 007022 000205                                     RTS R5
1323 007024 NO.ATRAP:
1324 007024 104001 HLT 1
1325 007026 000002 RTI
1326
1327 007030 NO.BTRAP:
1328 007030 104002 HLT 2
1329 007032 000002 RTI
1330

```

```

1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346 007034 012737 000001 001226
1347 007042 012737 007370 001216
1348 007050 105737 001322
1349 007054 001002
1350 007056 000137 007356
1351 007062 105737 001342
1352 007066 001040
1353 007070 012701 002010
1354 007074 105737 001334
1355 007100 001011
1356 007102 040137 007360
1357 007106 040137 007362
1358 007112 040137 007364
1359 007116 107337 001344
1360 007122 001113
1361 007124 105737 001336
1362 007130 001010
1363 007132 040137 007360
1364 007136 040137 007362
1365 007142 040137 007364
1366 007146 107337 001344
1367 007152 107337 001344
1368 007156 001004
1369 007160 007337 007360
1370 007164 007337 007362
1371 007170 004137 006756
1372 007174 007360
1373
1374 007176 005077 172206
1375 007202 002777 000400 172206
1376 007210 004737 004610
1377 007214 013703 001410
1378 007220 052777 010000 172170
1379 007225 052713 000016
1380 007229 012737 001010 007262
1381 007230 032777 004000 172152
1382 007246 001374
1383 007250 032777 004000 172142
1384 007253 001774
1385 007260 005327
1386 007262 000110

```

```

***** TEST 1 *****
*THIS TEST PROVES THE INTERACTION OF DTR!RTS!STD
*WITH RING DSR,CTS,CARDET,STD,SRO
*AND DATA SET CHANGE ONE AND DATA SET CHANGE TWO.
*SET THE BIT AND VERIFY THE OTHER BITS ARE SET. CLEAR
*THE BIT AND VERIFY CLEAR. REPEAT FOR MRESET.
*****

*****
TEST 1
*****
*****
TST1: MOV #1,2#TSTNO
      MOV #TST2,NEXT
      TSTB TCNCFG
      BNE 100$
      JMP 6$
100$: TSTB ALJMF
      BNE 12$
      MOV #STD!SRD,R1
      TSTB STJMF
      BNE 101$
      BIC R1,7$
      BIC R1,7$+2
      BIC R1,7$+4
      CLRB DSCFLG
      BR 102$
101$: TSTB SRJMF
      BNE 102$
      BIC R1,7$
      BIC R1,7$+2
      BIC R1,7$+4
      CLRB DSCFLG
102$: TSTB DSCFLG
      BNE 12$
      DEC 7$
      DEC 7$+2
12$: JSR R1,0JUMPER
      7$
      CLR 2#RXCSR
      BIS #MRESET,2#TXCSR
      JSR PC,5,ALL
      MOV RXCSR,R3
      BIS #NEXT,2#TXCSR
      BIS #DTR!RTS!STD,(R3)
      MOV #110,68$
      BIT #TIMER,2#TXDBUF
      BNE 66$
      BIT #TIMER,2#TXDBUF
      BEQ 67$
      DEC (PC)+
      68$: 110

```

```

;THIS CALL DETERMINES IF TURNAROUND CONNECTOR
;AND OPTIONAL JUMPER ARE USED
;AND LOADS R5 (EXPECTED) ACCORDINGLY.

;RESET THE DEVICE
;WAIT FOR RESET TO FINISH
;LOAD THE RECEIVER CONTROL REGISTER TO R3.
;ENTER EXTERNAL MAINT. MODE
;TURN ON DTR!RTS!STD
;LOAD THE NUMBER
;CHECK THE TIMER BIT
;BR IF SET
;CHECK THE BIT
;BR IF CLEAR
;DECREMENT THE NUMBER
;OF TIMES TO REPEAT

```

1387	007264	001365			BNE	66\$		;BR IF MORE TO GO
1388	007266	011304			MOV	(R3),R4		;GET THE BITS FROM THE RXCSR
1389	007270	020504			CMP	R5,R4		;R5=GOOD R4=?
1390	007272	001401			BEQ	2\$		;BRANCH IF THEY MATCH
1391	007274	104003			HLT	3		;NO MATCH - SHOW OPR.
1392	007276	012737	007310	001220	2\$: MOV	#3\$,LOCK		;SW09 SETUP
1393	007304	042705	073016		BIC	#RING!CTS!CARDET!		;SRD!DSR!STD!RTS!DTR,R5
1394	007310	005013			3\$: CLR	(R3)		;CLEAR OUT THE REGISTER
1395	007312	012737	000005	007342	MOV	#5,73\$		;LOAD THE NUMBER
1396	007320	032777	004000	172072	71\$: BIT	#TIMER,2TXDBUF		;CHECK THE TIMER BIT
1397	007326	001374			BNE	71\$		;BR IF SET
1398	007330	032777	004000	172062	72\$: BIT	#TIMER,2TXDBUF		;CHECK THE BIT
1399	007336	001774			BEQ	72\$		;BR IF CLEAR
1400	007340	005327			DEC	(PC)+		;DECREMENT THE NUMBER
1401	007342	000005			73\$: 5			;OF TIMES TO REPEAT
1402	007344	001365			BNE	71\$		;BR IF MORE TO GO
1403	007346	011304			MOV	(R3),R4		;READ BACK THE REGISTER
1404	007350	020504			CMP	R5,R4		;R5=GOOD R4=?
1405	007352	001401			BEQ	6\$		;BRANCH IF ONLY THE DSC BITS ARE SET
1406	007354	104003			HLT	3		;NO-GO TELL OPR
1407	007356	104400			6\$: SCOPE			;SCOPE THE WHOLE TEST
1408	007360	173017			7\$: .WORD	173017		
1409	007362	173001			.WORD	173001		
1410	007364	001016			.WORD	1016		
1411	007366	000000			.WORD	0		

```

***** TEST 2 *****
;TEST OF THE DUP RUNNING A BINARY COUNT
;PATTERN WITH A CRC CALCULATION AS A SECONDARY STATION
*****

```

```

*****
;TEST 2
*****
*****
*****
1425 007370 012737 000002 001226 1$T2: MOV #2,2$TSTNO
1426 007376 012737 010360 001216 MOV #TST3,NEXT
1427 007404 052777 000400 172004 BIS #MRESET,2TXCSR ;RESET THE DEVICE
1428 007412 004737 004610 JSR PC,SMALL ;WAIT FOR RESET TO FINISH
1429 007416 105737 001322 TSTB TCNPLG
1430 007422 001403 BEQ 102$
1431 007424 012737 010070 007530 MOV #NEXT,103$+2
1432 007432 012737 000005 001244 102$: MOV #5,TEMP4
1433 007440 005001 CLR R1 ;CLEAR OUT DATA
1434 007442 012737 102010 010352 MOV #CRC.CCITT,XPOLY ;SET UP THE POLYNOMIAL
1435 007450 012737 177777 010776 MOV #-1,CALBCC ;SETUP FOR THE FIRST TIME
1436 007456 013737 010356 007500 16$: MOV CALBCC,20$ ;ALLOW FOR THE NEXT CHARACTER
1437 007464 010137 007476 MOV R1,17$ ;LOAD DATA
1438 007470 004537 010200 JSR R5,SIMBCC ;GO CALCULATE SOFTWARE BCC
1439 007474 007010 B. ;BASED ON THESE PARAMETERS
1440 007476 000001 17$: .BLKW 1 ;DATA
1441 007500 000001 20$: .BLKW 1 ;PREVIOUS BCC
1442 007502 105201 INCB R1 ;INCREMENT DATA

```

```

1443 007504 001364 BNE 16$ ;BR IF MORE TO GO
1444 007506 012737 000001 001236 MOV #1,TEMP1 ;LOAD DATA
1445 007514 005037 001240 CLR TEMP2 ;CLEAR EXPECTED
1446 007520 012737 000340 177776 MOV #340,PS ;PS = 7
1447 007526 052777 004000 171662 103$: BIS #SYSTST,ATXCSR ;ENTER SYSTEM TEST MODE
1448 007534 004537 007002 JSR RS,SETVEC ;LOAD INTERRUPT VECTORS
1449 007540 007734 11$ RECEIVER
1450 007542 010002 12$ TRANSMITTER
1451 007544 340 340 .BYTE 340,340 ;LEVEL
1452 007546 052777 000020 171634 BIS #RCVEN,ARXCSR ;TURN ON THE RECEIVER
1453 007554 052777 000100 171626 BIS #RINTEN,ARXCSR ;TURN ON REC INTERRUPT ENABLE
1454 007562 105777 171630 1$: TSTB ATXCSR ;TEST FOR TX DONE
1455 007566 100375 BPL 1$ ;BR IF NOT SET
1456 007570 052777 000020 171620 2$: BIS #SEND,ATXCSR ;TURN ON SEND
1457 007576 012777 000400 171614 MOV #TSOM,ATXDBUF ;TURN ON START OF MESSAGE
1458 007604 101$:
1459 007604 012737 000005 007634 MOV #5,68$ ;LOAD THE NUMBER
1460 007612 012777 004000 171600 66$: BIT #TIMER,ATXDBUF ;CHECK THE TIMER BIT
1461 007620 001374 BNE 66$ ;BR IF SET
1462 007622 032777 004000 171570 67$: BIT #TIMER,ATXDBUF ;CHECK THE BIT
1463 007630 001774 BEQ 67$ ;BR IF CLEAR
1464 007632 005327 DEC (PC)+ ;DECREMENT THE NUMBER
1465 007634 000005 68$: 5 ;OF TIMES TO REPEAT
1466 007636 001365 BNE 66$ ;BR IF MORE TO GO
1467 007640 005337 001244 DEC TEMP4
1468 007644 001001 BNE 3$
1469 007646 104004 HLT 4
1470 007650 105777 171542 3$: TSTB ATXCSR ;WAIT FOR DONE
1471 007654 100353 BPL 101$ ;BR IF NOT SET
1472 007656 012777 171536 4$: CLR ATXDBUF ;PUSH OUT DATA
1473 007662 012777 000100 171526 BIS #TXINTE,ATXCSR ;TURN ON TRANSMITTER INT ENABLE
1474 007670 005037 177776 CLR PS ;LOWER PROCESOR STATUS
1475 007674 5$:
1476 007674 012737 000040 007724 MOV #32,73$ ;LOAD THE NUMBER
1477 007702 032777 004000 171510 71$: BIT #TIMER,ATXDBUF ;CHECK THE TIMER BIT
1478 007710 001374 BNE 71$ ;BR IF SET
1479 007712 032777 004000 171500 72$: BIT #TIMER,ATXDBUF ;CHECK THE BIT
1480 007720 001774 BEQ 72$ ;BR IF CLEAR
1481 007722 005327 DEC (PC)+ ;DECREMENT THE NUMBER
1482 007724 000040 73$: 32 ;OF TIMES TO REPEAT
1483 007726 001365 BNE 71$ ;BR IF MORE TO GO
1484 007730 104004 HLT 4 ;FAILED TO INTERRUPT IN TIME
1485 007732 104400 6$: SCOPE ;SCOPE THIS TEST
1486
1487 ;INTERRUPT SERVICE ROUTINES
1488 -----
1489 ;RECEIVER:
1490 007734 017737 171452 001324 11$: MOV ARXDBUF,DATA ;GET THE REGISTER AND DATA
1491 007742 123737 001240 001324 CMPB TEMP2,DATA ;CHECK IT
1492 007750 001401 BEQ .+4 ;BR IF OK
1493 007752 104004 HLT 4 ;COMPARISON ERROR
1494 007754 105237 001240 INCB TEMP2 ;COUNT UP EXPECTED
1495 007760 105737 001240 TSTB TEMP2 ;CHECK TO SEE IF DONE
1496 007764 001005 BNE 7$ ;BR IF NO
1497 007766 004537 007002 JSR RS,SETVEC ;YES--RESET THE VECTORS
1498 007772 010106 14$ RECEIVER

```

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 35
 DZDPEA.CMB END OF PASS ROUTINE

1499	007774	010002			12\$		: TRANSMITTER
1500	007776	340	340		.BYTE	340,340	: LEVEL
1501	010000	000002			7\$: RTI		: RETURN
1502					: TRANSMITTER:		
1503	010002	113777	001236	171410	12\$: MOV	TEMP1, @TXDBUF	: LOAD THE TRANSMITTER BUFFER
1504	010010	105237	001236		INCB	TEMP1	: UP THE COUNT
1505	010014	122737	000377	001236	CMPB	#377, TEMP1	: ARE WE DONE
1506	010022	001026			BNE	13\$	: BR IF NO
1507	010024	012777	010034	171352	MOV	#21\$, @DUPTVC	: SETUP FOR NEXT PART
1508	010032	000422			BR	13\$	: LEAVE
1509	010034	012777	000377	171356	21\$: MOV	#377, @TXDBUF	: LOAD BUFFER
1510	010042	012777	010052	171334	MOV	#22\$, @DUPTVC	: SETUP NEXT PART
1511	010050	000413			BR	13\$	: LEAVE
1512	010052	012777	001000	171340	22\$: MOV	#TEOM, @TXDBUF	: SET END OF MSG
1513	010060	000240			NOP		: STALL
1514	010062	000240			NOP		: DITTO
1515	010064	042777	000120	171324	BIC	#SEND!TXINTE, @TXCSR	: TURN OFF TRANSMITTER
1516	010072	012777	007030	171304	MOV	#NO.BTRAP, @DUPTVC	: LOAD VECTOR
1517	010100	012716	007674		13\$: MOV	#5\$, (SP)	: CRUNCH STACK
1518	010104	000002			RTI		: RETURNS
1519					: CRC CATCH INT SVC		
1520	010106	117737	171300	001324	14\$: MOV	@RXDBUF, DATA	: GET FIRST PART OF CRC
1521	010114	105777	171270		TSTB	@RXCSR	: WAIT FOR SECOND PART
1522	010120	100375			BPL	-4	: DITTO
1523	010122	117737	171264	001325	MOV	@RXDBUF, DATA+1	: GET THE REST OF THE CRC
1524	010130	012716	010136		MOV	#15\$, (SP)	: SETUP FOR RETURN
1525	010134	000002			RTI		: RETURN
1526	010136	012737	000340	177776	15\$: MOV	#340, PS	: RAISE PS
1527	010144	005137	010356		COM	CALBCC	: INVERT BCC
1528	010150	023737	010356	001324	CMP	CALBCC, DATA	: COMPARE SOFTWARE AND HARDWARE BCC
1529	010156	001401			BEQ	+4	: BR IF OK
1530	010160	104004			HLT	4	: BCC COMPARISON ERROR
1531	010162	052777	000400	171226	BIS	#MRESET, @TXCSR	: RESET THE DEVICE
1532	010170	004737	004610		JSR	PC, SMALL	: WAIT FOR RESET TO FINISH
1533	010174	000137	007732		JMP	6\$	: LEAVE
1534							
1535							
1536	010200	010046			SIMBCC: MOV	R0, -(SP)	
1537	010202	010146			MOV	R1, -(SP)	
1538	010204	010246			MOV	R2, -(SP)	
1539	010206	010237	001236		MOV	(R5)+, TEMP1	
1540	010212	010237	001240		MOV	(R5)+, TEMP2	
1541	010216	010237	001242		MOV	(R5)+, TEMP3	
1542	010222	005037	010354		1\$: CLR	BCCFBK	
1543	010226	013700	001242		MOV	TEMP3, R0	
1544	010232	006037	001240		ROR	TEMP2	
1545	010236	005500			ADC	R0	
1546	010240	032700	000001		BIT	#BIT0, R0	
1547	010244	001402			BEQ	2\$	
1548	010246	005137	010354		COM	BCCFBK	
1549	010252	013700	010352		2\$: MOV	XPOLY, R0	
1550	010256	005100			COM	R0	
1551	010260	040037	010354		BIC	R0, BCCFBK	
1552	010264	000241			CLC		
1553	010266	006037	001242		ROR	TEMP3	
1554	010272	013700	010354		MOV	BCCFBK, R0	

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 36
 DZDPEA.CMB END OF PASS ROUTINE

```

1555 010276 013701 001242      MOV      TEMP3,R1
1556 010302 010102      MOV      R1,R2
1557 010304 040100      BIC      R1,R0
1558 010306 043702 010354      BIC      BCCFBK,R2
1559 010312 050200      BIS      R2,R0
1560 010314 043737 010352 001242      BIC      XPOLY,TEMP3
1561 010322 050037 001242      BIS      R0,TEMP3
1562 010326 005337 001236      DEC      TEMP1
1563 010332 001333      BNE      1$
1564 010334 013737 001242 010356      MOV      TEMP3,CALBCC
1565 010342 012602      MOV      (SP)+,R2
1566 010344 012601      MOV      (SP)+,R1
1567 010346 012600      MOV      (SP)+,R0
1568 010350 000205      RTS      R5
1569 010352 000000      XPOLY: 0
1570 010354 000000      BCCFBK: 0
1571 010356 000000      CALBCC: 0
1572      120001      CRC16=120001
1573      102010      CRC.CCITT=102010
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587

```

```

***** TEST 3 *****
;THIS TEST PROVES THE DEVICE WILL HANDLE THE
;DDCMP PROTOCOL. SEND AND RECEIVE SYNCs,
;FOLLOWED BY DATA,BCC,DATA AND FINAL BCC.
*****

```

```

;*****
;
; TEST 3
;
;*****
*****
1588 010360 012737 000003 001226 TST3:  MOV      #3,0#TSTNO
1589 010366 012737 002646 001216      MOV      #.EOP_NEXT
1590 010374 105737 001322      TSTB      TCNFLAG
1591 010400 001403      BEQ      101$
1592 010402 012737 010000 010476      MOV      #NEXT,102$+2
1593 010410 012737 000340 177776 101$:  MOV      #340,PS      ;RAISE PROCESSOR STATUS
1594 010416 004537 007002      JSR      R5,SETVEC      ;SET UP VECTORS
1595 010422 011066      10$      ;BASED ON
1596 010424 010710      2$      ;THESE
1597 010426      340      340      .BYTE      340,340      ;PARAMETERS
1598 010430 005037 001236      CLR      TEMP1
1599 010434 005037 001240      CLR      TEMP2
1600 010440 005037 001242      CLR      TEMP3
1601 010444 005037 001244      CLR      TEMP4
1602 010450 005037 001246      CLR      TEMP5
1603 010454 052777 000400 170734      BIS      #MRESET,0TXCSR ;RESET THE DEVICE
1604 010462 004737 004610      JSR      PC,SMALL      ;WAIT FOR RESET TO FINISH
1605 010466 012777 100026 170720      MOV      #DECMOD!26,0PARCSR ;LOAD THE MODE AND SYNC CHARACTER
1606 010474      102$:
1607 010474 052777 004000 170714      BIS      #SYSTST,0TXCSR ;ENTER SYSTEM TEST MODE
1608 010502 052777 000420 170700      BIS      #RCVEN!STPSYN,0RXCSR ;LOAD RCVEN!STPSYN
1609 010510 052777 000020 170700      BIS      #SEND,0TXCSR ;TURN ON TRANSMITTER
1610 010516 012777 000426 170674      MOV      #TSM!26,0TXDBUF ;OUTPUT A SYNC CHAR

```



```

1611 010524 012737 000005 010554      MOV      #5,68$      ;LOAD THE NUMBER
1612 010532 032777 004000 170660 66$:  BIT      #TIMER,ATXDBUF ;CHECK THE TIMER BIT
1613 010540 001374      BNE      66$      ;BR IF SET
1614 010542 032777 004000 170650 67$:  BIT      #TIMER,ATXDBUF ;CHECK THE BIT
1615 010550 001774      BEQ      67$      ;BR IF CLEAR
1616 010552 005327      DEC      (PC)+      ;DECREMENT THE NUMBER
1617 010554 000005      S      68$:      ;OF TIMES TO REPEAT
1618 010556 001365      BNE      66$      ;BR IF MORE TO GO
1619 010560 105777 170632      TSTB     ATXCSR
1620 010564 100401      BMI      +4
1621 010566 104005      HLT      5      ;EXTERNAL CLOCKING STOPPED
1622 010570 012777 000426 170622      MOV      #TSOM!26,ATXDBUF
1623 010576 105777 170614 69$:  TSTB     ATXCSR      ;CHECK DONE
1624 010602 100375      BPL      69$      ;BR IF NOT SET
1625 010604 012777 000426 170606      MOV      #TSOM!26,ATXDBUF ;SEND SYNC
1626 010612 052777 000100 170570      BIS      #RINTEN,ATXCSR ;TURN ON INTERRUPTS
1627 010620 052777 000100 170570      BIS      #TXINTE,ATXCSR ;DITTO
1628 010626 005037 177776      CLR      PS      ;LOWER PROCESSOR STATUS
1629 010632
1630 010632 012737 000144 010662 100$:  MOV      #100,74$      ;LOAD THE NUMBER
1631 010640 032777 004000 170552 72$:  BIT      #TIMER,ATXDBUF ;CHECK THE TIMER BIT
1632 010646 001374      BNE      72$      ;BR IF SET
1633 010650 032777 004000 170542 73$:  BIT      #TIMER,ATXDBUF ;CHECK THE BIT
1634 010656 001774      BEQ      73$      ;BR IF CLEAR
1635 010660 005327      DEC      (PC)+      ;DECREMENT THE NUMBER
1636 010662 000144      74$:  100.      ;OF TIMES TO REPEAT
1637 010664 001365      BNE      72$      ;BR IF MORE TO GO
1638 010666 104005      HLT      5      ;FAILED TO FINISH TEST
1639 010670
1640 010670 052777 000400 170520 1$:   BIS      #MRESET,ATXCSR ;RESET THE DEVICE
1641 010676 004737 004610      JSR      PC,SMALL ;WAIT FOR RESET TO FINISH
1642 010702 012706 001150      MOV      #STACK,SP ;RESET THE STACK
1643 010706 104400      SCOPE ;SCOPE THIS TEST
1644
1645 ;INTERRUPT SERVICE ROUTINES
1646 ;TRANSMITTER
1647
1648 010710 012777 000252 170502 2$:   MOV      #252,ATXDBUF ;LOAD FIRST DATA CHAR
1649 010716 012737 000326 001236      MOV      #26,TEMP1 ;LOAD DATA
1650 010724 012777 010734 170452      MOV      #35,ADUPTVC ;RELOAD VECTOR
1651 010732 000452      BR      7$      ;LEAVE
1652 010734 013777 001236 170456 3$:   MOV      TEMP1,ATXDBUF ;MOV DATA TO BUFFER
1653 010742 105237 001236      INCB     TEMP1 ;UPDATE DATA
1654 010746 122737 000032 001236      CMPB     #32,TEMP1 ;CHECK FOR DONE
1655 010754 001041      BNE      7$      ;BR IF MORE TO SEND
1656 010756 012777 010766 170420      MOV      #45,ADUPTVC ;RELOAD VECTOR
1657 010764 000435      BR      7$      ;RETURN
1658 010766 012777 001000 170424 4$:   MOV      #TEOM,ATXDBUF ;PUT OUT BCC
1659 010774 012777 011004 170402      MOV      #55,ADUPTVC ;RELOAD VECTOR
1660 011002 000426      BR      7$      ;RETURN
1661 011004 013777 001240 170406 5$:   MOV      TEMP2,ATXDBUF ;LOAD DATA
1662 011012 105237 001240      INCB     TEMP2 ;UPDATE DATA
1663 011016 122737 000100 001240      CMPB     #100,TEMP2 ;CHECK FOR FINISH
1664 011024 001015      BNE      7$      ;BR IF MORE TO GO
1665 011026 012777 011036 170350      MOV      #65,ADUPTVC ;RELOAD VECTOR
1666 011034 000411      BR      7$      ;RETURN

```

1667	011036	012777	001000	170354	6\$:	MOV	#TEOM, @TXDBUF	: PUSH OUT DATA BCC
1668	011044	042777	000120	170344		BIC	#SEND!TXINTE, @TXCSR	: SHUT DOWN TRANSMITTER
1669	011052	012777	007030	170324		MOV	#NO.BTRAP, @DUPTVC	: RESET VECTOR
1670	011060	012716	010632		7\$:	MOV	#100\$, (SP)	: SETUP RETURN
1671	011064	000002				RTI		: RETURN
1672								
1673								
1674								: RECEIVER
1675	011066	017737	170316	001242	10\$:	MOV	@RXCSR, TEMP3	: SAVE CSR
1676	011074	017737	170312	001244		MOV	@RXDBUF, TEMP4	: SAVE BUFFER
1677	011102	105737	001242			TSTB	TEMP3	: CHECK FOR DONE
1678	011106	100401				BMI	11\$	: BR IF SET
1679	011110	104005				HLT	5	: FALSE INTERRUPT
1680	011112	005737	001244		11\$:	TST	TEMP4	: CHECK FOR ERROR
1681	011116	100001				BPL	12\$	: BR IF NO ERROR
1682	011120	104005				HLT	5	: RECEIVER ERROR
1683	011122	122737	000252	001244	12\$:	CMPB	#252, TEMP4	: CHECK DATA
1684	011130	001401				BEQ	13\$	: BR IF A MATCH
1685	011132	104005				HLT	5	: DATA COMPARE ERROR
1686	011134	012737	000026	001246	13\$:	MOV	#26, TEMP5	: LOAD NEXT EXPECTED
1687	011142	012777	011152	170230		MOV	#14\$, @DUPRVC	: RELOAD VECTOR
1688	011150	000531				BR	26\$	: LEAVE
1689	011152	017737	170234	001244	14\$:	MOV	@RXDBUF, TEMP4	: GET DATA
1690	011160	005737	001244			TST	TEMP4	: CHECK FOR ERROR
1691	011164	100001				BPL	15\$	: BR IF NO ERROR
1692	011166	104005				HLT	5	: DATA ERROR
1693	011170	123737	001246	001244	15\$:	CMPB	TEMP5, TEMP4	: CHECK DATA
1694	011176	001401				BEQ	16\$	: BR IF A MATCH
1695	011200	104005				HLT	5	: DATA COMPARE ERROR
1696	011202	105237	001246		16\$:	INCB	TEMP5	: UPDATE DATA
1697	011206	122737	000032	001246		CMPB	#32, TEMP5	: CHECK FOR FIRST PART FINISH
1698	011214	001107				BNE	26\$	: BR IF MORE TO GO
1699	011216	012777	011226	170154		MOV	#17\$, @DUPRVC	: SET UP NEXT VECTOR
1700	011224	000503				BR	26\$	: LEAVE
1701	011226	017737	170160	001244	17\$:	MOV	@RXDBUF, TEMP4	: GET THE BUFFER
1702	011234	005737	001244			TST	TEMP4	: TEST FOR ERROR
1703	011240	100001				BPL	.+4	: BR IF OK
1704	011242	104005				HLT	5	: RECEIVER ERROR
1705	011244	012777	011254	170126		MOV	#18\$, @DUPRVC	: RELOAD THE VECTOR
1706	011252	000470				BR	26\$	: LEAVE
1707	011254	017737	170132	001324	18\$:	MOV	@RXDBUF, DATA	: GET DATA
1708	011262	032737	010000	001324		BIT	#CRCERR, DATA	: CHECK FOR CRC ERROR
1709	011270	001001				BNE	19\$	: BR IF OK
1710	011272	104005				HLT	5	: CRC ERROR!!!!!!
1711	011274	012777	011310	170076	19\$:	MOV	#20\$, @DUPRVC	: SET UP VECTOR
1712	011302	005037	001330			CLR	MIND	: SETUP FOR NEXT DATA
1713	011306	000452				BR	26\$	: LEAVE
1714	011310	017737	170076	001244	20\$:	MOV	@RXDBUF, TEMP4	: GET DATA
1715	011316	005737	001244			TST	TEMP4	: CHECK FOR ERROR
1716	011322	100001				BPL	21\$	: BR IF NO ERROR
1717	011324	104005				HLT	5	: RECEIVER ERROR
1718	011326	123737	001330	001244	21\$:	CMPB	MIND, TEMP4	: CHECK DATA
1719	011334	001401				BEQ	22\$	: BR IF A MATCH
1720	011336	104005				HLT	5	: DATA ERROR
1721	011340	105237	001330		22\$:	INCB	MIND	: UPDATE SOFTWARE DATA
1722	011344	122737	000100	001330		CMPB	#100, MIND	: CHECK FOR FINISH

PC	Instruction	PC	Instruction	PC	Instruction	PC	Instruction
1723	011352 001030		BNE	26\$	;BR IF MORE TO GO		
1724	011354 012777	011364 170016	MOV	#23\$,ADUPRVC	;RELOAD FINAL VECTOR		
1725	011362 000424		BR	26\$	;LEAVE		
1726	011364 017737	170022 001244	MOV	ARXDBUF,TEMP4	;GET DATA		
1727	011372 005737	001244	TST	TEMP4	;CHECK FOR ERROR		
1728	011376 100001		BPL	24\$	;BR IF OK		
1729	011400 104005		HLT	5	;RECEIVER ERROR ON FIRST OCTET		
1730					;OF SECOND BCC		
1731	011402 105777	170002	TSTB	ARXCSR	;TEST DONE		
1732	011406 109375		BPL	24\$	;BR IF NOT SET		
1733	011410 017737	167776 001324	MOV	ARXDBUF,DATA	;GET SECOND BCC OCTET		
1734	011416 032737	010000 001324	BIT	#CRCERR,DATA	;CHECK FOR BCC ERROR		
1735	011424 001001		BNE	25\$	;BR IF OK		
1736	011426 104005		HLT	5	;BCC ERROR ON SECOND PART OF MSG		
1737	011430 012716	010670	MOV	#1\$, (SP)	;SETUP TO FINISH TEST		
1738	011434 000002		RTI		;RETURN		
1739							
1740							

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 40
 DZDPEA.CMB END OF PASS ROUTINE

1741	011436	052777	042516	050130	EM1:	.ASCIZ	<377>/UNEXPECTED INTERRUPT ON VECTOR "A" /
(1)	011503	377	047125	054105	EM2:	.ASCIZ	<377>/UNEXPECTED INTERRUPT ON VECTOR "B" /
(1)	011550	042777	051122	051117	EM3:	.ASCIZ	<377>/ERROR WHEN USING MODEM LEADS /
(1)	011607	377	051105	047522	EM4:	.ASCIZ	<377>/ERROR IN SDLC /
(1)	011627	377	051105	047522	EM5:	.ASCIZ	<377>/ERROR IN DECMODE /
(1)	011652	051377	047125	042040	EM6:	.ASCIZ	<377>/RUN DIAGNOSTICS /
(1)							
(1)					.EVEN		
(1)	011674				.ERRTAB:		
(1)	011674	000000			0		
(1)	011676	000000			0		
(1)	011700	000000			0		
(1)	011702	011436			EM1		
(1)	011704	011652			EM6		;HALT 1
(1)	011706	000000			0		
(1)							
(1)	011710	011503			EM2		
(1)	011712	011652			EM6		;HALT 2
(1)	011714	000000			0		
(1)							
(1)	011716	011550			EM3		
(1)	011720	011652			EM6		;HALT 3
(1)	011722	000000			0		
(1)							
(1)	011724	011607			EM4		
(1)	011726	011652			EM6		;HALT 4
(1)	011730	000000			0		
(1)							
(1)	011732	011627			EM5		
(1)	011734	011652			EM6		;HALT 5
(1)	011736	000000			0		
(1)	011740						
1742		000001			CORMAX:		
					.END		

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 42
 DZDPEA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

ADRCNT= 003617	925*	961*	970*					
ALJMF 001342	533*	657*	681*	693	694	1351		
BCCFBK 010354	1542*	1548*	1551*	1554	1558	1570*		
BINWRD 004122	985*	986	1023*					
BITW = 002000	533*							
BIT0 = 000001	466*	533	1546					
BIT1 = 000002	465*	533						
BIT10 = 002000	456*	533						
BIT11 = 004000	455*	533						
BIT12 = 010000	454*	533						
BIT13 = 020000	453*	533						
BIT14 = 040000	452*	533	843					
BIT15 = 100000	451*	533						
BIT2 = 000004	464*	533	781					
BIT3 = 000010	463*	533						
BIT4 = 000020	462*	533						
BIT5 = 000040	461*	533						
BIT6 = 000100	460*	533						
BIT7 = 000200	459*	533						
BIT8 = 000400	458*	533						
BIT9 = 001000	457*	533						
BRW 003170	787	865*						
BRX 003172	788	866*						
CALBCC 010356	1435*	1436	1527*	1528	1564*	1571*		
CARDCT= 010000	533*	1393						
CHARCT 004120	92*	987	1003*	1021*	1022			
CLK = 020000	533*							
CLK.A 001437	601*	1248						
CLVREC 004650	1157*							
CNVRT = 104411	570*	804	806	808	810	1096	1098	
CONVRT= 104410	568*	740	1112					
COR-MAX 011740	1741*	1742						
CRCEN = 001000	533*							
CRCERR= 010000	533*	1708	1734					
CRC.CC= 102010	1434	1573*						
CRC16 = 120001	1572*							
CREAM 001316	533*	623*	1202*	1203	1205*	1210	1211*	1212
CTS = 020000	533*	1393						
CYCLE 006166	789	824	825	1192*				
DATA 001324	533*	1490*	1491	1520*	1523*	1528	1707*	1708
DATABP 004474	1085*	1088	1110	1113*				
DATA ID 004462	1084*	1106	1109*					
DECMOD= 100000	533*	1605						
DEVADR 003614	923*	958	968*					
DISPLA 001200	510*	636*	801*					
DISPRE 000174	498*	636						
DSCA = 100000	533*							
DSCB = 070001	533*							
DSCFLG 011344	533*	711	1359*	1366*	1367			
DSINTE= 000040	533*							
DSR = 001000	533*	1393						
DTR = 000002	533*	1379	1393					
DUPACT 001310	533*	721*	762*	763	1192	1197		
DUPCR0 001500	604*	663	724					
DUPCR1 001506	604*	725						
DUPCR2 001514	604*							

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 43
 DZDPEA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

DUPCR3	001522	604#																	
DUPCR4	001530	604#																	
DUPCR5	001536	604#																	
DUPCR6	001544	604#																	
DUPCR7	001552	604#																	
DUPNUM	001311	533#	619	713#	814														
DUPRPS	001402	580#	1238#	1239#	1240	1320#													
DUPRVC	001400	579#	831	1217#	1238	1318#	1687#	1699#	1705#	1711#	1724#								
DUPSEC	001422	588#	1224#																
DUPTPS	001406	582#	1242#	1243#	1321#														
DUPTR0	001502	604#	671																
DUPTR1	001510	604#																	
DUPTR2	001516	604#																	
DUPTR3	001524	604#																	
DUPTR4	001532	604#																	
DUPTR5	001540	604#																	
DUPTR6	001546	604#																	
DUPTR7	001554	604#																	
DUPTVC	001404	581#	1240#	1241#	1242	1319#	1507#	1510#	1516#	1650#	1656#	1659#	1665#	1669#					
DUP.EN	001560	604#	654	1203	1212														
DUP.MA	001500	533	604#	623	652	737	1205	1215											
DUP0.A	001504	604#																	
DUP1.A	001512	604#																	
DUP2.A	001520	604#																	
DUP3.A	001526	604#																	
DUP4.A	001534	604#																	
DUP5.A	001542	604#																	
DUP6.A	001550	604#																	
DUP7.A	001556	604#																	
EM1	011436	1741#																	
EM2	011503	1741#																	
EM3	011550	1741#																	
EM4	011607	1741#																	
EM5	011627	1741#																	
EM6	011652	1741#																	
ERRCNT	001232	531#	625#	837	1123#														
ERRFLG	001347	539#	621#	799#	857#	1073#	1086	1100#											
ERRMSG	004450	1083#	1101	1104#															
ERTAB0	004574	1098	1132#																
EXIT =	000000	467#																	
EXITER	004530	1118	1123#																
FIX.00	006704	1249	1283#																
FLAG	001332	533#	650#	746	749#														
HALTS	004500	1069	1115#																
HOXEN =	000010	533#																	
HILIM	003612	922#	949	967#															
HUPPSR	001424	589#	1228#	1229															
HUP.F	001426	590#	1226#	1227#	1228														
HUP.FOR	001430	591#	1220#	1221#	1222														
HUPTBF	001432	592#	1235#	1236#															
HUPTCR	001434	593#	1231#	1232#	1233														
ICOUNT	001222	527#	855	860#															
INCJF	006122	892	928	1032#	1033	1037	1179#												
INIFLG	001346	508#	640	645#	651#														
INSTER=	104404	560#	943	1042															
INSTR =	104403	558#	658	666	674	684	690	696	700	704	708	1255							

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 44
 DZDPEA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

INSTR2	003412	899	911#						
JUMPER	006710	1291#							
LIMITS	003540	938	949#						
LOBITS	003616	924*	953	969#	970				
LOCK	001220	526#	859*	867	1092	1392*			
LOKFLG	001350	540#							
LOLIM	003610	921*	971	966#					
LPCNT	001224	528#	4*	855	858*				
LSTERR	001234	532#	6*	798*	841*	1070	1072*		
LOO.OO	001440	603#	1248#	1246					
MALLJN	005663	691	1168#						
MASK.A	001436	599#	1247						
MASTEX	005155	1094	1168#						
MCRLF	004702	877	974	1090	1091	1099	1168#	1254	1271
MCSR	005237	659	1168#						
MCSRX	005105	803	1168#						
MDATA	006060	1001	1011	1177#					
MDMFLG	001340	533#	656*	677	678				
MDSC	005737	709	1168#						
MEPASS	004723	802	1168#						
MEPPPC	005232	1097	1168#						
MERAX	005132	809	1168#						
MEF92	004751	1168#	1194						
MEF93	005020	759	1168#						
NEXT =	010000	533#	1378	1431	1592				
NIND	001330	533#	1712*	1718	1721*	1722			
NJMPR	005271	697	1168#						
NLOCK	005256	783	1168#						
NMODE =	015000	533#							
NMODEA =	005000	533#							
NMODEB =	010000	533#							
NMODEH	005606	675	1168#						
MNEW	005157	754	1168#						
NPAR	005446	1168#							
NPASSX	005121	807	1168#						
NPOWER	004705	1152	1168#						
NQM	004676	909	1168#	1277					
NR	004746	790	1169#	1272					
NRESET =	000400	467#	533#	1375	1427	1531	1603	1640	
MSRJN	005525	705	1168#						
MSTJN	005472	701	1168#						
MTCN	005344	685	1168#						
MTDATA =	040000	533#							
MTITLE	001000	505#	644						
MTOTAL	005413	1168#							
MTSTN	005143	1095	1168#	1256					
MTSTPC	005044	1168#							
MVEC	005256	66*	1168#						
MVECX	005113	805	1168#						
NEXT	001216	525#	861	1128	1347*	1426*	1589*		
NO.ATR	007024	1323#							
NO.BTR	007030	1327#	1516	1669					
OJUMPE	006756	1307#	1371						
OPCLRJ	001323	533#	699	1294	1300				
OVRUN =	040000	533#							
PARAM =	104405	562#	660	668	1257				

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 45
 DZDPEA.CMB CROSS REFERENCE TABLE -- USER SYMBOLS

PARAM1 003460	927#	944												
PARBIT= 000000	467#													
PARCSR 001414	585#	1225*	1605*											
PARERR 003534	930	932	934	943#	950	952	954							
PASCNT 001230	530#	620*	800*	801	834									
PC =%000007	431#	819*	1141*	1164*	1249*	1284*	1376*	1385*	1400*	1428*	1464*	1481*	1532*	
	1604*	1616*	1635*	1641*										
PERFOR= 000000	467#													
POPR0 = 012600	445#	1122												
POP1SP= 005726	443#													
POP2SP= 022626	447#	863												
PRIRTY 001320	533#													
PRISEC= 010000	533#													
PS = 177776	436#	616*	777*	1446*	1474*	1526*	1593*	1628*						
PUSHRO= 010046	444#	1119												
PUSHIS= 005746	442#													
PUSH2S= 024646	446#													
PURUP 004624	1146	1148#												
QV.FLG 001351	541#	622*	813*	852										
RABORT= 007000	533#													
RCRCIN= 07.000	533#													
RCRC7T= 100000	533#													
RCVEN = 000020	533#	1452	1608											
RECACT= 004000	533#													
REOM = 001000	533#													
RESREG 004476	1111	1114#												
RESTRT 003000	812	816	824#											
RESOS = 104407	566#	1114												
RETURN 001214	524#	628*	789*	791	824*	861*	864	1128*	1130	1153	1270*	1280*	1281	
RING = 040000	533#	1393												
RINTEN= 000100	533#	1453	1626											
RSOM = 000400	533#													
RTS = 000004	533#	1379	1393											
RUN 001314	533#	624*	1197	1200*	1201*	1208*	1209*							
RXCSR 001410	583#	828	1216*	1220	1374*	1377	1452*	1453*	1521	1608*	1626*	1675	1731	
RXDBUF 001412	584#	1222*	1223*	1224	1225	1226	1490	1520	1523	1676	1689	1701	1707	
	1714	1726	1733											
RXDERR= 100000	533#													
RXDONE= 000200	533#													
RD =%000000	424#	652*	653*	654	724*	729	731	733	755*	763*	764*	766*	768*	
	770	771	842	849*	842*	971*	975	988*	992*	1002	1018*	1120*	1210*	
	1216	1217	1218	1219*	1239	1241	1243	1246*	1263*	1264	1266	1268	1270	
	1274	1275	1293	1291*	1296	1299	1302	1304	1307*	1310	1312	1313	1536	
	1543*	1545*	1546	1549*	1550*	1551	1554*	1557*	1559*	1561	1567*			
R1 =%000001	425#	725*	729*	730*	731*	732*	733*	767*	768	769*	770	815*	819	
	971*	976	983*	982	983	984	985	1017*	1158*	1159	1161	1162	1247*	
	1291	1305*	1307	1314*	1353*	1356	1357	1358	1363	1364	1365	1371*	1433*	
	1437	1442*	1537	1555*	1556	1557	1566*							
R2 =%000002	426#	971*	1157*	1159*	1160*	1248*	1538	1556*	1558*	1559	1565*			
R3 =%000003	427#	886	893*	903*	906*	907	912*	971*	977	989*	990*	991*	992	
	1001*	1002*	1007*	1010*	1016*	1283*	1377*	1379*	1388	1394*	1403			
R4 =%000004	428#	887	892*	896*	897*	898	905*	908	911*	919	928*	929	931	
	933	935*	936	937	959*	959*	963*	971*	978	986*	989	994*	996*	
	998#	1015*	1077*	1078*	1079*	1080*	1081*	1082*	1083	1084	1085	1388*	1389	
	1403#	1404												
R5 =%000005	429#	870	871*	875	880	882*	918	920*	921	922	923	924	925	

		926	927*	936*	939*	940*	941*	949	951	953	959	960*	964*	971*
		979	987*	999*	1014*	1031*	1035*	1039*	1075*	1076*	1077	1079	1296*	1302*
		1304*	1310*	1313*	1318	1319	1320	1321	1322*	1389	1393*	1404	1438*	1448*
		1497*	1539	1540	1541	1568*	1594*							
SAYACT	001312	533#	712*	718*	720	721	757							
SAYNUM	001313	533#	619*	714*	811*	814*								
SAYPC	001266	533#	971*	1134										
SAYPCA	001306	533#												
SAVRO	001250	533#	971*											
SAVRQA	001270	533#												
SAVR1	001252	533#	971*											
SAVR1A	001272	533#												
SAVR2	001254	533#	971*											
SAVR2A	001274	533#												
SAVR3	001256	533#	971*											
SAVR3A	001276	533#												
SAVR4	001260	533#	971*											
SAVR4A	001300	533#												
SAVPS	001262	533#	971*											
SAVPSA	001302	533#												
SAVSP	001264	533#												
SAVS-PA	001304	533#												
SAVD5 =	104406	564#	1074											
SCOPE =	104400	552#	1407	1485	1643									
SCOP1 =	104401	554#												
SEND =	000020	533#	1456	1515	1609	1668								
SETFLG=	104412	572#	676	686	692	698	702	706	710					
SETVEC	007002	1318#	1448	1497	1594									
SHIFTS	001326	533#												
SIMBCC	010200	1438	1536#											
SMALL	004610	1140#	1376	1428	1532	1604	1641							
SP	=/000006	430#	617*	630*	631*	637	638	639	778*	842*	862	867*	870*	871
		872*	882	886*	887*	888	889*	905	906	907*	908*	911	912	918*
		919*	920	926*	963	964	971	975*	976*	977*	978*	979*	980	981*
		1014	1015	1016	1017	1018	1031	1040*	1051*	1052*	1053*	1054*	1055*	1056*
		1057*	1058	1070	1072	1075	1120	1129*	1149*	1517*	1524*	1536*	1537*	1538*
		1565	1566	1567	1642*	1670*	1737*							
SPACNT=	004121	984*	1005	1008*	1022#									
SRO =	002000	533#	1353	1393										
SRJHFL	001336	533#	707	1361										
STACK =	001150	437#	617	778	1129	1149	1642							
STD =	000010	533#	1353	1379	1393									
STJMFL	001334	533#	703	1354										

[illegible]

[illegible]

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 50
 DZDPER.CMB CROSS REFERENCE TABLE -- MACRO NAMES

DUPEND	18	792												
DUPFRN	18	385												
HLT	4488	1324	1328	1391	1406	1469	1484	1493	1530	1621	1638	1679	1682	1685
	1695	1704	1710	1717	1720	1729	1736							
OOO	18													
SBINCT	3858	1412												
SBT	3858	1331												
SBUFFE	18	1174												
SCLRVE	18	1155												
SCYCLE	18	1183												
SDECST	3858	1603												
SDEC22	3858	1575												
SEOP	18	792												
SEARRA	18													
SFINI	18	1741												
SGETFL	18	674	684	690	696	700	704	708						
SGETPA	18	658	666	1255										
SHEADE	18	385												
SJMPR	3858	195												
SMSG	18	1168												
SMSGA	17418													
SPFAIL	18	1143												
SQUEST	18	650												
SRESET	18	1375	1427	1531	1603	1639								
SSCOPE	18	838												
SSETFL	18	1025												
SSETVE	3858	1316												
SSIMBC	3858	1536												
SSMALL	18	1138												
SSYNC	3858	1623												
STRPDE	18	552	554	556	558	560	562	564	566	568	570	572		
STSTN	18	1340	1419	1582										
SVARIA	18	503												
SWAIT	18	1380	1395	1458	1475	1611	1630							
SXZ	18	1332	1338	1414	1417	1576	1580							

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 52
 DZDPEA.CMB CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ADC	1545														
ADCB	1201	1209													
ADD	730	732	742	872	889	960	981	991	1040	1056	1079	1082	1202	1211	1239
	1241	1243													
ASL	929	940	941	1054	1078	1080									
BCC	728														
BEQ	643	679	689	716	739	745	753	782	816	844	853	867	899	930	938
	1006	1064	1071	1087	1093	1102	1107	1111	1116	1127	1253	1293	1295	1301	1309
	1384	1390	1399	1405	1430	1463	1480	1492	1529	1547	1591	1615	1634	1684	1694
	1719														
BGT	934														
BHI	950														
BIC	764	990	1032	1055	1081	1356	1357	1358	1363	1364	1365	1393	1515	1551	1557
	1558	1560	1668												
BICB	897	935													
BIS	1375	1378	1379	1427	1447	1452	1453	1456	1473	1531	1559	1561	1603	1607	1608
	1609	1626	1627	1640											
BISB	936														
BIT	744	752	781	843	850	867	873	1063	1068	1124	1126	1252	1381	1383	1396
	1398	1460	1462	1477	1479	1546	1612	1614	1631	1633	1708	1734			
BITB	953	1197													
BLO	952														
BLOS	758														
BLT	932														
BMI	1620	1678													
BNE	641	655	695	747	751	772	780	812	851	856	874	881	904	954	962
	1000	1004	1009	1013	1034	1038	1069	1089	1125	1163	1193	1198	1204	1214	1251
	1265	1267	1269	1276	1349	1352	1355	1362	1368	1382	1387	1397	1402	1443	1461
	1466	1468	1478	1483	1496	1506	1563	1613	1618	1632	1637	1655	1664	1698	1709
	1723	1735													
BR	647	847	876	879	895	901	1066	1118	1455	1471	1522	1624	1681	1691	1703
	1716	1728	1732												
BR	634	682	719	734	743	761	786	845	849	910	942	944	1036	1043	1196
	1206	1273	1278	1360	1508	1511	1651	1657	1660	1666	1688	1700	1706	1713	1725
CLC	722	726	993	995	997	1199	1207	1552							
CLR	620	625	626	653	749	755	769	798	841	858	859	927	1160	1374	1394
	1433	1445	1472	1474	1542	1598	1599	1600	1601	1602	1628	1712			
CLRB	621	622	656	657	680	681	799	857	1010	1035	1073	1359	1366		
CMP	637	642	654	770	771	855	949	951	1070	1115	1161	1162	1203	1212	1264
	1266	1268	1275	1299	1312	1389	1404	1528							
CMPB	757	898	929	931	933	937	1033	1037	1491	1505	1654	1663	1683	1693	1697
	1718	1722													
COM	1527	1548	1550												
COMB	645	650													
DEC	715	903	999	1012	1369	1370	1385	1400	1464	1467	1481	1562	1616	1635	
DECB	811	961	1003	1008											
EMT	448														
HALT	479	756	760	765	1121	1147	1195								
INC	800	854	1123	1221	1223	1227	1230	1232	1234	1236					
INCB	1442	1494	1504	1653	1662	1696	1721								
JMP	501	648	748	791	825	864	1058	1130	1153	1281	1298	1303	1311	1350	1533
JSR	819	1249	1371	1376	1428	1438	1448	1497	1532	1594	1604	1641			
MOV	616	617	618	623	627	628	630	631	632	635	636	638	639	652	683
	724	725	729	731	733	737	738	766	767	768	777	778	784	785	787
	788	789	801	815	824	842	848	860	861	862	867	870	871	882	886
	887	888	892	893	902	905	906	907	908	911	912	918	919	920	921

	922	923	926	928	958	959	963	964	971	975	976	977	978	979	980
	982	985	986	988	989	1001	1014	1015	1016	1017	1018	1031	1051	1053	1057
	1072	1075	1077	1083	1084	1085	1120	1128	1129	1146	1149	1150	1157	1158	1159
	1205	1210	1215	1216	1217	1218	1219	1220	1222	1224	1225	1226	1228	1229	1231
	1233	1235	1238	1240	1242	1246	1247	1248	1263	1270	1280	1283	1291	1296	1302
	1304	1307	1310	1313	1318	1319	1346	1347	1353	1377	1380	1398	1392	1395	1403
	1425	1426	1431	1432	1434	1435	1436	1437	1444	1446	1457	1459	1476	1490	1507
	1509	1510	1512	1516	1517	1524	1526	1536	1537	1538	1539	1540	1541	1543	1549
	1554	1555	1556	1564	1565	1566	1567	1588	1589	1592	1593	1605	1610	1611	1622
	1625	1630	1642	1648	1649	1650	1652	1656	1658	1659	1661	1665	1667	1669	1670
	1675	1676	1686	1687	1689	1699	1701	1705	1707	1711	1714	1724	1726	1733	1737
MOV B	619	624	651	712	713	714	720	721	762	763	813	814	880	896	924
	925	983	984	987	992	1002	1007	1039	1067	1100	1320	1321	1503	1520	1523
NOP	784	785	820	821	822	823	1140	1513	1514						
RESET	817	1148													
ROL B	718	1200	1208												
ROR	994	996	998	1544	1553										
ROR B	723	727													
RTI	867	883	913	965	971	1019	1041	1131	1325	1329	1501	1518	1525	1671	1738
RTS	1141	1164	1284	1305	1314	1322	1568								
SEC	717														
SUB	1052	1076													
TRAP	552	554	556	558	560	562	564	566	568	570	572				
TST	633	746	750	779	867	1088	1092	1101	1106	1110	1117	1250	1274	1680	1690
	1702	1715	1727												
TST B	640	646	678	688	694	846	852	875	878	894	900	1005	1065	1086	1192
	1292	1294	1300	1308	1348	1351	1354	1361	1367	1429	1454	1470	1495	1521	1590
	1619	1623	1677	1731											
.ASCIZ	505	1168	1741												
.BLKB	533														
.BLKW	533	604	1181	1440	1441										
.BYTE	538	539	540	541	599	601	664	665	672	673	827	830	833	836	1133
	1136	1169	1171	1261	1262	1451	1500	1597							
.ENABL	1	385													
.END	1742														
.ENDC	1333	1338	1339	1348	1388	1403	1415	1417							

L04

DZDPE-A MACY11 27(732) 21-OCT-76 16:11 PAGE 54
DZDPEA.CMB CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.SBTTL	385	419	469	503	608	792
.TITLE	398					
.WORD	533	1408	1409	1410	1411	

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*,DZDPEA.SEG/SOL/CRF/PAGNUM/NL:TOC=DZDPEA.MAC,DZDPEA.CMB
RUN-TIME: 15 21 3 SECONDS
RUN-TIME RATIO: 120/41=2.9
CORE USED: 18K (35 PAGES)

Spooler runtime 7 Seconds, 32 KCS, 104 disk reads, 0 disk writes, 50 pages

0000000011111111112222222223333333334444444445555555556666666667777777777888888888999999999000000000111111111222222222333312