

# DFC-11A

OFF-LINE EXERCISER  
MD-11-DZDFA-B

EP DZDFA-B-DL-A  
COPYRIGHT 1976  
FICHE 1 OF 1

NOV 1976  
**digital**  
MADE IN USA

The image shows a grid of 40 small, illegible data tables or charts arranged in 8 rows and 5 columns on the left side of the page. Each cell in the grid contains a small, dense table of data, likely representing test results or performance metrics. The text within these tables is too small to be read, but they appear to be organized into a structured format with multiple columns and rows of data points.

11-11-76

.REM 5

## IDENTIFICATIONS

PRODUCT CODE: MAINDEC-11-DZDFA-B-D  
PRODUCT NAME: DU11/DFC11 & DP11/DFC11 OFFLINE EXERCISOR  
DATE CREATED: APRIL, 1975  
MAINTAINER: DIAGNOSTIC GROUP  
AUTHOR: R. OCHESTER/J. EGOLF

FIRST PRINTING, APRIL, 1975

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 MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITAL'S COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

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

.REM \$

## 1.0 ABSTRACT

THIS MAINDEC CONSISTS OF TWO PROGRAMS TO CHECK OUT THE DFC11A. FIRST, A DP11A/DFC11 EXERCISER WHICH IS RUN WITH A DP11 TEST CONNECTOR IN PLACE OF THE MODEM. THE FUNCTION OF THIS TEST IS TO CHECK OUT THE DFC11A AND CABLE.

SECOND, THERE IS A DU11/DFC11 EXERCISER WHICH IS RUN WITH AN H315 TEST CONNECTOR IN PLACE OF THE MODEM. THE FUNCTION OF THIS TEST IS TO CHECK OUT THE DFC11A AND CABLE.

THESE TWO PROGRAMS ARE TOTALLY INDEPENDENT OF EACH OTHER AND ONLY ONE PROGRAM NEED BE RUN. THE INTENT IS TO HAVE ONLY ONE LISTING AND DOCUMENTATION.

THE TWO PROGRAMS OPERATE (MONITOR-WISE) LIKE THEIR DP11 OR DU11 DIAGNOSTIC COUNTERPARTS-- THE "MONITOR" OF EACH WILL FUNCTION SWITCH-WISE OR CONVERSATION-WISE IDENTICALLY.

## 2.0 REQUIREMENTS

## 2.1 ANY PDP11 FAMILY CPU WITH 4K MINIMUM

- . A TTY OR EQUIVALENT
- . A DFC11A
- . EITHER A DP11A OR DU11
- . EITHER A DP11 CABLE TERMINATOR OR H315 CONNECTOR RESPECTIVELY
- . MODEM CABLE

## 2.2 STORAGE

MINIMUM OF 4K

## 3.0 LOADING PROCEDURE

THE PROGRAM MAY BE LOADED LIKE ANY OTHER PROGRAM SUCH AS: PAPER TAPE, DEC TAPE, MAG TAPE, CASSETTE, DISC, ETC. MOST COMMON WILL BE PAPER TAPE LOADING THROUGH THE USE OF ABSOLUTE LOADER.

## 3.1 DFC11A TESTING

THIS PROGRAM(S) WILL EXERCISE DFC11A  
METHOD: INSERT DU11 OR DP11 AS REQUIRED  
INSERT DFC11 AND REQUIRED CABLE.

INSERT PROPER TEST CONNECTOR ON END

## OF DFC11 CABLE PER SECTION 1.0.

## 4.0 STARTING PROCEDURE

## 4.1 CONTROL SWITCH SETTINGS

## FOR THE DP11A/DFC11A

SWITCH 07 = 1 INDICATES SINGLE DP11 MODE WHEN STARTING  
(PLACE RESPECTIVE LINE NUMBER AND VECTOR  
IN SWR WHEN PROGRAM HALTS)  
= 0 INDICATES CYCLE MODE (MORE THAN ONE)  
LOC "BASCSR" AND "BASVEC" ARE USED TO PICK UP  
FIRST DP11 CSR + VECTOR. THIS MODE WILL BE USED  
WHEN UNDER ACT-11. NOTE: ALL DP11'S IN SYSTEM  
MUST BE CONFIGURED THE SAME IN THIS MODE OF OPERATION.

## FOR THE DU11/DFC11A

SWITCH 00 = 1 INDICATES RE-SELECT ADDRESSES AND VECTORS.  
(THIS IS A FLOATING ADDRESS DEVICE - MORE THAN  
LIKELY YOU SHOULD ALWAYS START WITH SW00= 1 AND  
ANSWER THE RESPECTIVE QUESTIONS.)  
= 0 NORMAL DEFAULT START OR RESTART.

## 4.2 STARTING ADDRESSES

200 = START ADDRESS FOR DU11/DFC11 TEST - CABLE & RESPECTIVE  
TERMINATOR MUST BE ON.

210 = START ADDRESS FOR DP11/DFC11 TEST - CABLE & RESPECTIVE  
TERMINATOR MUST BE ON.

1000 = RESTART FOR PRODUCTION RUNNING DU/DFC

SEE DFC11-A DOCUMENTATION FOR CORRECT JUMPERS & SWITCH  
SETTINGS FOR DU11 OR DP11. THIS IS IMPORTANT!!!!

## 4.3 OPERATOR ACTION

## 4.3.1 DP11A/DFC11A

IF SWITCH 07 WAS LEFT UP THE PROGRAM WILL HALT WITH  
THE SWITCH REG. IN THE DATA LIGHTS. (EXCLUDING 1105 &  
11/10 CPU) AND REQUIRE THE FOLLOWING OPERATOR ACTION:

- A. SW00-SW08 MUST BE SET TO THE VECTOR ADDRESS OF THE  
FIRST DP11. NOTE FIRST DP11 VECTOR, NOT  
THE LINE SELECTED UNLESS IT IS LINE 0.
- B. SW09-SW15 MUST BE SET TO THE OCTAL LINE NUMBER OF  
THE DP11 TO BE TESTED. E.G. THE FIRST  
DP11 IS LINE 0.

PRESS CONTINUE. RESET SWR AS PER 5.1.1. SWITCH SETTINGS.

# E01

DZDFA-B MACY11 27.732) 17-SEP-75 16:06 PAGE 5  
DZDFAB.P11

PRESS CONTINUE. THIS STARTS PROGRAM. ALL DP11 ADDRESSES SHALL BE ASSIGNED FROM 774400 TO 774770 (CONTIGUOUSLY)

C. SEE SECTION 4.3.2 PARAGRAPHS D,E,AND F FOR BAUD RATE AND SWITCH LOOP EXPLANATIONS.

D. SEE SECTION 8.3

## 4.3.2 DU11/DFC11-A

IF SWITCH 00 WAS ASSERTED AFTER LOADING ADDRESS 200: THE PROGRAM WILL ASK THE FOLLOWING QUESTIONS:

- A. FIRST DEVICE CSR ADDRESS: EG: TYPE 160040(160010 TO 164000)
- B. DEVICE VECTOR: EG: TYPE 300(300 TO 776)
- C. ARE YOU RUNNING MULTIPLE DEVICES(Y OR N): EG TYPE N  
IF YES WAS ANSWERED THE MONITOR WILL ASK FOR LAST DEVICE'S CSR ADDRESS. EG. TYPE 160060 THUS 3 DEVICES WILL BE RUN.  
IF WRONG ANSWERS WERE TYPED -- RESPECTIVE ERROR MESSAGES WILL BE TYPED. NOTE: UP TO 16 DU11/DFC11'S CAN BE RUN.
- D. DO YOU WANT TO RUN BAUD RATE TEST (Y OR N) EG: TYPE Y  
IF YES, THEN WILL ASK IF AC LINE IS 60 HERTZ (Y OR N) EG: ANSWER Y  
AT THIS POINT THERE WILL BE A 5 SECOND DELAY BEFORE THE BAUD RATE OR AN ERROR MESSAGE IS TYPED. NOTE: THE BAUD RATE CHECK CAN ONLY BE RUN ON ONE DEVICE.
- E. LOWER CONSOLE SWITCHES WITH THE EXCEPTION OF ERROR SWITCHES.
- F. IF RUNNING IN CLOCK RECOVERY MODE ABOVE 9600 BAUD PUT JP SW03 = 1 TO ELEIMINATE ERRONEOUS INTERMITTANT ERRORS.
- G. SEE SECTION 8.3

## 5.0 OPERATING PROCEDURE

### 5.1.1 SWITCH SETTINGS (APPLICABLE TO BOTH TESTS)

SW15 = 1	HALT ON ERROR
SW14 = 1	SCOPE LOOP FOR WHOLE CURRENT TEST
SW13 = 1	INHIBIT ERROR PRINTOUT
SW12 = 1	INHIBIT ALL PRINTOUT, BELL ON ERROR
SW11 = 1	INHIBIT ITERATIONS
SW10 = 1	ESCAPE TO NEXT TEST ON ERROR (DP11 ONLY)
SW07 = 1	SINGLE DP11 MODE (DP11 ONLY ON START UP) ALSO ASKS IF BAUD RATE CHECK IS WANTED.
SW03 = 1	PREVENT ERRONEOUS ERRORS WHEN RUNNING DU/DFC IN CLOCK RECOVERY ABOVE 9600 BAUD.
SW01 = 1	RETURN TO BAUD RATE TEST AFTER EOP. (SW11 MUST ALSO BE UP)
SW00 = 1	RESELECT ADDRESS & VECTORS AND ASKS IF BAUD RATE TEST IS WANTED (DU11 ONLY ON START UP). ANY RESTARTS NEED NOT START WITH SW01 =1 SINCE THE PROGRAM WILL REMEMBER ANSWERS.

## 6.0 ERRORS

### 6.1 ERROR PRINTOUT

PRINTS ALL ERRORS UNLESS INHIBITED BY SW13 OR SW12

ERROR PC: XXXXX THIS IS MEMORY ADDRESS +2 LOCATION. BACK UP 2 TO FIND HLT IN LISTING  
DEPENDING ON THE ERROR AN ADDITIONAL MESSAGE MAY BE TYPED OLT

## 6.2 ERROR RECOVERY

- A. IF IN A SCOPE LOOP, SET SW14
- B. TO RECOVER FROM HALT ON ERROR, DEPRESS CONTINUE
- C. IT MAY BE DESIRABLE TO SET SW13 OR SW12 IN AN ERROR  
CONDITION SO THAT AN OSCILLOSCOPE CAN BE PLACE ON LOGIC.

## 7.0 RESTRICTIONS

### 7.1 STARTING RESTRICTIONS

MOST LIKELY START:

START@ LOC 200 WITH SW00=1 FOR STARTING OF DU11/DFC11A  
SW00=0 FOR RESTART  
START@ LOC 210 WITH SW07=1 FOR SINGLE DP11A/DFC11A  
SW07=0 FOR ACT-11 & CYCLE MODE

SEE SECTIONS 4.1, 4.3.1, AND 4.3.2

## 8.0 MISCELLANEOUS

THE MONITORS ARE "CARBON COPIES" OF THE DP11 & DU11  
DIAGNOSTICS. THIS WAS TO KEEP UNIFORMITY IN THE OPERATING  
PROCEDURES, SINCE THE CHECK OUT OF THE DFC11A IS LARGELY  
DEPENDENT ON THE DEVICE DRIVING IT. KEEP IN MIND THAT  
THIS PROGRAM IS JUST AN EXERCISOR FOR THE DFC11A, FOR DETAILED  
TROUBLE-SHOOTING EXTENSIVE SCOPING WILL BE REQUIRED.

### 8.1 MULTIPLE DP11A/DFC11A

REFER TO DP11A DIAGNOSTICS

### 8.2 MULTIPLE DU11/DFC11A

REFER TO DU11 DIAGNOSTIC

### 8.3 SWITCH SETTINGS FOR THE DFC11A

THE FOLLOWING APPLIES:

- A. MAKE SURE THAT THE PROPER MODEM JUMPERS ARE IN FOR  
THE RESPECTIVE DEVICE THAT IS DRIVING IT. IE DP11 OR DU11.
- B. SELECT APPROPRIATE BAUD RATE (S2). THE BAUD RATE  
CAN BE SOFTWARE CHECKED IF A KW11 IS INSTALLED IN THE  
SYSTEM. USE AN OSCILLOSCOPE IF NO KW11 IS AVAILABLE.  
IF THE SOFTWARE BAUD RATE CHECK IS MADE, A DELAY

OF FIVE SECONDS WILL OCCUR AND THEN THE APPROPRIATE MESSAGE WILL BE TYPED OUT.

- C. SELECT APPROPRIATE OPTION MODES (S1). BY CAREFUL ANALYSIS OF THE RESPECTIVE PROGRAM, WRONG (S1) SWITCH SETTINGS WILL CAUSE A PREDETERMINED ERROR. IE S1-2 & S1-8 =0 WILL FAIL TO CLOCK DATA IN OR OUT.
- D. MAKE SURE THE DFC11/CABLE IS CORRECTLY TERMINATED.
- E. TYPICAL SWITCH & OPTION MODE SELECTIONS OF THE DFC11A ARE INCLUDED IN THE LISTING FOR QUICK REFERENCE.

#### 9.0 PROGRAM DESCRIPTION

SEE LISTING FOR THE DETAILS. HOWEVER THE DP11 & DU11 PROGRAMS BASICALLY CHECK EIA LINES AND INSURE THAT DATA IS TRANSMITTED & RECEIVED CORRECTLY.

\$

H01

DZDFA-B MACY11 27(732) 17-SEP-76 16:06 PAGE 8  
DZDFAB.P11

303  
304  
305  
306

35500 :DU11 AND DP11A ----DFC11A INTERFACE EXERCISER  
35600 :COPYRIGHT, DIGITAL EQUIPMENT CORPORATION\*\*\*\*\*  
35700 :MA NARD, MASSACHUSETTS 01754  
35800 :PROGRAMMER: R.CHESTER/J.EGOLF

307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362

.REM 2  
+++++  
THE FOLLOWING DESCRIBES THE SWITCH SETTINGS FOR THE DFC11-A  
.....HOWEVER READ ENGINEERING'S BLURBS FOR THE STRAIGHT SKINNY  
+++++  
BAUD RATE SELECTION NOTE 1: A SCOPE LOOP IS AVAILABLE  
TO VERIFY BAUD RATE IF NO KW11  
IS AVAILABLE.

S2-4=1 150 BAUD  
S2-3=1 300 BAUD  
S2-6=1 600 BAUD  
S2-5=1 1200 BAUD  
S2-8=1 2400 BAUD  
S2-10=1 4800 BAUD NOTE 2: ALL OTHERS MUST BE ZERO  
S2-7=1 9600 BAUD FOR THE BAUD RATE SELECTION  
S2-9=1 19200 BAUD  
+++++

OTHER SWITCH SELECTIONS  
S1-1=1 REC CLK TURNED ON BY OFF TO ON CARRIER TRANSITION  
S1-1=0 NO EFFECT BY CARRIER TRANSITION

S1-2=1 ENABLE DFC-11A  
S1-2=0 DISABLE DFC-11A

S1-3=1 REC CLK TURNED ON BY 1ST MARK TO SPACE TRANSITION  
S1-3=0 NO CLK DATA HAS NO EFFECT

S1-4=1 REC CLK RESYNCD BY ANY MARK TO SPACE TRANSITION  
S1-4=0 NOT SYNCRONIZED

S1-5=1 REC CLK TURNED OFF BY ON TO OFF CARRIER TRANSITION  
S1-5=0 TIME OUT AFTER NO DATA FOR .5 SEC

S1-8=1 EXTERNAL CLK  
S1-8=0 NOTHING

S1-9=1 CTS NO DELAY  
S1-9=0 CTS WITH DELAY BASED UPON THE SWITCH SETTINGS OF S1-6 & S1-7

S1-6	S1-7	
0	0	CTS DELAY =0 NOT APPLICABLE
0	1	CTS DELAY= .2 SEC
1	0	CTS DELAY= .1 SEC
1	1	CTS DELAY= .3 SEC

NOTE::::::::::S1-8 & S1-2 SWITCH SETTINGS ARE INTERRACTIVE !!!!!!!!

+++++

TYPICAL DFC11 SWITCH SETTINGS FOR THE DU11 AND DP11 ARE AS FOLLOWS:::

DU11 DP11

363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411  
412  
413  
414  
415  
416  
417  
418

S1-1,2,3,4,5,6,7,8,9,10  
1,1,1,1,1,0,0,0,1,1  
\* \* \* \*

S1-1,2,3,4,5,6,7,8,9,10  
1,1,1,1,1,0,0,0,1,1  
\* \* \* \*

\*= CUSTOMER OPTION

SET S2-1 THRU S2-10 FOR DESIRED BAUD RATE

+++++  
THE DFC11 MODEM JUMPERS SHALL BE CONFIGURED AS FOLLOWS:

NOTE: 0= JUMPER REMOVED  
1= JUMPER IN

JUMPER		FOR DU11/DFC11	FOR DP11/DFC11
NO.	NAME		
W1	EIA	0	0
W2	202	1	1
W3	EIA	0	0
W4	202	1	1
W5	811	0	0
W6	BSY	1	1
W7	301	1	1
W8	301	1	1

+++++

TERMINATOR:::::::::::THESE MUST BE ON THE DFC11 !!!!!!!  
DU11:: THE DFC11 SHALL BE TERMINATED WITH H315 CONNECTOR

DP11:: THE DFC11 SHALL BE TERMINATED WITH DP11 TERMINATOR

+++++

2

:DP11/DU11-DFC11A  
:COPYRIGHT APRIL 1975. DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754

:STARTING PROCEDURE  
:LOAD PROGRAM  
:LOAD ADDRESS 000200  
:PRESS START OR PRESS START WITH SW00=1 TO RESELECT  
:ADDRESSES AND VECTORS  
:THIS IS FOR THE DU11/DFC11 TEST

:LOAD ADDRESS 000210  
:PRESS START  
:THIS IS FOR THE DP11/DFC11 TEST

:FOR MORE INFORMATION SEE THE  
:DOCUMENTATION IN FRONT OF THE LISTING

:SWITCH REGISTER OPTIONS

SW15=100000 :=1,HALT ON ERROR

100000

419  
420  
421  
422  
423  
424  
425  
426  
427  
428  
429  
430  
431  
432  
433

040000  
020000  
010000  
004000  
002000  
001000  
000400  
000200  
000100  
000040  
000020  
000010  
000004  
000002  
000001

SW14=40000  
SW13=20000  
SW12=10000  
SW11=4000  
SW10=2000  
SW09=1000  
SW08=400  
SW07=200  
SW06=100  
SW05=40  
SW04=20  
SW03=10  
SW02=4  
SW01=2  
SW00=1

:=1, LOOP ON CURRENT TEST  
:=1, INHIBIT ERROR TYPEOUT  
:=1, DELETE TYPEOUT/BELL ON ERROR.  
:=1, INHIBIT ITERATIONS  
:=1, ESCAPE TO NEXT TEST ON ERROR  
:=1, LOOP WITH CURRENT DATA  
:=1, LOOP ON ERROR  
:=1, SINGLE SELECTED DP11. =0 CYCLE ALL DP11S

:=1, BAUD RATE TEST RESELECT - SW11 MUST ALSO BE UP.  
:=1, RESELECT ADDRESSES & VECTORS

434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482

000000  
000001  
000002  
000003  
000004  
000005  
000006  
000007

177570  
177570  
177776  
001050

005746  
005726  
010046  
012600  
024646  
022626

100000  
040000  
020000  
010000  
004000  
002000  
001000  
000400  
000200  
000100  
000040  
000020  
000010  
000004  
000002  
000001

;REGISTER DEFINITIONS

R0=%0 : GENERAL REGISTER  
R1=%1 : GENERAL REGISTER  
R2=%2 : GENERAL REGISTER  
R3=%3 : GENERAL REGISTER  
R4=%4 : GENERAL REGISTER  
R5=%5 : GENERAL REGISTER  
SP=%6 : PPROCESSOR STACK POINTER  
PC=%7 : PROGRAM COUNTER

;LOCATION EQUIVALENCIES

SWR=.77570 : CONSOLE SWITCH REGISTER  
LIGHTS=177570 : PCP-11 45 DISPLAY REGISTER  
PS=177776 : PROCESSOR STATUS WORD  
STACK=1050 : START OF PPROCESSOR STACK

;INSTRUCTION DEFINITIONS

PUSH1SP=5746 : DECREMENT PROCESSOR STACK 1 WORD  
POP1SP=5726 : INCREMENT PROCESSOR STACK 1 WORD  
PUSHR0=10046 : SAVE R0 ON STACK  
POP R0=12600 : RESTORE R0 FROM STACK  
PUSH.SP=24646 : DECREMENT STACK TWICE  
POP.SP=22626 : INCREMENT STACK TWICE  
.EQUIV EMT,HLT : BASIC DEFINITION OF ERROR CALL

BIT15=100000  
BIT14=40000  
BIT13=20000  
BIT12=10000  
BIT11=4000  
BIT10=2000  
BIT9=1000  
BIT8=400  
BIT7=200  
BIT6=100  
BIT5=40  
BIT4=20  
BIT3=10  
BIT2=4  
BIT1=2  
BIT0=1

# MO1

```

483                                     ;TRAPCATCAER FOR ILLEGAL INTERRUPTS
484                                     ;THE STANDARD "PC+2.....HALT" WILL BE PLACED
485                                     ;BETWEEN ADDRESS 0 AND 776. THIS WILL CATCH ANY
486                                     ;STRAY INTERUPTS.
487         000000
488
489                                     .:=0
490         000024         006414         59200         ;VECTOR INITIALIZATION
491         000026         000340         59700         .:=24
492         000030         010106         59800         .PFAIL           ;POWER FAIL VECTOR
493         000032         000340         59900         340             ;PRIORITY 7
494         000034         011266         60000         .HLT
495         000036         000340         60100         340
496                                     .TRPSRV
497                                     340
498
499         000200         005037         001074         60200         .:=200
500         000204         000137         001642         60300         CLR           PASCNT           ;CLEAR # OF PASSES
501                                     60400         JMP           DU11             ;GO TO START OF TEST
502
503         000210         012737         002370         001232         60500         .:=210
504         000216         005037         001156         60600         MOV           #DP11.A,DD.A     ;SETUP FOR RETURN AT START OF TEST
505         000222         005037         001074         60700         CLR           XLINEX           ;CLEAR LINES
506         000226         005037         001234         60800         CLR           PASCNT           ;CLEAR # OF PASSES
507         000232         000137         001374         60900         CLR           TAG              ;SETUP FOR BAUD RATE TEST
508                                     61000         JMP           DP11             ;GO TO START OF TEST
509
510         001000         000137         003110         61100         .:=1000
511                                     61200         JMP           DU11.A           ;PRODUCTION RESTART AT REGULAR DU TESTS
512                                     61300
513                                     61400         .:=1050
514                                     61500
515                                     61600
516                                     61700
517         001050         177560         TKCSR: 177560           ;TELETYPE KEYBOARD CONTROL REGISTER
518         001052         177562         TKDBR: 177562           ;TELETYPE KEYBOARD DATA BUFFER
519         001054         177564         TPCSR: 177564           ;TELEPRINTER CONTROL REGISTER
520         001056         177566         TPDBR: 177566           ;TELEPRINTER DATA BUFFER
521
522                                     ;PROGRAM CONTROL PARAMETERS
523
524         001060         000000         RETURN: 0               ;SCOPE ADDRESS FOR LOOP ON TEST
525         001062         000000         NEXT: 0                 ;ADDRESS OF NEXT TEST TO BE EXECUTED
526         001064         000000         LOCK: 0                 ;ADDRESS FOR LOCK ON CURRENT DATA
527         001066         000000         ICOUNT: 0              ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE
528         001070         000000         LPCNT: 0                ;NUMBER OF ITERATIONS COMPLETED
529         001072         000000         TSTNO: 0                ;NUMBER OF TEST IN PROGRESS
530         001074         000000         PASCNT: 0               ;NUMBER OF PASSES COMPLETED
531         001076         000000         ERRCNT: 0               ;TOTAL NUMBER OF ERRORS
532         001100         000000         LSTERR: 0               ;PC OF LAST ERROR CALL
533
534                                     ;PROGRAM VARIABLES
535
536         001102         000000         TEMP1: 0                ;TEMPORARY STORAGE
537         001104         000000         TEMP2: 0                ;TEMPORARY STORAGE
538         001106         000000         TEMP3: 0                ;TEMPORARY STORAGE

```

539	001110	000000	TEMP4:	0	; TEMPORARY STORAGE
540	001112	000000	TEMP5:	0	; TEMPORARY STORAGE
541	001114	000000	TEMP10:	0	
542	001116	000000	TEMP11:	0	
543	001120	000000	TEMP12:	0	
544	001122	000000	TEMP10:	0	
545	001124	000000	SAVR0:	0	; R0 STORAGE
546	001126	000000	SAVR1:	0	; R1 STORAGE
547	001130	000000	SAVR2:	0	; R2 STORAGE
548	001132	000000	SAVR3:	0	; R3 STORAGE
549	001134	000000	SAVR4:	0	; R4 STORAGE
550	001136	000000	SAVR5:	0	; R5 STORAGE
551	001140	000000	SAVSP:	0	; STACK POINTER STORAGE
552	001142	000000	SAVPC:	0	; PROGRAM COUNTER STORAGE
553	001144	000000	SAVSR1:	0	; NEW DEVICE STORAGE
554	001146	000000	TMPDAT:	0	; DP DATA STORAGE
555	001150	000000	SLIM:	0	
556	001152	000000	BPC:	0	
557	001154	000000	TSYNC:	0	; SYNC CHARACTER STORAGE
558	001156	000000	XLINEX:	0	; LINE NUMBER STORAGE
559	001160	000000	CABLE:	0	
560	001162	000000	TDATA:	0	; TRANSMITTER DATA STORAGE
561	001164	000000	RDATA:	0	; RECEIVE DATA STORAGE
562	001166	000000	CHLEN:	0	; CHARACTER LENGTH STORAGE
563	001170	000000	LIMIT:	0	; END OF DATA STORAGE
564	001172	000000	SCNT:	0	; SYNC COUNT STORAGE
565	001174	000000	SAVSR2:	0	; CONSOLE SWITCH STORAGE
566	001176	000000	TIME:	0	
567	001200	000000	TP:	0	
568	001202	000000	RP:	0	
569	001204	000000	BACK:	0	; PROGRAM RTI RETURN STORAGE
570	001206	000000	CONT.:	0	; RETURN TO BEGIN TEST STORAGE
571	001210	000000	DEVICE:	0	
572	001212	000000	NO.CLOCK:	0	; CLOCK STORAGE
573	001214	000	CPSFLG:	.BYTE 0	; 50/60 HERTZ STORAGE
574	001215	000	FLG19.2:	.BYTE 0	; HF CRYSTAL PROGRAM CONTROL STORAGE
575	001216	000	BAUDFLG:	.BYTE 0	; BAUD RATE TEST STORAGE
576	001217	000	INIF.A:	.BYTE 0	; ONCE ONLY CODE STORAGE
577	001220	000	BDRFLG:	.BYTE 0	; BAUD RATE FLAG STORAGE
578	001221	000	BDTSTFLG:	.BYTE 0	; BAUD RATE TEST STORAGE
579				.EVEN	
580	001222	000000	CHCNT:	.WORD 0	; CHARACTER COUNT STORAGE
581	001224	000000	CLKSET:	.WORD 0	; PRESET CYCLE COUNT STORAGE FOR KW11
582	001226	000000	INTCNT:	.WORD 0	; INTERRUPT COUNT STORAGE
583	001230	000000	CLKSTATUS:	.WORD 0	; KW11 CSR STORAGE
584	001232	000000	DD.A:	.WORD 0	; START OF TEST STORAGE
585	001234	000000	TAG:	.WORD 0	; CONTROL STORAGE FOR BAUD RATE TEST
586	001236	000000	PSTART:	.WORD 0	; POWER FAIL CONTROL STORAGE

001240  
001241  
001242  
001243  
  
001244  
001245  
001246  
001247  
001250  
001251  
  
001252  
001254  
001256  
001260  
001262  
001264  
  
001266  
  
  
  
  
  
  
  
  
001270  
001270  
001272  
001274  
001276  
001300  
001302  
001304  
001306

000  
000  
000  
000  
  
377  
377  
377  
377  
000  
377  
  
000000  
000000  
000000  
000000  
000000  
000000  
  
000000  
  
  
  
  
  
  
  
  
104400  
007626  
104401  
007552  
104402  
010026  
104403  
010044  
104404  
010612  
104405  
010652  
104406  
010412  
104407  
010416

62200  
62300  
62400  
62500  
62600  
62700  
62800  
62900  
63000  
63100  
63200  
63300  
63400  
63500  
63600  
63700  
63800  
63900  
64000  
64100  
64200  
64300  
64400  
64500

```

:PROGRAM CONTROL FLAGS
INIFLG: .BYTE 0 ;PROGRAM INITIALIZATION FLAG
STFLG: .BYTE 0 ;TEST START FLAG
ERRFLG: .BYTE 0 ;ERROR OCCURED FLAG
LOKFLG: .BYTE 0 ;LOCK ON CURRENT TEST FLAG

:PROGRAM CONVERSATIONAL PARAMETERS
SYNCNO: .BYTE 377 ;# OF SYNC CHARS REQ'D FOR SYNC'ZATION
SEXMIT: .BYTE 377 ;SEC XMIT JUMPER "IN"
SEREC: .BYTE 377 ;SEC REC JUMPER "IN"
OPTCLR: .BYTE 377 ;OPTIONAL JUMPER CLR "IN"
MULTD: .BYTE 0 ;NO MULTIPLE DEVICE FLAG
JMRBY: .BYTE 377 ;EXTERNAL MODEM BYPASS JUMPER "IN"
.EVEN

:PROGRAM MULTIPLE DEVICE PARAMETERS
BASEADD: 0 ;PROG CONTROLLED 1ST DEVICE ADDR
KEEPADD: 0 ;SAVED 1ST DEVICE ADDR
LASTADD: 0 ;LAST DEVICE RXCSR ADDR
BASEIV: 0 ;PROG CONTROLLED IV
KEEPIV: 0 ;SAVED INTR VECTOR
ACTREG: 0 ;ACTIVE REGISTER , MODIFY THIS
;LOCATION TO DISQUALIFY OR QUALIFY
;DEVICES (1= RUN , 0= DON'T RUN)
ROTADD: 0 ;ROTATING POINTER FOR ACTREG. POINTS
;TO DEVICE PRESENTLY UNDER TEST WHEN RUNNING MULTIPLE DEVICES

:PROGRAM CONTROL FLAGS

$Y=0

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

:*****
:*****
.TRPTAB:
SCOPE=TRAP+0 ;CALL TO SCOPE LOOP AND ITERATION HANDLER
.SCOPE
DELAY=TRAP+1 ;CALL TO DELAY FOR SPEC. TIME.
.DELAY
SCOPI=TRAP+2 ;CALL TO LOOP ON CURRENT DATA HANDLER
.SCOPI
TYPE=TRAP+3 ;CALL TO TELETYPE OUTPUT ROUTINE
.TYPE
SAVOS=TRAP+4 ;CALL TO REGISTER SAVE ROUTINE
.SAVOS
RESOS=TRAP+5 ;CALL TO REGISTER RESTORE ROUTINE
.RESOS
CONVRT=TRAP+6 ;CALL TO DATA OUTPUT ROUTINE
.CONVRT
CNVRT=TRAP+7 ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
.CONVRT
```

643 104410  
644 001310 011070  
645 104411  
646 001312 010704  
647 104412  
648 001314 011172  
649 104413  
650 001316 011622  
651 104414  
652 001320 011240

INSTR=TRAP+10 ;CALL TO ASCII STRING INPUT ROUTINE  
.INSTR  
PARAM=TRAP+11 ;CALL TO NUMERICAL DATA INPUT ROUTINE  
.PARAM  
SETFLG=TRAP+12 ;CALL TO FLAG SET ROUTINE  
.SETFLG  
INSTER=TRAP+13 ;CALL TO INPUT ERROR HANDLER  
.INSTER  
SCOP2=TRAP+14 ;CALL TO SEQUENCE TEST SCOPE LOOP  
.SCOP2

:\*\*\*\*\*  
:\*\*\*\*\*

653 100000  
654 040000  
655 020000  
656 010000  
657 004000  
658 002000  
659 001000  
660 000400  
661 000200  
662 000100  
663 000040  
664 000020  
665 000010  
666 000004  
667 000002  
668 000001  
669 100000  
670 040000  
671 020000  
672 010000  
673 001000  
674 000400  
675 000200  
676 000100  
677 000040  
678 000020  
679 000010  
680 000004  
681 000002  
682 000001  
683 030000  
684 020000  
685 000000  
686 000000  
687 002000  
688 004000  
689 006000  
690 000000  
691 001000  
692 001400  
693 100000  
694 040000  
695 020000  
696 002000  
697 000400  
698 000200

64700 :RXCSR BIT DEFINITIONS  
64800 DSC=BIT15 ;DATA SET CHANGE  
64900 RING=BIT14 ;RING  
65000 CTS=BIT13 ;CLR TO SEND  
65100 CARDET=BIT12 ;CARRIER DETECT  
65200 RECACT=BIT11 ;REC ACTIVE  
65300 SRD=BIT10 ;SEC REC DATA  
65400 DSR=BIT9 ;DATA SET RDY  
65500 STPSYN=BIT8 ;STRIP SYNC  
65600 RXDONE=BIT7 ;REC DONE  
65700 RINTEN=BIT6 ;REC INTR ENABLE  
65800 DSINTE=BIT5 ;DSC INTR ENABLE  
65900 SYN SCH=BIT4 ;SYNC SEARCH  
66000 STD=BIT3 ;SEC XMIT DATA  
66100 RTS=BIT2 ;REQ TO SEND  
66200 DTR=BIT1 ;DATA TERM RDY  
66300 VOID=BIT0  
66400 :RXDBUF BIT DEFINITIONS  
66500 RXERR=BIT15 ;REC ERROR  
66600 OVRUN=BIT14 ;OVERRUN  
66700 FRMERR=BIT13 ;FRAME ERROR  
66800 PARER=BIT12 ;PARITY ERROR  
66900 :PARCSR BIT DEFINITIONS  
67000 PAREN=BIT9 ;PARITY ENABLE  
67100 EVPAR=BIT8 ;EVEN PARITY SENSE  
67200 :PARCSR WPD DEFINITIONS  
67300 SYNINT=30000 ;SYNC EXTERNAL MODE  
67400 SYNEXT=20000 ;SYNC INTERNAL MODE  
67500 ISYMOD=0 ;ISOC MODE  
67600 FIVE=0 ;WORD LENGTH 5 BITS  
67700 SIX=2000 ;WORD LENGTH 6 BITS  
67800 SEVEN=4000 ;WORD LENGTH 7 BITS  
67900 EIGHT=6000 ;WORD LENGTH 8 BITS  
68000 NOPAR=0 ;NO PARITY  
68100 ODDPAR=1000 ;ODD PARITY  
68200 EVEPAR=1400 ;EVEN PARITY  
68300 :TXCSR BIT DEFINITIONS  
68400 DNA=BIT15 ;DATA NOT AVAILABLE  
68500 MTDATA=BIT14 ;MAINT DATA  
68600 CLK=BIT13 ;CLK  
68700 BITW=BIT10 ;BIT WINDOW  
68800 MRESET=BIT8 ;MASTER RESET  
68900 TXDONE=BIT7 ;XMIT DONE

699 000100  
700 000040  
701 000020  
702 000010  
703 000001  
704  
705 000000  
706 004000  
707 010000  
708 014000  
709  
710  
711  
712  
713 001322 000001  
714 001324 000001  
715 001326 000001  
716 001330 000001  
717 001332 000001  
718 001334 000001  
719  
720  
721  
722 001336 000001  
723 001340 000001  
724 001342 000001  
725 001344 000001  
726 001346 000300  
727 001350 174770  
728  
729  
730  
731  
732  
733  
734  
735  
736 001352 160040  
737 001354 160042  
738 001356 160042  
739 001360 160044  
740 001362 160046  
741  
742 001364 000770  
743 001366 000772  
744 001370 000774  
745 001372 000776  
746

69000  
69100  
69200  
69300  
69400  
69500  
69600  
69700  
69800  
69900  
70000  
70100  
70200  
70300  
70400  
70500  
70600  
70700  
70800  
70900  
71000  
71100  
71200  
71300  
71400  
71500  
71600  
71700  
71800  
71900  
72000  
72100  
72200  
72300  
72400  
72500  
72600  
72700  
72800  
72900  
73000  
73100  
73200  
73300  
73400  
73500  
73600

TXINTE=BIT6 ;XMIT INTR ENABLE  
DNAINTE=BITS ;DNA INTR ENAB  
SEND=BIT4 ;SEND  
HDXEN=BIT3 ;HDX/FDX  
BREAK=BIT0 ;BREAK  
;TXCSR WRD DEFINITIONS  
JSER=0 ;USER MODE  
MINT=4000 ;MAINT INT MODE  
MEXT=10000 ;MAINT EXT MODE  
SYSTST=14000 ;SYSTEM TEST MODE

\*\*\*\*\*

DPRS: .BLKW 1 ;DP11 RECEIVER STATUS  
DPRB: .BLKW 1 ;DP11 RECEIVER BUFFER  
SYNC: .BLKW 1 ;SYNC BUFFER  
DPTS: .BLKW 1 ;DP11 TRANSMITTER STATUS  
DPTB: .BLKW 1 ;DP11 TRANSMITTER BUFFER  
SEXT: .BLKW 1 ;DP11 SYNC EXTENSION

\*\*\*\*\*

DPRIV: .BLKW 1 ;DP11 RECEIVER INTERRUPT VECTOR  
DPRP: .BLKW 1 ;DP11 RECEIVER PRIORITY  
DPTIV: .BLKW 1 ;DP11 TRANSMITTER INTERRUPT VECTOR  
DPTP: .BLKW 1 ;DP11 TRANSMITTER PRIORITY  
BASVEC: 300 ;THIS IS THE FIRST VECTOR. PATCH FOR YOUR FIRST  
BASCSR: 174770 ;FIRST CSR ADDRESS.MAKE IT YOURS.

\*\*\*\*\*

;DU11 REGISTER POINTERS

\*\*\*\*\*

;DEFAULT DU ADDRESSES

RXCSR: 160040  
RXDBUF: 160042  
PARCSR: 160042  
TXCSR: 160044  
TXDBUF: 160046

;DEFAULT DU VECTORS

DURIV: 770 ;REC INTR VECTOR  
DURIS: 772 ;REC INTR STATUS  
DUTIV: 774 ;XMIT INTR VECTOR  
DUTIS: 776 ;XMIT INTR STATUS

# E02

DZDFA-6 MACY11 27(732) 17-SEP-75 16:06 PAGE 19  
 DZDFAB.P11

747				00200
748				00300
749	001374			00400
750	001374	000005		00500
751	001376	012737	001050	00600
752	001402	012737	000340	177776
753	001410	012737	001374	001236
754	001416	012737	002370	001060
755	001424	012737	002370	001062
756	001432	012737	011320	010242
757	001440	105037	001217	01200
758	001444	005137	001234	01300
759	001450	005037	001070	01400
760	001454	005037	001100	01500
761	001460	105037	001240	01600
762	001464	005037	001076	01700
763	001470	012737	000012	001066
764	001476	005327		01800
765	001500	000001		01900
766	001502	001002		02000
767	001504	104403	011563	02100
768	001510	105737	177570	02200
769	001514	100005		02300
770	001516	004737	006306	02400
771				02500
772				02600
773	001522	004737	006014	02700
774	001526	000404		02800
775	001530	004737	006306	02900
776	001534	004737	006064	03000
777	001540			03100
778	001540	012737	001632	001206
779	001546	105737	001240	03200
780	001552	001013		03300
781	001554	105137	001240	03400
782	001560	104410		03500
783	001562	013547		03600
784	001564	104412		
785	001566	001221		
786	001570	105737	001221	03800
787	001574	001402		03900
788	001576	000137	002636	04000
789	001602	012737	002370	001060
790	001610	032737	000002	177570
791	001616	001403		04100
792	001620	012737	002636	001060
793	001626	000177	177226	04200
794	001632	105737	001144	04300
795	001636	100740		04400
796	001640	000733		04500
797				04600
				04700
				04800

```

*****PART2 DB255 CONNECTOR TEST SECTION*****
DP11:
  RESET
  MOV #STACK,SP ;SET UP STACK POINTER
  MOV #340,PS ;SET PROCESSOR PRICRITY = 7
  MOV #DP11,PSTART ;SETUP POWER FAIL
  MOV #DP11.A,RETURN ;SETUP FOR REGULAR TEST
  MOV #CP11.A,NEXT ;DITTO
  MOV #EDP11,WHO
  CLRB INIF.A ;DP PROGRAM SETUP
  COM TAG ;DITTO
  CLR LPCNT ;DITTO
  CLR LSTERR ;DITTO
  CLRB INIFLG ;DITTO
  CLR ERRCNT ;DITTO
  MOV #10.,ICOUNT ;#OF ITERATIONS
  DEC (PC)+ ;ONCE ONLY
  .WORD 1
  BNE .+6
  TYPE ,DPTITLE
  TSTB SWR ;TEST FOR CHANGE IN DP ADRS
  BPL BGNDA ;BRANCH IF NO CHANGE
  JSR PC,CLAVEC ;LOAD ENTIRE VECTOR AREA WITH
  ; .+2
  ; HALT
  JSR PC,LINE.N ;FETCH LINE NUMBER FROM SWR
  BR PART2
  BGNDA: JSR PC,CLAVEC ;CLEAR OUT VECTOR AND GET NEW NUMBER
  JSR PC,LINE.X ;GET NEW LINE NUMBER
  PART2:
  MOV #DP.UPDATE,CONT. ;SETUP FOR RETURN
  TSTB INIFLG ;FIRST TIME?
  BNE PART3 ;NO
  COMB INIFLG ;YES
  INSTR ;OUTPUT MESSAGE & GET INPUT STRING
  MBDTEST ;MESSAGE
  SETFLG ;SET FLAG BASED UPON INPUT STRING
  BDTSTFLG ;THIS FLAG
  TSTB BDTSTFLG ;DO YOU WANT TO RUN BAUD TEST?
  BEQ .+6 ;NO
  JMP BAUD ;YES
  PART3: MOV #DP11.A,RETURN ;SETUP FOR RETURN TO START PROGRAM
  BIT #SW01,SWR
  BEQ 1$
  MOV #BAUD,RETURN
  1$: JMP @RETURN
  DP.UPDATE: TSTB SAVSR1 ;GO START
  BMI PART2 ;NEW DEVICE?
  BR BGNDA ;YES
  
```

```

798          00100
799          00200
800 001642 000005 003110 001060 00300
801 001644 012737 003110 001060 00400
802 001652 012737 003110 001062 00500
803 001660 013737 001352 001210 00600
804 001666 005037 001234 000000 00700
805 001672 105037 001217 000000 00800
806 001676 012737 000340 177776 00900
807 001704 012706 001050 000000 01000
808 001710 012737 001642 001236 01100
809 001716 012737 003110 001232 01200
810 001724 005037 001100 000000 01300
811 001730 005037 001076 000000 01400
812 001734 005037 001070 000000 01500
813 001740 012737 002000 001206 01600
814 001746 012737 000012 001066 01700
815 001754 012737 011345 010242 01800
816 001762 105737 001240 000000 01900
817 001766 001004 000000 000000 02000
818 001770 104403 011476 000000 02100
819 001774 105137 001240 000000 02200
820 002000 032737 000001 177570 02300
821 002006 001017 000000 000000 02400
822 002010 032737 000002 177570 02500
823 002016 001404 000000 000000 02600
824 002020 012737 004274 001060 02700
825 002026 000405 000000 000000 02800
826 002030 104403 012250 000000 02900
827 002034 012737 003110 001060 03000
828 002042 000177 177012 000000 03100
829 002046 000000 000000 000000 03200
830 002046 104410 000000 000000
831 002050 011672 000000 000000
832 002052 104411 000000 000000
833 002054 160000 000000 000000
834 002056 167776 000000 000000
835 002060 007270 000000 000000
836 002062 001 000000 000000
837 002063 001 000000 000000
838 002064 004737 006334 003400
839 002070 013737 007270 001254 03500
840 002076 004737 007200 000000 03600
841 002102 013737 001254 001252 03700
842 002110 104410 000000 000000
843 002112 011650 000000 000000
844 002114 104411 000000 000000
845 002116 000300 000000 000000
846 002120 000776 000000 000000
847 002122 001364 000000 000000
848 002124 001 000000 000000
849 002125 004 000000 000000
850 002126 013737 001364 001262 03900
851 002134 013737 001364 001260 04000
852 002142 104410 000000 000000
853 002144 011753 000000 000000

```

```

:***** PART3 *****
DU11:  RESET
      MOV #DU11.A,RETURN ;SETUP FOR REGULAR TEST
      MOV #DU11.A,NEXT ;DITTO
      MOV RXCSR,DEVICE ;SETUP FOR EOP MESSAGE
      CLR TAG
      CLRB INIF.A ;CLEAR OUT FOR FIRST TIME THRU
      MOV #340,PS ;SET PROCESSOR PRIORITY = 7
      MOV #STACK,SP ;SETUP STACK POINTER
      MOV #DU11,PSTART ;SET UP POWER FAIL
      MOV #DU11.A,DD.A ;SET UP FOR RETURN TO START OF TEST
      CLR LSTERR ;CLEAR LIST
      CLR ERRCNT ;CLEAR ERROR COUNT
      CLR LPCNT
      MOV #ONCE,CONT. ;SETUP FOR REPEAT
      MOV #10,COUNT ;SET ITERATIONS = 10
      MOV #EDU11,WHO
      TSTB INIFLG ;CHECK FOR FIRST TIME
      BNE ONCE ;NO
      TYPE DUTITLE ;YES
      COMB INIFLG
      BIT #SW00,SWR ;CHECK FOR CONVERSATIONAL
      BNE AAA ;YES
      BIT #SW01,SWR ;CHECK FOR BAUD RATE
      BEQ 1$ ;NO
      MOV #START,RETURN
      BR 2$
1$:  TYPE .MR ;LET OPERATOR KNOW TEST IS RUNNING
      MOV #DU11.A,RETURN ;GO TO START
2$:  JMP @RETURN
AAA:  INSTR ;OUTPUT MESSAGE & GET INPUT STRING
      MREGAD ;MESSAGE
      PARAM ;CONVERT STRING
      160000 ;LOW LIMIT
      167776 ;HIGH LIMIT
      DUBASE ;STORE AT THIS LOCATION
      .BYTE 1 ;MASK
      .BYTE 1 ;HOW MANY TIMES + 2
      JSR PC,OKADR ;CHECK FOR VALID ADDRESS
      MOV DUBASE,KEEPADD ;SAVE
      JSR PC,DUADDR ;GET ADDRESS
      MOV KEEPADD,BASEADD ;RESTORE FOR ROTATION
      INSTR ;OUTPUT MESSAGE & GET INPUT STRING
      MVECTO ;MESSAGE
      PARAM ;CONVERT STRING
      300 ;LOW LIMIT
      776 ;HIGH LIMIT
      DURIV ;STORE AT THIS LOCATION
      .BYTE 1 ;MASK
      .BYTE 4 ;HOW MANY TIMES + 2
      MOV DURIV,KEEPIV ;SAVE
      MOV DURIV,BASEIV ;SET UP FOR ROTATION
      INSTR ;OUTPUT MESSAGE & GET INPUT STRING
      MMULT ;MESSAGE

```

854	002146	104412				SETFLG		:SET FLAG BASED UPON INPUT STRING
855	002150	001250				MULTD		:THIS FLAG
856	002152	105737	001250		04200	TSTB	MULTD	:ARE THERE MULTIPLE DEVICES
857					04300			:ON THE SYSTEM ?
858	002156	100406			04400	BMI	BBB	:YES,ASK NEXT QUESTION
859	002160	005037	001264		04500	CLR	ACTREG	
860	002164	005037	001266		04600	CLR	ROTADD	:CLEAR POINTER
861	002170	000137	002334		04700	JMP	DDD	:JUMP AROUND NEXT QUESTION
862	002174				04800			
863	002174	104410				63B:	INSTR	:OUTPUT MESSAGE & GET INPUT STRING
864	002176	012032					MLASTD	:MESSAGE
865	002200	104411					PARAM	:CONVERT STRING
866	002202	160000					160000	:LOW LIMIT
867	002204	167776					167776	:HIGH LIMIT
868	002206	001256					LASTADD	:STORE AT THIS LOCATION
869	002210	001				.BYTE	1	:MASK
870	002211	001				.BYTE	1	:HOW MANY TIMES + 2
871					04900			:THE FOLLOWING ROUTINE SETS UP ACTREG FOR THE FIRST TIME
872	002212	012737	000001	001266	05000	1\$:	MOV #1,ROTADD	:SET UP POINTER
873	002220	005037	001264		05100		CLR ACTREG	:CLR ACTIVE REGISTER
874	002224	053737	001266	001264	05200	2\$:	BIS ROTADD,ACTREG	:MAKE THIS DEVICE ACTIVE
875	002232	000241			05300		CLC	
876	002234	006137	001266		05400		ROL ROTADD	:SET UP POINTER
877	002240	103421			05500		BCS 3\$	:ARE YOU OUT OF RANGE ?
878	002242	062737	000010	001252	05600		ADD #10,BASEADD	:SET UP BASE ADDRESS
879	002250	023737	001256	001252	05700		CMP LASTADD,BASEADD	:IS THIS THE LAST DEVICE ?
880	002256	101362			05800		BHI 2\$	:NO DO IT AGAIN
881	002260	053737	001266	001264	05900		BIS ROTADD,ACTREG	:THIS ASSUMES THAT THERE ARE AT
882					06000			:LEAST TWO DEVICES WHEN YOU ANSWER YES TO
883					06100			:MULTIPLE DEVICE QUESTION
884	002266	012737	000001	001266	06200	4\$:	MOV #1,ROTADD	:SET UP FOR LATER USE IN END OF PASS ROUTINE
885	002274	013737	001254	001252	06300		MOV #FE,PADD,BASEADD	:DITTO
886	002302	000425			06400		BR MUL	:CONTINUE QUESTIONS
887	002304	013737	001254	001252	06500	3\$:	MOV #PADD,BASEADD	:RESTORE
888	002312	104410					INSTR	:OUTPUT MESSAGE & GET INPUT STRING
889	002314	012162					MRANGE	:MESSAGE
890	002316	104411					PARAM	:CONVERT STRING
891	002320	160000					160000	:LOW LIMIT
892	002322	167776					167776	:HIGH LIMIT
893	002324	001256					LASTADD	:STORE AT THIS LOCATION
894	002326	001				.BYTE	1	:MASK
895	002327	001				.BYTE	1	:HOW MANY TIMES + 2
896	002330	000137	002212		06700		JMP 1\$	:DO IT AGAIN
897	002334				06800	DDD:		
898	002334	104410					INSTR	:OUTPUT MESSAGE & GET INPUT STRING
899	002336	013547					MBDTEST	:MESSAGE
900	002340	104412					SETFLG	:SET FLAG BASED UPON INPUT STRING
901	002342	001221					BDTSTFLG	:THIS FLAG
902	002344	105737	001221		07000		TSTB BDTSTFLG	:DO YOU WANT TO RUN BAUD TEST?
903	002350	001402			07100		BEQ .+6	:NO
904	002352	000137	004274		07200		JMP START	:YES
905	002356	012737	003110	001060	07300	OUTMUL:	MOV #DJ11.A,RETURN	:GO START REGULAR TEST
906	002364	000177	176470		07400		JMP @RETURN	
907					07500			
908								

```

909 00200
910 00300
911 00400
912 00500
913 00600
914 00700
915 00800
916 00900
917 01000
918 01100
919 01200
920 01300
921 01400
922 01500
923 01600
924 01700
925 01800
926 01900
927 02000
928 002370 02100
929
930
931
932
933
934
935 002370 012737 000001 001072
936 002376 012737 003124 001062
937 002404 012737 001632 001206 02300
938 002412 012737 002370 001060 02400
939 002420 012737 007736 001062 02500
940 002426 005077 176676 02600
941 002432 005077 176664 02700
942 002436 005002 02800
943 002440 012700 013000 02900
944 002444 010277 176662 03000
945 002450 017701 176654 03100
946 002454 042701 164777 03200
947 002460 022701 013000 03300
948 002464 001406 03400
949 002466 005202 03500
950 002470 001367 03600
951 002472 104001 03700
952 03800
953 002474 012737 000005 001120 03900
954 002502 005002 04000
955 002504 005202 04100
956 002506 001376 04200
957 002510 032777 013000 176612 04300
958 002516 001404 04400
959 002520 005337 001120 04500
960 002524 001366 04600
961 002526 104000 04700
962 04800
963 04900
964 05000

```

```

;CABLE TEST WITH OUT EXERCISING THE SOFTWARE CLOCK.
;IN THIS TEST FJUNCTIONS OF THE CABLE WILL BE
;TESTED WITHOUT THE SOFTWARE CLOCK. THE CLOCK MUST BE
;SUPPLIED BY THE DFC11.

: DFC11-A TEST!!!
;VERIFY THAT LOADING THE TRANSMITTER BUFFER
;BRINGS UP "REQUEST TO SEND" WHICH IN TURN WILL
;BRING UP "CLEAR TO SEND" AND "MODEM READY".
;VERIFY ALSO THAT THEY GO AWAY.
;NOTE: THE DFC11 CLOCK IS NECESSARY FOR
;THIS TEST TO WORK.

:*****:
: DP11 TEST FOR THE DFC11-A
:*****

DP11.A:
:*****
: TEST 1
:*****
:*****
TST1: MOV #1,TSTNO
MOV #TST2,NEXT
MOV #DP.UPDATE,CONT. ;SET FOR RETURN AT END PASS
MOV #DP11.A,RETURN ;SET RETURN ADDRESS.
MOV #.EOP,NEXT ;GOTO END PASS AT END OF TEST.
CLR @DPTS ;CLEAR THE TX STATUS
CLR @DPRS ;CLEAR THE RX STATUS.
CLR R2 ;SET TIME OUT
MOV #13000,R0
R2,@DPTB ;LOAD THE TX BUFFER
1$: MOV @DPTS,R1
BIC #C<13000>,R1
CMP #13000,R1
BEQ 2$
INC R2 ;UPDATE DELAY
BNE 1$ ;IS IT DONE?
HLT 1 ;ERROR REQUEST TO SEND,CLEAR TO
;SEND AND MODEM READY NOT UP.

2$: MOV #5,TEMP12
CLR R2 ;SET FOR TIME OUT
INC R2 ;DELAY
BNE .-2
BIT #13000,@DPTS ;ARE THEY GONE?
BEQ 3$
DEC TEMP12
BNE 2$
HLT

;VERIFY THAT THE SETTING OF "TERMINAL READY" BRINGS
;UP "RING" AND "CARRIER DOWN" ALSO VERIFY THAT

```

965				05100					:CLEARING "TERMINAL READY" BRINGS DOWN "RING"
966				05200					:AND "CARRIER DOWN".
967				05300					
968	002530	012777	000001	176572	05400	3\$:	MOV	#BIT0, @DPTS	:SET TERMINAL READY
969	002536	104401	001500		05500		DELAY	, 1500	:WAIT
970	002542	017701	176562		05600		MOV	@DPTS, R1	:GET WORD
971	002546	042701	153776		05700		BIC	#1C<24001>, R1	:CLEAR UNWANTED BITS
972	002552	022701	024001		05800		CMP	#24001, R1	:DID RING AND CARRIER DOWN SET?
973	002556	001401			05900		BEQ	. +4	:YES
974	002560	104001			06000		HLT	i	:NO--TYPE ERROR
975	002562	042777	000001	176540	06100		BIC	#BIT0, @DPTS	:CLEAR TERMINAL READY
976	002570	104401	001500		06200		DELAY	, 1500	:WAIT
977	002574	022777	120000	176526	06300		CMP	#120000, @DPTS	:DID THEY CLEAR?
978	002602	001401			06400		BEQ	. +4	:YES
979	002604	104000			06500		HLT		:NO--TYPE ERROR
980					06600				
981					06700				:NOW TEST THAT DATA CAN BE TRANSFERED.
982					06800				:A BINARY COUNT PATTERN WILL BE TRANSMITTED AND RECEIVED
983					06900				:WITH OUT THE USE OF THE SOFTWARE CLOCK.
984					07000				
985	002606	005037	001102		07100		CLR	TEMP1	:CLEAR OUT INFO
986	002612	005037	001104		07200		CLR	TEMP2	:DITTO
987	002616	004737	006450		07300		JSR	PC, SEQ. DATA	:GO RUN SEQUENTIAL DATA TEST
988	002622	104400			07400		SCOPE		

```

989          07600 ; THIS IS THE SETUP FOR THE BAUD RATE TEST USING THE DP
990          07700 ; *****
991 002624 012737 000372 001224 07800 NOT60: MOV #250,CLKSET ;250 HERTZ SETUP
992 002632 000137 004462          07900      JMP KWSETUP
993 002636 012737 000340 177776 08000 BAUD:  MOV #340,PS ;SET PROCESSOR STATUS = 7
994 002644 012706 001050          08100      MOV #STACK,SP ;SET UP STACK
995 002650 000005          08200 RX:    RESET
996 002652 012777 003004 176456 08300 DPSETUP:MOV #CON.RX,ADPRIV ;SETUP RX VECTOR
997 002660 012777 000340 176452 08400      MOV #340,ADPRP ;PROCESSOR STATUS = 7
998 002666 012777 00301E 176446 08500      MOV #N.TRAP,ADPTIV ;SETUP TX TRAPCATCHER
999 002674 012777 000340 176442 08600      MOV #340,ADPTP ;PROCESSOR STATUS = 7
1000 002702 012737 177777 001226 08700      MOV #-1,INTCNT ;ONCE ONLY
1001 002710 112777 000100 176404 08800 TX:    MOVB #100,ADPRS ;TURN ON RECEIVER
1002 002716 112777 000252 176402 08900      MOVB #252,ASYN ;SET SYNC CHARACTER
1003 002724 112777 000252 176400 09000      MOVB #252,ADPTB ;LOAD TX BUFFER
1004 002732 005037 001222          09100      CLR CHCNT ;CLEAR CHARACTER COUNTER
1005          09200
1006 002736 004737 005456          09300      JSR PC,CLOCK ;FIND OUT IF KW11 IS AVAILABLE
1007 002742 005737 001212          09400      TST NO.CLOCK
1008 002746 001002          09500      BNE 1$ ;YES
1009 002750 000137 003042          09600      JMP TST.Z ;NO,CHECK SW 14 FOR SCOPE LOOP
1010 002754          09700 1$:
1011 002754 104410          INSTR ;OUTPUT MESSAGE & GET INPUT STRING
1012 002756 012600          MCPS ;MESSAGE
1013 002760 104412          SETFLG ;SET FLAG BASED UPON INPUT STRING
1014 002762 001214          CPSFLG ;THIS FLAG
1015 002764 105737 001214          09800 TSTB CPSFLG ;IS AC 50 OR 60 HERTZ
1016 002770 001715          09900 BEQ NOT60 ;50 HERTZ
1017 002772 012737 000454 001224 10000 MOV #300,CLKSET ;60 HERTZ
1018 003000 000137 004462          10100 JMP KWSETUP ;GO SETUP KW
1019          10200
1020          10300 ; THIS IS THE BAUD RATE INTERRUPT SVC ROUTINE
1021          10400 ; *****
1022 003004 005237 001222          10500 CON.RX: INC CHCNT ;COUNT FOR BAUD RATE CHECK
1023 003010 005777 176310          10600      TST ADPRB ;CLR BUFFER
1024 003014 000002          10700      RTI
1025 003016 011637 006012          10800 N.TRAP: MOV (SP),TRP.PC ;TRAPCATCHER FOR TX INTERRUPT
1026 003022 104010          10900      HLT 10
1027 003024 000002          11000      RTI
1028 003026 032737 040000 177570 11100 SW14.B: BIT #BIT14,SWR ;FIND OUT IF BAUD RATE CHECK IS WANTED
1029 003034 001402          11200 BEQ TST.Z ;NOT YET
1030 003036 000137 003072          11300 JMP OUT.PT ;YES
1031 003042          11400 TST.Z:
1032 003042 104410          INSTR ;OUTPUT MESSAGE & GET INPUT STRING
1033 003044 012663          MRBAUD ;MESSAGE
1034 003046 104412          SETFLG ;SET FLAG BASED UPON INPUT STRING
1035 003050 001216          BAUDFLG ;THIS FLAG
1036 003052 105737 001216          11500 TSTB BAUDFLG ;DO YOU STILL WANT TO RUN BAUD TEST?
1037 003056 001412          11600 BEQ SWMP ;NO
1038 003060 104403 012535          11700 TYPE ,MSCOPBAUD ;YES,FIND OUT IF SWITCH 14 IS UP
1039 003064 000000          11800 HALT ;WAIT FOR OPR TO PUT UP SW 14
1040 003066 000137 003026          11900 JMP SW14.B
1041 003072 112777 000103 176230 12000 OUT.PT: MOVB #103,ADPTS ;TURN ON TRANSMITTER
1042 003100 000137 003026          12100 JMP SW14.B
1043 003104 000137 005700          12200 SWMP:  JMP MAINPR

```

```

1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059 003110 032737 000010 177570
1060 003116 001402
1061 003120 000137 003512
1062 003124
1063
1064
1065
1066
1067
1068
1069 003124 012737 000002 001072
1070 003132 012737 003512 001062
1071 003140 052777 000400 176212 14400
1072 003146 012777 000000 176202 14500
1073
1074 003154 052777 000400 176176 14700
1075 003162 012777 006026 176166 14800
1076
1077
1078 003170 042777 003010 176154 15100
1079 003176 012700 073016 15200
1080 003202 013737 001352 001110 15300
1081 003210 012737 040000 001106 15400
1082
1083 003216 052777 000016 176126 15500
1084 003224 005337 001106 15600
1085 003230 001002 15700
1086 003232 104002 15800
1087
1088 003234 000406 15900
1089 003236 017701 176110 16000
1090 003242 042701 104761 16100
1091 003246 020001 16200
1092 003250 001365 16300
1093
1094 003252 112777 000025 176102 16400
1095 003260 012777 003360 176076 16500
1096 003266 012777 000340 176072 16600
1097 003274 005037 177776 16700
1098 003300 013737 001354 001110 16800
1099 003306 012700 000025 16900

```

```

;:*****
;:DU11 TEST FOR THE DFC-11A
;:*****
;:THIS TEST VERIFYS THAT THE DFC-11A IS CLOCKING DATA
;:IN & OUT OF THE DU11 IT ALSO VERIFY THAT RING,CARRIER,
;:CLEAR TO SEND,DATA TERMINAL READY,AND REQUEST TO SEND
;:SIGNALS ARE WORKING CORRECTLY
;:NOTE: THE BCOSC MODEM CABLE MUST BE ATTACHED TO
;:THE DFC-11A AND MUST BE TERMINATED WITH H315 CONNECTOR
;:MODE: ISYMOD (ISOCRONOUS)
;:LENGTH: EIGHT
;:THIS TEST USES BOTH THE RECEIVER & TRANSMITTER LOGIC OF THE DU11
DU11.A: BIT #BIT3,SWR ;TEST BIT 3
        BEJ 64$
        JMP DU11.B ;JUMP AROUND
64$:
;*****
; TEST 2
;*****
;*****
TST2: MOV #2,TSTNO
      MOV #TST3,NEXT
      BIS #MRESET,@TXCSR ;MASTER RESET
      MOV #ISYMOD,@PARCSR ;LOAD THE MODE
      BIS #MRESET,@TXCSR ;MASTER RESET
      MOV #ISYMOD!EIGHT!NOPAR!26,@PARCSR ;LOAD THE MODE.
      ;# OF BITS PER CHAR,PARITY SENSE(NO PARITY),
      ;&SYNC CHARACTER (25)
      BIC #SRD!DSR!STD,@RXCSR
      MOV #RING!CTS!CARDET!SRD!DSR!STD!RTS!DTR,RO ;SET UP FOR ERROR MSG
      MOV RXCSR,TEMP4 ;DITTO
      MOV #40000,TEMP3 ;SET UP FOR CABLE DELAY + CTS DELAY
      ;OF .3 SEC MAX.
      BIS #DTR!RTS!STD,@RXCSR ;SET DTR & RTS & STD
1$: DEC TEMP3
   BNE 6$
   HLT 2 ;RING,CTS,CARDET,SRD,DSR,STD,RTS,OR DTR FAILED TO SET
      ;CHECK THE BCOSC CABLE
6$: BR 7$ ;GO RUN THE TEST ANYWAY !
   MOV @RXCSR,R1 ;SAVE ACTUAL
   BIC #<RING!CTS!CARDET!SRD!DSR!STD!RTS!DTR>,R1 ;STRIP JUNK
   CMP RO,R1 ;GOT THEM YET ?
   BNE 1$ ;NO NOT YET MAYBE THE BCOSC
      ;CABLE IS NOT ON OR TERMINATED WITH H315
7$: MOVB #25,@TXDBUF ;LOAD THE CHAR
   MOV #25,@DURIV ;SET UP TRAPCATCHER
   MOV #340,@DURIS ;SET LEVEL 7
   CLR PS ;ALLOW INTERRUPTS
   MOV RXDBUF,TEMP4 ;SET UP FOR ERROR MESSAGE
   MOV #25,RO ;EXPECTED

```



```

1148      00200
1149      00300
1150      00400
1151      00500
1152      00600
1153      00700
1154      00800
1155      00900
1156      003512 01000
1157
1158
1159
1160
1161
1162
1163      003512 012737 000003 001072
1164      003520 012737 007736 001062
1165      003526 052777 000400 175624 01200
1166      003534 012777 030000 175614 01300
1167      003542 052777 000400 175610 01400
1168      003550 012777 000020 175602 01500
1169      003556 012777 036026 175572 01600
1170
1171
1172      003564 052777 000436 175560 01800
1173
1174
1175
1176
1177
1178
1179
1180      003572 005037 001112
1181      003576 005002
1182      003600 005202
1183      003602 001376
1184      003604 005237 001112
1185      003610 022737 000003 001112
1186      003616 002367
1187      003620 012777 004022 175536
1188      003626 012777 000340 175532
1189      003634 012777 004100 175526
1190      003642 012777 000340 175522
1191      003650 005037 177776
1192      003654 013737 001354 001110
1193      003662 012700 000025
1194      003666 012737 000003 001106
1195
1196      003674 052777 000100 175450
1197      003702 052777 000100 175450
1198      003710 000137 003726
1199
1200      003714 112777 000026 175440
1201      003722 005037 001112
1202      003726 005002
1203      003730 005202
  
```

```

;: THIS TEST VERIFYS THAT THE DFC11A IS CLOCKING DATA IN & OUT
;: OF THE DU11
;: NOTE: THE BCOSC MODEM CABLE MUST BE ATTACHED TO THE DFC11A
;: & MUST BE TERMINATED WITH H315 CONNECTOR
;: MODE: SYNINT
;: LENGTH: EIGHT
;: THIS TEST USES BOTH THE RECEIVER & TRANSMITTER LOGIC
;: *****
DU11.B:
;: *****
;: *
;: TEST 3
;: *
;: *****
;: *****
TST3:  MOV    #3,TSTNO
      MOV    #.EOP,NEXT
      BIS    #MRESET,@TXCSR ;MASTER RESET
      MOV    #SYNINT,@PARCSR ;SET THE MODE
      BIS    #MRESET,@TXCSR ;MASTER RESET
      MOV    #USER!SEND,@TXCSR ;SET USER MODE AND SEND
      MOV    #SYNINT!EIGHT!NOPAR!26,@PARCSR ;SET SYNC INTERNAL,
      ;EIGHT BITS PER CHAR,NO PARITY,
      ;AND "26" FOR THE SYNC CHARACTER
      BIS    #SYNSCH!STPSYN!DTR!RTS!STD,@RXCSR ;SET SEARCH SYNC &
      ;STRIP SYNC SO THAT RXDONE ASSERTS
      ;WHEN CHAR "25" ARRIVES AND NOT BEFORE...
      ;THEREFORE,SET STRIP SYNC
      ;ALSO SET DTR, RTS,& STD FOR THE DFC11A

;:.....WAIT FOR SYNSCH TO BE
;:CLOCKED IN BY CLK
      CLR    TEMPS
      CLR    R2
      INC    R2 ;WAIT
      BNE    #-2
      INC    TEMPS
      CMP    #3,TEMPS
      BGE    #-20 ;GO BACK TO CLR R2 AND WAIT SOME MORE
      MOV    #25,@DURIV ;SET UP TRAPCATCHER
      MOV    #340,@DURIS ;SET PROCESSOR STATUS = 7
      MOV    #35,@DUTIV ;SET UP TRAPCATCHER
      MOV    #340,@DUTIS ;SET PROCESSOR STATUS = 7
      CLR    PS ;ALLOW INTERRUPTS
      MOV    RXDBUF,TEMP4 ;SET UP FOR ERROR MSG
      MOV    #25,R0 ;EXPECTED CHAR
      MOV    #3,TEMP3 ;# OF SYNC CHARS TO GET INTO
      ;SYNCRONIZATION
9$:   BIS    #RINTEN,@RXCSR ;SET INTERRUPT ENABLES
      BIS    #TXINTE,@TXCSR
      JMP    8$ ;THE FIRST XMIT INTERRUPT SHOULD COME
      ;FROM TXDONE = 1 AFTER A MASTER RESET
1$:   MOVB  #26,@TXDBUF ;LOAD SYNC CHAR
      CLR    TEMPS
8$:   CLR    R2 ;WAIT FOR INTERRUPT
      INC    R2
  
```

1204	003732	001376		05100	BNE	.-2	:CONTINUE TO WAIT
1205	003734	005237	001112	05200	INC	TEMPS	
1206	003740	022737	000003	05300	CMP	#3,TEMPS	:DONE YET?
1207	003746	002367		05400	BGE	8\$	:GO BACK,NOT YET
1208	003750	012737	000340	177776	MOV	#340,PS	:PREVENT INTERRUPTS
1209	003756	042777	000100	175374	BIC	#TXINTE,@TXCSR	:CLR INTR ENABLES
1210	003764	042777	000100	175360	BIC	#RINTEN,@RXCSR	:DITTO
1211	003772	013777	001366	175364	MOV	DURIS,@DURIV	:RESTORE TRAPCATCHER
1212	004000	005077	175362	05900	CLR	@DURIS	
1213	004004	013777	001372	175356	MOV	DUTIS,@DUTIV	:RESTORE INTERRUPT CATCHER
1214	004012	005077	175354	06100	CLR	@DUTIS	
1215	004016	104003		06200	HLT	3	:TXDONE INTERRUPT FAILED TO OCCUR
1216				06300			:WATCH OUT HERE::: THIS FAILURE MAY
1217				06400			:ALSO BE CAUSED BY TRANSMIT DATA NOT
1218				06500			:BEING CLOCKED OUT. I.E. TXDONE
1219				06600			:NOT RE-ASCERTING SO THAT THE 2ND
1220				06700			:SYNC CHARACTER CAN BE LOADED
1221	004020	000521		06800	BR	7\$	:GET OUT OF THE TEST
1222				06900			
1223				07000			:THE FOLLOWING IS THE RECEIVER INTERRUPT SVC ROUTINE
1224	004022			07100	2\$:		
1225	004022	017704	175324	07200	MOV	@RXCSR,R4	:SAVE
1226	004026	017701	175322	07300	MOV	@RXDBUF,R1	:ACTUAL
1227	004032	013777	001366	175324	MOV	DURIS,@DURIV	:RESTORE TRAPCATCHER
1228	004040	005077	175322	07500	CLR	@DURIS	:CLEAR OUT INTERRUPTS
1229	004044	013777	001372	175316	MOV	DUTIS,@DUTIV	
1230	004052	005077	175314	07700	CLR	@DUTIS	
1231	004056	012716	004230	07800	MOV	#4\$,(SP)	:SET UP RETURN LOCATION
1232	004062	042777	000100	175262	BIC	#RINTEN,@RXCSR	:CLR INTERRUPT ENABLES
1233	004070	042777	000100	175262	BIC	#TXINTE,@TXCSR	:DITTO
1234	004076	000002		08100	RTI		
1235				08200			:END OF RECEIVER INTERRUPT SVC ROUTINE
1236				08300			:...THE FOLLOWING IS THE XMITTER INTERRUPT SVC ROUTINE
1237	004100	005337	001106	08400	3\$:	DEC	TEMP3
1238	004104	100403		08500	BMI	5\$	:# OF SYNC CHARACTERS LEFT
1239	004106	012716	003714	08600	MOV	#1\$,(SP)	:SET UP RETURN LOCATION
1240				08700			: (LOAD SYNC CHARACTER AGAIN)
1241	004112	000002		08800	RTI		
1242	004114	012716	004122	08900	5\$:	MOV	#6\$,(SP)
1243	004120	000002		09000	RTI		:SET UP RETURN LOCATION
1244				09100			:END OF XMITTER INTERRUPT SVC ROUTINE
1245	004122	112777	000025	175232	6\$:	MOVB	#25,@TXDBUF
1246	004130	042777	000100	175222	BIC	#TXINTE,@TXCSR	:LOAD CHARACTER
1247	004136	005037	001112	09400	CLR	TEMPS	:CLR INTR ENABLE
1248	004142	005002		09500	10\$:	CLR	R2
1249	004144	005202		09600	INC	R2	:WAIT FOR INTERRUPT(RECEIVER)
1250	004146	001376		09700	BNE	.-2	:CONTINUE TO WAIT
1251	004150	005237	001112	09800	INC	TEMPS	
1252	004154	022737	000003	09900	CMP	#3,TEMPS	:DONE YET?
1253	004162	002367		10000	BGE	10\$	:GO BACK,NOT YET
1254	004164	012737	000340	177776	MOV	#340,PS	:PREVENT INTERRUPTS
1255	004172	042777	000100	175152	BIC	#RINTEN,@RXCSR	:CLR INTR ENABLE
1256	004200	013777	001366	175156	MOV	DURIS,@DURIV	:RESTORE TRAPCATCHER
1257	004206	005077	175154	10400	CLR	@DURIS	
1258	004212	013777	001372	175150	MOV	DUTIS,@DUTIV	
1259	004220	005077	175146	10600	CLR	@DUTIS	

1260	004	104004				10700	HLT	4	:RECEIVER INTR FAILED TO OCCUR
1261	004	000416				10800	BR	75	:GET OUT OF TEST
1262	004	020001				10900	CMP	RO,R1	
1263	004	001401				11000	BEQ	.+4	
1264	004	104002				11100	HLT	2	:CHARACTERS DID NOT MATCH
1265	004	013737	001352	001110		11200	MOV	RXC SR,TEMP4	:SET UP FOR ERROR MSG
1266	004	012700	000200			11300	MOV	RX DONE,RO	:EXPECTED RXDONE
1267	004	010401				11400	MOV	R4,R1	:ACTUAL
1268	004	042701	177577			11500	BIC	R1<RXDONE>,R1	:SAVE ONLY RXDONE
1269	004	020001				11600	CMP	RO,R1	
1270	004	001401				11700	BEQ	.+4	
1271	004	104006				11800	HLT	6	:FALSE INTERRUPT
1272	004	012737	000340	177776		11900	MOV	R340,PS	:INHIBIT INTERRUPTS
1273	004	04401				12000	SCOPE		

```

1275 02100
1276 02200
1277 02300
1278 02400
1279 02500
1280 02600
1281 02700
1282 02800
1283 02900
1284 03000
1285 03100
1286 03200
1287 03300
1288 03400
1289 03500
1290 03600
1291 03700
1292 03800
1293 03900
1294 04000
1295 04100
1296 004274 012737 000340 177776 04200
1297 004302 012706 001050 04300
1298 004306 000005 04400
1299 004310 005037 001222 04500
1300 004314 012737 177777 001226 04600
1301 004322 052777 000400 175030 04700
1302 004330 012777 026252 175020 04800
1303 004336 000240 04900
1304 004340 012777 000040 175012 05000
1305 004346 042777 000100 175004 05100
1306 004354 112777 000252 175000 05200
1307 004362 052777 000122 174762 05300
1308 004370 012777 005770 174766 05400
1309 004376 012777 000340 174762 05500
1310 004404 012777 006002 174756 05600
1311 004412 012777 000340 174752 05700
1312 05800
1313 004420 004737 005456 05900
1314 004424 005737 001212 06000
1315 004430 001002 06100
1316 004432 000137 005634 06200
1317 004436 06300
1318 004436 104410
1319 004440 012600
1320 004442 104412
1321 004444 001214
1322 004446 105737 001214 06400
1323 004452 001500 06500
1324 004454 012737 000454 001224 06600
1325 004462 022737 177546 001212 06700
1326 004470 001402 06800
1327 004472 000137 005572 06900
1328 004476 012737 004552 000100 07000
1329 004504 012737 000340 000102 07100
1330 004512 012737 177546 001230 07200

```

```

:*****
:
: BAUD RATE TEST
:*****
: THIS TEST DETERMINES THE BAUD RATE OF THE DFC USING
: A KW11-L/P AND DU/DP-11 AND PRINTS OUT THAT RATE. THE
: TEST ALSO GIVES AN ERROR TYPEOUT IF THE BAUD RATE IS
: OFF BY MORE THAN 10%.
: MODE: IDLE SYNC
: LENGTH: EIGHT
: THIS TEST USES BOTH TRANSMITTER AND RECEIVER LOGIC.
: A SCOPE LOOP IS AVAILABLE BY USING SWITCH 14.
: MAKE SURE THE PROPER CONNECTOR IS INSTALLED!!!!!! SEE
: SECTION 1.0!!!!!!

START:  MOV    #340,PS      ;PREVENT INTERRUPTS
        MOV    #STACK,SP   ;SET UP STACK
TXSETUP: RESET
        CLR    CHCNT
        MOV    #-1,INTCNT
        BIS    #MRESET,@TXCSR ;MASTER RESET TRANSMITTER
        MOV    #026252,@PARCSR ;LOAD THE MODE,WORD LENGTH
        NOP    ;PARITY SENSE AND SYNC CHARACTER
        MOV    #40,@TXCSR    ;ASSERT DRA INTR
        BIC    #100,@TXCSR  ;CLR TXINTEB
        MOV    #252,@TXDBUF ;LOAD SYNC CHAR INTO BUFFER
        MOVB  #122,@RXCSR   ;SET UP RECEIVER
RXSETUP: BIS    #CNT,RX,@URIV ;SET
DUSETUP: MOV    #340,@URIS  ;
        MOV    #NO.TRAP,@OUTIV ;
        MOV    #340,@OUTIS  ;          UP
                                ;          FOR DU
                                ;          INTERRUPTS

        JSR    PC,CLOCK     ;GO CHECK FOR CLOCK
        TST    NO.CLOCK    ;IS IT THERE?
        BNE    IS
        JMP    SW14A       ;NO,CHECK IF SCOPE LOOP WANTED

IS:
        INTR  ;OUTPUT MESSAGE & GET INPJT STRING
        MCPS ;MESSAGE
        SETFLG ;SET FLAG BASED UPON INPLT STRING
        CPSFLG ;THIS FLAG
        TSTB  CPSFLG      ;TEST FOR 50/60 CYCLES
        BEQ  NO60CPS     ;50 CYCLE MACHINE
        MOV  #300,CLKSET ;SETUP FOR 60 CYCLE COUNT
KWSETUP: CMP  #177546,NO.CLOCK ;FIND OUT WHICH CLOCK
        BEQ  KW11L      ;KW11L SETUP
        JMP  KW11P      ;KW11P SETUP
KW11L:  MOV  #INTSVC,@#100 ;SET UP SERVICE ROUTINE
        MOV  #340,@#102   ;SET UP LEVEL 7
        MOV  #177546,CLKSTATUS ;KW11L CSR

```

```

1331 004520 012777 000100 174502 07300      MOV      #100,CLKSTATUS ;SETUP FOR CLOCK INTERRUPT
1332 004526 005037 177776 07400  BEGIN:  CLR      PS          ;ALLOW INTERRUPTS
1333 004532 005000 07500      CLR      RO
1334 004534 012701 000005 07600      MOV      #5,R1          ;WAIT FOR INTERRUPTS
1335 004540 005300 07700  1$:      DEC      RO          ;SO THAT WE CAN
1336 004542 001376 07800      BNE     1$          ;SYNC UP KW AND DEVICE
1337 004544 005301 07900      DEC     R1
1338 004546 001374 09000      BNE     1$
1339 004550 104000 08100      HLT
1340 004552 042777 000200 174450 08200  INTSVC: BIC     #200,CLKSTATUS ;TOOK TOO LONG---TIME OUT!!!
1341 004560 005237 00122E 08300      INC     INTCNT        ;CLEAR THE MONITOR
1342 004564 001015 08400      BNE     EEE          ;COUNT INTERRUPTS
1343 004566 005737 001234 08500      TST     TAG          ;FIRST TIME?
1344 004572 001404 08600      BEQ     INTS.A       ;DP OR DU?
1345 004574 052777 000103 174526 08700      BIS     #103,DPPTS   ;DU
1346 004602 000403 08800      BR      INTS.B       ;YES, START OUTPUT OF DP
1347 004604 112777 000020 174546 08900  INTS.A: MOVB   #20,DTXCSR  ;KEEP GOING WITH TEST
1348 004612 012716 004526 09000  INTS.B: MC/   #BEGIN,(SP) ;YES, START OUTPUT OF DU
1349 004616 000002 09100      RTI
1350 004620 023737 001224 001226 09200  EEE:    CMP     CLKSET,INTCNT ;ARE WE DONE?
1351 004626 001403 09300      BEQ     2$          ;YES
1352 004630 012716 004526 09400      MOV     #BEGIN,(SP) ;NO, CONTINUE TO TEST
1353 004634 000002 09500      RTI
1354 004636 042777 000101 174364 09600  2$:    BIC     #101,CLKSTATUS ;SHUT OFF CLOCK
1355 004644 012716 004666 09700      MOV     #BDTAG,(SP) ;GO DETERMINE BAUD RATE
1356 004650 000005 09800      RESET
1357 004652 000002 09900      RTI
1358 004654 012737 000372 001224 10000  NO60CPS:MOV  #250,CLKSET ;50 CYCLE SETUP
1359 004662 000137 004462 10100      JMP     KWSETUP
1360 004666 010500 10500  BDTAG:
1361 004666 012737 013112 004732  MOV     #M19200,17$
1362 004674 023727 001222 031620  CMP     CHCNT,#13200. ;COMPARE TO THIS HIGH END
1363 004702 101404 013205  BLOS   .+12          ;ANSWER IS TOO HIGH
1364 004704 104403 005634  JMP     MWRONGRT
1365 004710 000137 005634  JMP     $W14A
1366 004714 023727 001222 025060  CMP     CHCNT,#10800. ;COMPARE TO THIS LOW END
1367 004722 103410 013306  BLO    18$          ;NOT HERE
1368 004724 104403 013306  TYPE   .MBAUDRT    ;TYPE THIS RATE
1369 004730 104403
1370 004732 000000 17$:    0
1371 004734 104403 013334  TYPE   ,MRITE
1372 004740 000137 005700  JMP     MAINPR
1373 004744
1374 004744 012737 013122 005010 18$:    MOV     #M9600,1$
1375 004752 023727 001222 014710  CMP     CHCNT,#6600. ;COMPARE TO THIS HIGH END
1376 004760 101404 013205  BLOS   .+12          ;ANSWER IS TOO HIGH
1377 004762 104403 005634  JMP     MWRONGRT
1378 004766 000137 005634  JMP     $W14A
1379 004772 023727 001222 012430  CMP     CHCNT,#5400. ;COMPARE TO THIS LOW END
1380 005000 103410 013306  BLO    2$          ;NOT HERE
1381 005002 104403 013306  TYPE   .MBAUDRT    ;TYPE THIS RATE
1382 005006 104403
1383 005010 000000 1$:    0
1384 005012 104403 013334  TYPE   ,MRITE
1385 005016 000137 005700  JMP     MAINPR
1386 005022 2$:

```

# E03

1387	005022	012737	013131	005066	MOV	#M4800,3\$	
1388	005030	023727	001222	006344	CMP	CHCNT,#3300.	;COMPARE TO THIS HIGH END
1389	005036	101404			BLOS	.+12	;ANSWER IS TOO HIGH
1390	005040	104403	013205		TYPE	,MWRONGRT	
1391	005044	000137	005634		JMP	\$W14A	
1392	005050	023727	001222	005214	CMP	CHCNT,#2700.	;COMPARE TO THIS LOW END
1393	005056	103410			BLO	4\$	;NOT HERE
1394	005060	104403	013306		TYPE	,MBAUDRT	;TYPE THIS RATE
1395	005064	104403			TYPE		
1396	005066	000000			0		
1397	005070	104403	013334		TYPE	,WRITE	
1398	005074	000137	005700		JMP	MAINPR	
1399	005100						
1400	005100	012737	013140	005144	MOV	#M2400,5\$	
1401	005105	023727	001222	003162	CMP	CHCNT,#1650.	;COMPARE TO THIS HIGH END
1402	005114	101404			BLOS	.+12	;ANSWER IS TOO HIGH
1403	005116	104403	013205		TYPE	,MWRONGRT	
1404	005122	000137	005634		JMP	\$W14A	
1405	005126	023727	001222	002506	CMP	CHCNT,#1350.	;COMPARE TO THIS LOW END
1406	005134	103410			BLO	6\$	;NOT HERE
1407	005136	104403	013306		TYPE	,MBAUDRT	;TYPE THIS RATE
1408	005142	104403			TYPE		
1409	005144	000000			0		
1410	005146	104403	013334		TYPE	,WRITE	
1411	005152	000137	005700		JMP	MAINPR	
1412	005156						
1413	005156	012737	013147	005222	MOV	#M1200,7\$	
1414	005164	023727	001222	001471	CMP	CHCNT,#825.	;COMPARE TO THIS HIGH END
1415	005172	101404			BLOS	.+12	;ANSWER IS TOO HIGH
1416	005174	104403	013205		TYPE	,MWRONGRT	
1417	005200	000137	005634		JMP	\$W14A	
1418	005204	023727	001222	001243	CMP	CHCNT,#675.	;COMPARE TO THIS LOW END
1419	005212	103410			BLO	8\$	;NOT HERE
1420	005214	104403	013306		TYPE	,MBAUDRT	;TYPE THIS RATE
1421	005220	104403			TYPE		
1422	005222	000000			0		
1423	005224	104403	013334		TYPE	,WRITE	
1424	005230	000137	005700		JMP	MAINPR	
1425	005234						
1426	005234	012737	013156	005300	MOV	#M600,9\$	
1427	005242	023727	001222	000635	CMP	CHCNT,#413.	;COMPARE TO THIS HIGH END
1428	005250	101404			BLOS	.+12	;ANSWER IS TOO HIGH
1429	005252	104403	013205		TYPE	,MWRONGRT	
1430	005256	000137	005634		JMP	\$W14A	
1431	005262	023727	001222	000521	CMP	CHCNT,#337.	;COMPARE TO THIS LOW END
1432	005270	103410			BLO	10\$	;NOT HERE
1433	005272	104403	013306		TYPE	,MBAUDRT	;TYPE THIS RATE
1434	005276	104403			TYPE		
1435	005300	000000			0		
1436	005302	104403	013334		TYPE	,WRITE	
1437	005306	000137	005700		JMP	MAINPR	
1438	005312						
1439	005312	012737	013164	005356	MOV	#M300,11\$	
1440	005320	023727	001222	000317	CMP	CHCNT,#207.	;COMPARE TO THIS HIGH END
1441	005326	101404			BLOS	.+12	;ANSWER IS TOO HIGH
1442	005330	104403	013205		TYPE	,MWRONGRT	

F03

1443	005334	000137	005634			JMP	SW14A	
1444	005340	023727	001222	000251		CMP	CHCNT, #169.	;COMPARE TO THIS LOW END
1445	005346	103410				BLO	12\$	;NOT HERE
1446	005350	104403	013306			TYPE	,MBAUDRT	;TYPE THIS RATE
1447	005354	104403				TYPE		
1448	005356	000000			11\$:	0		
1449	005360	104403	013334			TYPE	,WRITE	
1450	005364	000137	005700			JMP	MAINPR	
1451	005370				12\$:			
1452	005370	012737	013172	005434		MOV	#M150, 13\$	
1453	005376	023727	001222	000147		CMP	CHCNT, #103.	;COMPARE TO THIS HIGH END
1454	005404	101404				BLOS	.+12	;ANSWER IS TOO HIGH
1455	005406	104403	013205			TYPE	,MWRONGRT	
1456	005412	000137	005634			JMP	SW14A	
1457	005416	023727	001222	000125		CMP	CHCNT, #85.	;COMPARE TO THIS LOW END
1458	005424	103410				BLO	14\$	;NOT HERE
1459	005426	104403	013306			TYPE	,MBAUDRT	;TYPE THIS RATE
1460	005432	104403				TYPE		
1461	005434	000000			13\$:	0		
1462	005436	104403	013334			TYPE	,WRITE	
1463	005442	000137	005700			JMP	MAINPR	
1464	005446				14\$:			
1465	005446	104403	013205			TYPE	,MWRONGRT	;EITHER DEVICE DIDN'T TURN ON
1466				11400				;OR COUNT WAS OFF BY MORE THAN 10 PERCENT
1467	005452	000137	005634			JMP	SW14A	
1468				11500				
1469				11600				
1470				11700				
1471				11800				
1472				11900				
1473				12000				
1474	005456	013746	000004					
1475	005462	013746	000006					
1476	005466	012737	005524	000004				
1477	005474	012737	000340	000006				
1478	005502	005037	001212					
1479	005506	005737	177546					
1480	005512	000240						
1481	005514	012737	177546	001212				
1482	005522	000416						
1483	005524	012737	005552	000004				
1484	005532	022626						
1485	005534	005737	172540					
1486	005540	000240						
1487	005542	012737	172540	001212				
1488	005550	000403						
1489	005552	104403	012506					
1490	005556	022626						
1491	005560	012637	000006					
1492	005564	012637	000004					
1493	005570	000207						
1494	005572	012737	004552	000104				
1495	005600	012737	000340	000106				
1496	005606	012737	177777	172542				
1497	005614	012737	172540	001230				
1498	005622	012777	000135	173400				

  

```

;ROUTINE TO FIND OUT IF EITHER KW11-P OR
;KW11-L EXIST. ENTERED BY "JSR PC,CLOCK"
;*****
CLOCK:  MOV    2#4, -(SP)      ;SETUP FOR INTERRUPTS
        MOV    2#6, -(SP)      ;IN THIS ROUTINE
        MOV    2$ 2#4          ;SETUP TO SEE IF KW11P
        MOV    #340, 2#6       ;PREVENT INTERRUPTS
        CLR    NO.CLOCK
1$:     TST    2#177546        ;KW11-L CSR
        NOP
        MOV    #177546, NO.CLOCK;THIS IS THE ONE
        BR    3$
2$:     MOV    #4$, 2#4        ;SETUP TO CHECK KW11P
        CMP    (SP)+, (SP)+    ;POP STACK
        TST    2#172540        ;KW11-P CSR
        NOP
        MOV    #172540, NO.CLOCK;THIS IS THE ONE
        BR    3$
4$:     TYPE   MNOCLOCK        ;THERE IS NO CLOCK
        CMP    (SP)+, (SP)+    ;POP STACK
3$:     MOV    (SP)+, 2#6      ;RETURN ADDRESSES 4 AND 6 TO NORMAL
        MOV    (SP)+, 2#4      ;DITTO
        RTS    PC
KW11P:  MOV    #INTSVC, 2#104   ;SETUP FOR INTSVC RTN
        MOV    #340, 2#106     ;PREVENT INTERRUPTS
        MOV    #177777, 2#172542 ;PRESET COUNT SET BUFFER
        MOV    #172540, CLKSTATUS ;KW11P CSR
        MOV    #135, 2CLKSTATUS ;SETUP INTER EN AND STATE
  
```



# H03

DZDFA-B MACY11 27(732) 17-SEP-76 16:06 PAGE 34  
 DZDFAB.P11

1541				00100		
1542				00200	;LINE.N SUBROUTINE TO FETCH THE LINE	
1543				00300	;NUMBER AND FIRST DP11 VECTOR ADDRESS FROM	
1544				00400	;THE CONSOL SWITCHES	
1545				00500	;SW0-SW8=VECTOR ADDRESS OF FIRST DP11	
1546				00600	;SW9-SW15=LINE NUMBER OF DP11 SELECTED FOR TEST	
1547				00700		
1548	006014	005037	001076	00800	LINE.N: CLR	ERRCNT ;CLEAR ERROR COUNT
1549	006020	013737	177570	00900	MOV	SWR,SAVSR1 ;SAVE CONSOL SWITCH SETTINGS
1550	006026	013700	177570	01000	MOV	SWR,RO
1551	006032	000000		01100	HALT	
1552				01200		;SET SWR TO LINE NUMBER
1553				01300		;LOW BYTE = FIRST DP VECTOR
1554	006034	013737	177570	01400	MOV	SWR,SAVSR2 ;HIGH BYTE = LINE NUMBER(8)
1555	006042	013700	177570	01500	MOV	SWR,RO ;SAVE CONSOL SWITCHES
1556	006046	000000		01600	HALT	
1557	006050	005001		01700	CLR	R1
1558	006052	113701	001175	01800	MOVB	SAVSR2+1,R1 ;SAVE LINE NUMBER
1559	006056	042701	000001	01900	BIC	#1,R1
1560	006062	000405		02000	BR	XTAG
1561	006064	013701	001156	02100	LINE.X: MOV	XLINEX,R1
1562	006070	005037	001144	02200	CLR	SAVSR1
1563	006074	006301		02300	ASL	R1
1564	006076	006201		02400	XTAG: ASR	R1 ;CLEAR LSB
1565	006100	010137	001156	02500	MOV	R1,XLINEX
1566	006104	006301		02600	ASL	R1 ;SCALE LINE NUMBER TO ADDRESS
1567	006106	006301		02700	ASL	R1 ;MODULO 10(8)
1568	006110	006301		02800	ASL	R1
1569	006112	005777	173232	02900	TST	QBASCSR ;DOES LINE 0 REALLY EXIST??
1570	006116	000240		03000	NOP	
1571	006120	105737	001144	03100	TSTB	SAVSR1
1572	006124	100403		03200	BMI	.+10
1573	006126	012737	006240	03300	MOV	#3\$,Q#4
1574	006134	013702	001350	03400	MOV	BASCSR,R2 ;SET R2 = LINE 0 ADDRESS
1575	006140	160102		03500	SUB	R1,R2 ;MANUFACTURE DEVICE ADDRESS
1576	006142	012703	001322	03600	MOV	#DPRS,R3 ;R3 = ADDRESS OF RCY STATUS ADRS
1577	006146	010223		03700	MOV	R2,(R3)+ ;LOAD RCY STATUS ADRS
1578	006150	005722		03800	TST	(R2)+ ;INC TO RCY BUFFER ADRS
1579	006152	010223		03900	MOV	R2,(R3)+ ;LOAD RCY BUFFER ADRS
1580	006154	005202		04000	INC	R2 ;INC TO SYNC BUFFER ADRS
1581	006156	010223		04100	MOV	R2,(R3)+ ;LOAD SYNC ADRS
1582	006160	005202		04200	INC	R2 ;INC TO XMIT STATUS ADRS
1583	006162	010223		04300	MOV	R2,(R3)+ ;LOAD TRANSMITTER STATUS ADRS
1584	006164	005722		04400	TST	(R2)+ ;INC TO XMIT BUFFER
1585	006166	010223		04500	MOV	R2,(R3)+ ;LOAD XMIT BUFFER ADRS
1586	006170	005202		04600	INC	R2 ;INC TO SYNC EXTENSION
1587	006172	010223		04700	MOV	R2,(R3)+ ;LOAD SYNC EXTENSION ADRS
1588	006174	013702	001174	04800	MOV	SAVSR2,R2 ;SET UP VECTOR ADDRESS
1589	006200	042702	177000	04900	BIC	#1C<???,R2 ;CLEAR LINE NUMBER FROM VEC ADRS
1590	006204	105737	001144	05000	TSTB	SAVSR1
1591	006210	100402		05100	BMI	2\$
1592	006212	013702	001346	05200	MOV	BASVEC,R2
1593	006216	060102		05300	2\$: ADD	R1,R2 ;SET VECTOR ADDRESS TO LINE NUMBER
1594	006220	010223		05400	MOV	R2,(R3)+ ;LOAD RCY VECTOR ADRS
1595	006222	005722		05500	TST	(R2)+ ;INC TO NEXT VECTOR
1596	006224	010223		05600	MOV	R2,(R3)+ ;LOAD RCY PRIORITY ADRS

1597	006226	005722		05700
1598	006230	010223		05800
1599	006232	005722		05900
1600	006234	010213		06000
1601	006236	000414		06100
1602	006240	005037	001156	06200
1603	006244	013701	000042	06300
1604	006250	001405		06400
1605	006252	000005		06500
1606				06600
1607				06700
1608				06800
1609	006254	004711		06900
1610	006256	000240		07000
1611	006260	000240		07100
1612	006262	000240		07200
1613	006264	022626		07300
1614	006266	000676		07400
1615	006270	012737	000006 000004	07500
1616	006276	013737	001322 001210	07600
1617	006304	000207		07700
1618				07800
1619				07900
1620				08000
1621				08100
1622	006306	012702	000300	08200
1623	006312	012701	000302	08300
1624	006316	010122		08400
1625	006320	005022		08500
1626	006322	022121		08600
1627	006324	022701	000776	08700
1628	006330	001372		08800
1629	006332	000207		08900
1630				09000
1631				09100
1632				09200
1633				09300
1634				09400
1635				09500
1636	006334	013746	000004	09600
1637	006340	013746	000006	09700
1638	006344	012737	006370 000004	09800
1639	006352	012737	000340 000006	09900
1640	006360	005777	000704	10000
1641	006364	000240		10100
1642	006366	000405		10200
1643	006370	104403	012773	10300
1644	006374	022626		10400
1645	006376	000000		10500
1646	006400	000777		10600
1647				10700
1648	006402	012637	000006	10800
1649	006406	012637	000004	10900
1650	006412	000207		11000
1651				11100
1652				11200

```

TST      (R2)+          ;INC TO NEXT VECTOR
MOV      R2,(R3)+       ;LOAD XMIT VECTOR ADRS
TST      (R2)+          ;INC TO NEXT VECTOR
MOV      R2,(R3)        ;LOAD XMIT PRIORITY ADRS
BR       5$
3$: CLR   XLINEX
MOV      @#42,R1
BEQ     4$
RESET
;*****
;DP11 LOGICAL ENDING
;*****
JSR      PC,(R1)
NOP
NOP
NOP
4$: POP  .SP
BR       LINE.X
5$: MOV  #6,@#4
MOV      DPRS.DEVICE    ;SET FOR END PASS MSG.
RTS      PC
;*****CLRVEC*****
;CLRVEC,ROUTINE TO FILL COMMUNICATION VECTOR AREA WITH .+2.HALT
CLRVEC: MOV  #300,R2      ;R2 COMM VECTOR AREA ADRS
MOV      #302,R1        ;INIT R1 WITH ADRS OF HALT
1$: MOV  R1,(R2)+       ;MOV .+2 TO PC
CLR      (R2)+          ;MOV HALT TO PC
CMP      (R1)+,(R1)+    ;INC TO NEXT VECTOR AREA
CMP      #776,R1        ;END OF VECTOR AREA
BNE     1$              ;NO
RTS      PC             ;RETURN
;*****OKADR*****
;OKADR, ROUTINE TO CHECK FOR VALID DU ADDRESS
OKADR:  MOV  @#4,-(SP)    ;SETUP FOR INTERRUPTS
MOV      @#6,-(SP)      ;DITTO
MOV      #1$,@#4        ;SETUP TO CHECK ADDR YOU SELECTED
MOV      #340,@#6       ;PREVENT INTERRUPTS
TST      @DUBASE        ;ADDRESS
NOP
BR       2$             ;WAIT
BR       2$             ;ADDRESS OK
1$: TYPE MNODEV         ;ADDRESS NG--LET OPR KNOW
CMP      (SP)+,(SP)+    ;POP STACK
HALT
BR       .              ;WAIT FOR OPR TO DECIDE IF ADR IS OK
2$: MOV  (SP)+,@#6      ;IF ADRS WAS OK THEN RUN DU DIAGNOSTIC
MOV      (SP)+,@#4      ;IF ADRS WRONG RESTART TEST AND ANS QUESTIONS
RTS      PC             ;RETURN ADDRESS 4 AND 6 TO NORMAL
;DITTO

```

1653					11300			
1654					11400			
1655					11500			
1656					11600			;POWER FAIL ROUTINE
1657					11700			
1658	006414	012737	006424	000024	11800	.PFAIL:	MOV #PWRUP,24	;LOAD PFAIL VECTOR FOR POWER UP
1659	006422	000000			11900		HALT	;WAIT TTY TO COME UP
1660	006424	000005			12000	PWRUP:	RESET	;REINIT STACK POINTER
1661	006426	012706	001050		12100		MOV #STACK,SP	;LOAD PFAIL VECTOR FOR POWER DOWN
1662	006432	012737	006414	000024	12200		MOV #.PFAIL,24	
1663	006440	104403			12300		TYPE	
1664	006442	011372			12400		MPOWER	
1665	006444	000177	172566		12500		JMP @PSTART	
1666					12600			
1667					12700			
1668	006450				12800	SEQ.DATA:		
1669	006450	012737	006662	001204	12900		MOV #6\$,BACK	
1670	006456	105077	172652		13000		CLRB @SEXT	;CLEAR SYNC EXTENTION
1671	006462	005037	001164		13100		CLR RDATA	;RECEIVER DATA
1672	006466	005037	001162		13200		CLR TDATA	;TRANSMITTER DATA
1673	006472	005077	172624		13300		CLR @DPRS	;RECEIVER STATUS
1674	006476	005077	172626		13400		CLR @DPTS	;TRANSMITTER STATUS
1675	006502	052777	000001	172612	13500		BIS #BIT0,@DPRS	;STRIP SYNC
1676	006510	012737	000400	001166	13600		MOV #400,@CHLEN	;CHAR LENGTH INDEX
1677	006516	032737	000400	001144	13700		BIT #BIT8,SAVSR1	;TEST 12 BIT CHAR MODE
1678	006524	001414			13800		BFB 1\$	;NO
1679	006526	012737	010000	001170	13900		MOV #1000,LIMIT	;SELECT END OF DATA
1680	006534	052777	002000	172560	14000		BIS #BIT10,@DPRS	;SELECT 12 BITS/CHARACTER
1681	006542	012737	000426	001154	14100		MOV #426,TSYNC	;SYNC FOR 12 BIT CHAR
1682	006550	105277	172560		14200		INCB @SEXT	;PLACE MSB OF SYNC IN SYNC EXT
1683	006554	000406			14300		BR 2\$	
1684	006556	012737	000400	001170	14400	1\$:	MOV #400,LIMIT	;TEMPORARY CHARACTER LIMIT
1685	006564	012737	000026	001154	14500		MOV #26,TSYNC	;INIT SYNC STORAGE
1686	006572	012777	006664	172542	14600	2\$:	MOV #TV18,@DPTIV	;TRANSMITTER VECTOR
1687	006600	012777	006772	172530	14700		MOV #RV18,@DPRIV	;RECEIVER VECTOR
1688	006606	012737	000200	177776	14800		MOV #200,PS	;PRIORITY=4
1689	006614	012737	000004	001172	14900		MOV #4,SCNT	;SYNC COUNT=4
1690	006622	113777	001154	172476	15000		MOVBS TSYNC,@SYNC	;LOAD SYNC
1691	006630	052777	000100	172464	15100		BIS #BIT6,@DPRS	;RCV INT ENB
1692	006636	052777	000301	172464	15200		BIS #301,@DPTS	
1693					15300			;TRANS INT ENB
1694					15400			;TRANS DONE
1695	006644	005237	001102		15500	5\$:	INC TEMP1	
1696	006650	001375			15600		BNE 5\$	
1697	006652	005337	001104		15700		DEC TEMP2	
1698	006656	001372			15800		BNE 5\$	
1699	006660	104000			15900		HLT	
1700	006662	000207			16000	6\$:	RTS PC	
1701					16100			
1702					16200			
1703					16300			
1704	006664	013777	001154	172440	16400	TV18:	MOV TSYNC,@DPTB	;XMIT SYNC
1705	006672	113777	001155	172434	16500		MOVBS TSYNC+1,@SEXT	;LOAD SYNC EXT
1706	006700	005337	001172		16600		DEC SCNT	;HAVE 2 SYNC'S BEEN XMITED
1707	006704	001003			16700		BNE 1\$	;NO
1708	006706	012777	006716	172426	16800		MOV #TV19,@DPTIV	;YES CHANGE VECTOR

# K03

DZDFA-B MACY11 27(732) 17-SEP-75 16:06 PAGE 37  
 DZDFAB.P11

1709	006714	000002			16900	1\$: RTI	
1710					17000		
1711					17100	;SEQUENTIAL DATA TRANSMISSION ROUTINE	
1712					17200		
1713					17300		
1714	006716	032777	140000	172404	17400	TV19: BIT #140000,ADPTS	;ANY STATUS ERRORS
1715	006724	001401			17500	BEQ .+4	;NO
1716	006726	104000			17600	HLT	;REPORT ERROR
1717	006730	105777	172374		17700	TSTB ADPTS	;TRANSMITTER READY
1718	006734	100401			17800	BMI .+4	;YES
1719	006736	104000			17900	HLT	;REPORT ERROR
1720	006740	013777	001162	172364	18000	MOV TDATA,ADPTB	;LOAD BUFFER
1721	006746	005237	001162		18100	INC TDATA	;NEXT CHARACTER
1722	006752	023737	001170	001162	18200	CMP LIMIT,TDATA	;HAVE ALL CHARACTERS
1723	006760	001003			18300	BNE 1\$	;NO
1724	006762	042777	000140	172340	18400	BIC #140,ADPTS	;YES, CLEAR INTERRUPTS
1725	006770	000002			18500	1\$: RTI	
1726					18600		
1727					18700	;RECEIVE SEQUENTIAL DATA	
1728					18800		
1729	006772	105777	172324		18900	RV18: TSTB ADPRS	;RECEIVER DONE??
1730	006776	100401			19000	BMI .+4	;YES
1731	007000	104000			19100	HLT	;REPORT ERROR
1732	007002	013700	001164		19200	MOV RDATA,RC	
1733	007006	017701	172312		19300	MOV ADPRB,RI	
1734	007012	023777	001164	172304	19400	CMP RDATA,ADPRB	;CORRECT DATA
1735	007020	001404			19500	BEQ 1\$	
1736	007022	017737	172276	001146	19600	MOV ADPRB,TMPDAT	;STORE DATA
1737	007030	104001			19700	HLT 1	;REPORT ERROR
1738	007032	042777	000001	172262	19800	1\$: BIC #BIT0,ADPRS	;CLEAR STRIP SYNC
1739	007040	005237	001164		19900	INC RDATA	;NEXT CHARACTER
1740	007044	023737	001170	001164	20000	CMP LIMIT,RDATA	
1741	007052	001037			20100	BNE 3\$	
1742	007054	005037	001164		20200	CLR RDATA	
1743	007060	005037	001162		20300	CLR TDATA	
1744	007064	006237	001170		20400	ASR LIMIT	;DECREASE LIMIT TO 7 BITS
1745	007070	012777	006664	172244	20500	MOV #TV18,ADPTIV	;SET UP SYNC TRANSMISSION
1746	007076	012737	000004	001172	20600	MOV #4,SCNT	;SYNC COUNT =4
1747	007104	052777	000001	172210	20700	BIS #BIT0,ADPRS	;STRIP SYNC
1748	007112	042777	004000	172202	20800	BIC #BIT11,ADPRS	;CLEAR RCV ACTIVE
1749	007120	052777	000301	172202	20900	BIS #301,ADPTS	
1750	007126	053777	001166	172166	21000	BIS CHLEN,ADPRS	;CHANGE CHAR LENGTH
1751	007134	062737	000400	001166	21100	ADD #400,CHLEN	;DECREASE CHAR LENGTH
1752	007142	022737	001400	001166	21200	2\$: CMP #1400,CHLEN	;HAVE ALL LENGTHS BEEN TESTED
1753	007150	001401			21300	BEQ 4\$	;YES
1754	007152	000002			21400	3\$: RTI	;NO
1755	007154	005077	172150		21500	4\$: CLR ADPTS	;CLR TRANSMITTER STATUS
1756	007160	005077	172136		21600	CLR ADPRS	;CLR RECEIVER STATUS
1757	007164	042737	000040	177776	21700	BIC #BITS,PS	;PRIORITY = 4
1758	007172	013716	001204		21800	MOV BACK,(SP)	
1759	007176	000002			21900	RTI	
1760					22000	;NFW DU ADDRESSES	
1761	007200	013737	007270	001352	22100	DUADDR: MOV DUBASE,RXCSR	;XXX0
1762	007206	013737	001352	001210	22200	MOV RXCSR,DEVICE	
1763	007214	062737	000002	007270	22300	ADD #2,DUBASE	
1764	007222	013737	007270	001354	22400	MOV DUBASE,RXDBUF	;XXX2

1765	007230	013737	007270	001356	22500
1766	007236	062737	000002	007270	22600
1767	007244	013737	007270	001360	22700
1768	007252	062737	000002	007270	22800
1769	007260	013737	007270	001362	22900
1770	007266	000207			23000
1771	007270	000000			23100
1772					23200
1773					23300
1774					23400
1775					23500
1776					23600
1777	007272	105737	001250		23700
1778	007276	001510			23800
1779	007300	005737	001264		23900
1780	007304	001007			24000
1781	007306	104403	012113		24100
1782	007312	013700	001264		24200
1783	007316	000000			24300
1784					24400
1785	007320	000137	001642		24500
1786	007324	062737	000010	001252	24600
1787	007332	062737	000010	001260	24700
1788	007340	000241			24800
1789	007342	006137	001266		24900
1790	007346	103412			25000
1791					25100
1792	007350	033737	001266	001264	25200
1793	007356	001762			25300
1794	007360	004737	007424		25400
1795	007364	004737	006334		25500
1796	007370	000137	007540		25600
1797	007374	012737	000001	001266	25700
1798					25800
1799	007402	013737	001254	001252	25900
1800	007410	013737	001262	001260	26000
1801	007416	004737	007424		26100
1802	007422	000436			26200
1803	007424	013737	001252	007270	26300
1804	007432	004737	007200		26400
1805	007436	013737	001260	001364	26500
1806	007444	062737	000002	001260	26600
1807	007452	013737	001260	001366	26700
1808	007460	062737	000002	001260	26900
1809	007466	012737	001260	001370	26900
1810	007474	062737	000002	001260	27000
1811	007502	013737	001260	001372	27100
1812	007510	013737	001364	001260	27200
1813	007516	000207			27300
1814					27400
1815					27500
1816	007520				27600
1817	007520	013701	000042		27700
1818	007524	001405			27800
1819	007526	000005			27900
1820	007530	004711			28000

```

MOV DUBASE,PARCSR ;XXX2
ADD #2,DUBASE
MOV DUBASE,TXCSR ;XXX4
ADD #2,DUBASE
MOV DUBASE,TXDBUF ;XXX6
RTS PC
DUBASE: 0

;THE FOLLOWING CALCULATES THE NEXT DEVICE ADDRESS AND ASSOCIATED
;VECTOR ADDRESSES IF RUNNING MULTIPLE DEVICES (DU11...DFC11'S)

DU.UPDATE: TSTB MULTD ;ARE YOU RUNNING MULTIPLE DEVICES ?
BEQ CCC ;NO JUMP AROUND
TST ACTREG ;ARE ANY DEVICES ACTIVE ?
BNE RUNIT ;YES
TYPE MCOW ;NO
MOV ACTREG,PO ;DISPLAY ACTREG
HALT ;SELECT SOMETHING TO RUN @ ACTREG:
;SELECT SWITCHES & HIT CONTINUE (PUT SW00 =1)
JMP DU11 ;START OVER AGAIN.....YOU Deselected EVERYTHING
RUNIT: ADD #10,BASEADD ;NEXT BLOCK (ADDRESSES)
ZERO: ADD #10,BASEIV ;NEXT BLOCK (VECTORS)
CLC
ROL ROTADD ;UP DATE ROTATING POINTER
BCS Z$ ;IS IT THE LAST DEVICE
;TO BE TESTED IN THIS PASS ?
BIT ROTADD,ACTREG ;TEST THIS DEVICE FOR ACTIVE STATUS
BEQ RUNIT ;IF NOT ACTIVE, TRY NEXT ADDRESS
JSR PC,REPLAY ;CALCULATE NEW PARAMETERS
JSR PC,OKADR ;CHECK FOR GOOD ADDRESS
JMP ARESTRT ;YES IT WAS ACTIVE, TEST THIS DEVICE
Z$: MOV #1,ROTADD ;OK! NOW SET UP ROTATING
;POINTER FOR NEXT MULTIPLE PASS
MOV KEEPADD,BASEADD ;RESTORE BASE ADDRESS
MOV KEEPIV,BASEIV ;RESTORE BASE INTERRUPT VECTORS
JSR PC,REPLAY ;CALC NEW PARAMETERS
BR CCC ;JUMP AROUND REPLAY
REPLAY: MOV BASEADD,DUBASE ;SET UP FOR NEW ADDRESSES
JSR PC,DUADDR ;CREATE NEW ADDRESSES
MOV BASEIV,DURIV ;CREATE DURIV
ADD #2,BASEIV
MOV BASEIV,DURIS ;CREATE DURIS
ADD #2,BASEIV
MOV BASEIV,DUTIV ;CREATE DUTIV
ADD #2,BASEIV
MOV BASEIV,DUTIS ;CREATE DUTIS
MOV DURIV,BASEIV ;RESTORE
RTS PC

CCC: MOV @#42,R1 ;CHECK FOR ACT-11 OR DDP
BEQ ARESTRT ;IF NOT, CONTINUE TESTING
LOGICAL: JSR PC,(R1)
  
```

# M03

DZDFA-6 MACY11 27(732) 17-SEP-76 16:06 PAGE 39  
 DZDFAB.P11

1821	007532	000240			28100		NOP	
1822	007534	000240			28200		NOP	
1823	007536	000240			28300		NOP	
1824	007540	012737	000340	177776	28400	ARESTRT:	MOV	#340,PS ;PREVENT INTERRUPTS (PRIO: 7)
1825	007546	000177	171434		28500		JMP	@CONT.
1826					28600			
1827	007552	017637	000000	001106		.DELAY:	MOV	@(SP),TEMP3 ;SET UP COUNT
1828	007560	062716	000002				ADD	#2,(SP)
1829	007564	011637	001110				MOV	(SP),TEMP4
1830	007570	022626					POP	.SP
1831	007572	005037	177776				CLR	PS
1832	007576	162737	001106				SUB	(PC)+,TEMP3 ;FOR A SMALLER DELAY TIME
1833	007602					LESS.TIME:		;PUT A NUMBER IN HERE TO BE
1834	007602	000000					000	;SUBTRACTED FROM TOTAL.
1835	007604	062737	001106				ADD	(PC)+,TEMP3 ;FOR A LONGER DELAY TIME
1836	007610					MORE.TIME:		;PUT A NUMBER IN HERE TO BE ADDED
1837	007610	000000					000	;TO THE TOTAL TIME.
1838	007612	162737	000001	001106		1\$:	SUB	#1,TEMP3 ;TIME OUT
1839	007620	001374					BNE	1\$ ;NO
1840	007622	000177	171262				JMP	@TEMP4
1841								

```

1842
1843
1844
1845 007626 032737 040000 177570
1846 007634 001407
1847 007636 000432
1848 007640 105777 171204
1849 007644 100027
1850 007646 017700 171200
1851 007652 000412
1852 007654 032737 004000 177570
1853 007662 001006
1854 007664 005237 001070
1855 007670 023737 001070 001066
1856 007676 001012
1857 007700 105037 001242
1858 007704 005037 001070
1859 007710 012737 000062 001066
1860 007716 013737 001062 001060
1861 007724 013716 001060
1862 007730 000002
1863 007732 001407
1864 007734 000432
1865
1866
1867
1868
1869
1870
1871
1872
1873 007736 005037 001100
1874 007742 105037 001242
1875 007746 005237 001074
1876 007752 104403
1877 007754 011410
1878 007756 104407
1879 007760 010020
1880 007762 105737 001144
1881 007766 100402
1882 007770 005237 001156
1883 007774 013737 001074 177570
1884 010002 005737 001234
1885 010006 001002
1886 010010 000137 007272
1887 010014 000177 171166
1888 010020 000001
1889 010022 006 002
1890 010024 001210
1891
1892
1893
1894 010026 032737 001000 177570
1895 010034 001402
1896 010036 013716 001064
1897 010042 000002

;SCOPE LOOP AND INTERATION HANDLER
.SCOPE: BIT #BIT14,SWR
TTST: BEQ 1$
BR 3$
TSTB @TKCSR
BPL 3$
MOV @TKDBR,RO
BR 2$
1$: BIT #SW11,SWR.
BNE 2$
INC LPCNT
CMP LPCNT,ICOUNT
BNE 3$
2$: CLRB ERRFLG
CLR LPCNT
MOV #50,ICOUNT
MOV NEXT,RETURN
3$: MOV RETURN,(SP)
RTI
BRW: 1407
BRX: 432

;END OF PASS
;TYPE "END OF PASS CSR: XXXXXX"
;UPDATE PASS COUNT
;UPDATE LINE NUMBER
;IF IN CYCLE MODE
;RESTART TEST
.EOP: CLR LSTERR ;CLEAR LAST ERROR PC
CLRB ERRFLG ;CLEAR ERROR FLAG
INC PASCNT ;UPDATE PASS COUNT
TYPE
MEPASS
CNVRT
XCSR
TSTB SAVSR1
BMI .+6
INC XLINEX
MOV PASCNT,LIGHTS ;DISPLAY PASS COUNT
TST TAG ;DP OR DU
BNE RESTRT ;DP
JMP DU.UPDATE ;DU
RESTRT: JMP @CONT. ;CONTINUE TO TEST APPRO. DEVICE
XCSR: 1
.BYTE 6,2
DEVICE

;CHECK FOR FREEZE ON CURRENT DATA
.SCOPE1: BIT #SW09,SWR
BEQ 1$
MOV LOCK,(SP)
1$: RTI

```

1899  
1900  
1901  
1902  
1903  
1904  
1905  
1906  
1907  
1908  
1909  
1910  
1911  
1912  
1913  
1914  
1915  
1916  
1917  
1918  
1919  
1920  
1921  
1922  
1923  
1924  
1925  
1926  
1927  
1928  
1929  
1930  
1931  
1932  
1933  
1934  
1935  
1936  
1937  
1938  
1939  
1940  
1941  
1942  
1943  
1944  
1945  
1946  
1947  
1948  
1949  
1950  
1951  
1952  
1953

010044 017605 000000  
010050 062716 000002  
010054 032737 010000 177570  
010062 001010  
010064 105715  
010066 001406  
010070 105777 170760  
010074 100375  
010076 112577 170754  
010102 000764  
010104 000002  
  
010106 032737 010000 177570  
010114 001406  
010116 105777 170732  
010122 100003  
010124 112777 000207 170724  
010132 032737 020000 177570  
010140 001067  
010142 021637 001100  
010146 001404  
010150 011637 001100  
010154 105737 001242  
010160 104734  
010162 011605  
010164 162705 000002  
010170 011504  
010172 006304  
010174 061504  
010176 006304  
010200 042704 177001  
010204 062704 013632  
010210 012437 010270  
010214 012437 010302  
010220 011437 010314  
010224 105737 001242  
010230 001403  
010232 005737 010314  
010236 001022  
010240 104403  
010242 000001  
010244 104403  
010246 011456  
010250 104407  
010252 010404  
010254 104403  
010256 011473  
010260 112737 177777 001242  
010266 104403  
010270 000000  
010272 005737 010302

:TELETYPE OUTPUT ROUTINE  
.TYPE: MOV @ (SP), R5  
ADD #2, (SP)  
1\$: BIT #SW12, SWR  
BNE 3\$  
TSTB (R5)  
BEQ 3\$  
2\$: TSTB @TPCSR  
BPL 2\$  
MOVB (R5)+, @TPDBR  
BR 1\$  
3\$: RTI  
  
:ERROR HANDLER  
.HLT: BIT #SW12, SWR  
BEQ XBX  
TSTB @TPCSR  
BPL XBX  
MOVB #207, @TFDBR  
XBX: BIT #SW13, SWR  
ENE HALTS  
CMP (SP), LSTERR  
BEQ 1\$  
MOV (SP), LSTERR  
1\$: CLRB ERRFLG  
SAVOS  
MOV (SP), R5  
SUB #2, R5  
MOV (R5), R4  
ASL R4  
ADD (R5), R4  
ASL R4  
BIC #177001, R4  
ADD #.ERRTAB, R4  
MOV (R4)+, ERRMSG  
MOV (R4)+, DATAHD  
MOV (R4), DATABP  
TSTB ERRFLG  
BEQ TYPMSG  
TST DATABP  
BNE TYPDAT  
TYPMSG: TYPE  
WHO: .BLKW 1  
TYPE  
MERRPC  
CNVRT  
ERTAB0  
TYPE  
MCRLF  
MOVB #-1, ERRFLG  
ERRMSG: 0  
TST DATAHD

```

1954 0103276 001402
1955 0103300 104403
1956 0103302 000003
1957 0103304 005737 010314
1958 0103310 001402
1959 010312 104406
1960 010314 000000
1961 010316 104405
1962 010320 005737 177570
1963 010324 100005
1964 010326 010046
1965 010330 016600 000002
1966 010334 000000
1967 010336 012600
1968 010340 005237 001076
1969 010344 032737 000400 177570
1970 010352 001007
1971 010354 032737 002000 177570
1972 010362 001407
1973 010364 013737 001062 001060
1974 010372 012706 001050
1975 010376 000177 170456
1976 010402 000002
1977 010404 000001
1978 010406 006 002
1979 010410 001142
1980
1981
1982
1983 010412 104403
1984 010414 011473
1985 010416 017601 000000
1986 010422 013737 013756 001106
1987 010420 062716 000002
1988 010434 012137 010604
1989 010440 112137 010606
1990 010444 112137 010607
1991 010450 013137 010610
1992 010454 013704 010610
1993 010460 113705 010606
1994 010464 012700 013756
1995 010470 010403
1996 010472 042703 177770
1997 010476 062703 000260
1998 010502 110320
1999 010504 000241
2000 010506 006004
2001 010510 000241
2002 010512 006004
2003 010514 000241
2004 010516 006004
2005 010520 005305
2006 010522 001362
2007 010524 012703 014020
2008 010530 114023
2009 010532 105337 010606

```

```

BEQ TYPDAT
TYPE
DATAHD: 0
TYPDAT: TST DATABP
BEQ RESREG
CONVRT
DATABP: 0
RESREG: RESOS
HALTS: TST SWR
BPL EXITER
PUSHRO
MOV 2(SP),R0
HALT
POPRO
EXITER: INC ERRCNT
BIT #SWC9,SWR
BNE 1$
BIT #SW10,SWR
BEQ 2$
MOV NEXT,RETURN
1$: MOV #STACK,SP
JMP JRETURN
2$: RTI
ERRABD: 1
.BYTE 6,2
SAVPC

```

;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER

```

.CONVR: TYPE
MCR LF
.CNVRT: MOV 2(SP),R1
MOV TEMP,TEMP3
ADD #2,(SP)
MOV (R1)+,WORDCNT
1$: MOVB (R1)+,CHRCNT
MOVB (R1)+,SPACNT
MOV 2(R1)+,BINWRD
2$: MOV BINWRD,R4
MOVB CHRCNT,R5
MOV #TEMP,R0
3$: MOV R4,R3
BIC #177770,R3
ADD #260,R3
MOVB R3,(R0)+
CLC
ROR R4
CLC
ROR R4
CLC
ROR R4
DEC R5
BNE 3$
MOV #MDATA,R3
4$: MOVB -(R0),(R3)+
DECB CHRCNT

```

2010	010536	001374			BNE	4\$	
2011	010540	105737	010607		TSTB	SPACNT	
2012	010544	001405			BEQ	6\$	
2013	010546	112723	000240	5\$:	MOVB	#240,(R3)+	
2014	010552	105337	010607		DECB	SPACNT	
2015	010556	001373			BNE	5\$	
2016	010560	105013		5\$:	CLRB	(R3)	
2017	010562	104403			TYPE		
2018	010564	014020			MDATA		
2019	010566	005337	010604		DEC	WRDCNT	
2020	010572	001322			BNE	1\$	
2021	010574	013737	001106	013756	MOV	TEMP3,TEMP	
2022	010602	000002			RTI		
2023	010604	000000			WRDCNT:	0	
2024	010606	000000			CHRCNT:	0	
2025		010607			SPACNT=CHRCNT+1		
2026	C:0610	000000			BINWRD:	0	
2027							
2028							
2029							
2030	010612	016637	000004	001142	.SAV05:	MOV	4(SP),SAVPC
2031							
2032							
2033							
2034	010620	010537	001136		SV05:	MOV	R5,SAVR5
2035	010624	010437	001134			MOV	R4,SAVR4
2036	010630	010337	001132			MOV	R3,SAVR3
2037	010634	010237	001130			MOV	R2,SAVR2
2038	010640	010137	001126			MOV	R1,SAVR1
2039	010644	010037	001124			MOV	R0,SAVR0
2040	010650	000002				RTI	
2041							
2042							
2043							
2044	010652	013700	001124		.RES05:	MOV	SAVR0,R0
2045	010656	013701	001126			MOV	SAVR1,R1
2046	010662	013702	001130			MOV	SAVR2,R2
2047	010666	013703	001132			MOV	SAVR3,R3
2048	010672	013704	001134			MOV	SAVR4,R4
2049	010676	013705	001136			MOV	SAVR5,R5
2050	010702	000002				RTI	
2051							
2052							
2053							
2054	010704	011605			.PARAM:	MOV	(SP),R5 ;PUT CONTENTS OF SP INTO R5
2055	010706	012537	011060			MOV	(R5)+,LOLIM ;PUT LOW LIMIT INTO LOLIM
2056	010712	012537	011062			MOV	(R5)+,HILIM ;PUT HIGH LIMIT INTO HILIM
2057	010716	012537	011064			MOV	(R5)+,DEVADR ;PUT STORE LOC INTO DEVADR
2058	010722	112537	011066			MOVB	(R5)+,LOBITS ;PUT MASK INTO LOBITS
2059	010726	112537	011067			MOVB	(R5)+,ADRCNT ;PUT COUNT INTO ADRCNT
2060	010732	010516				MOV	R5,(SP) ;RESTORE RETURN ADDR ON STACK FOR RTI
2061	010734	005005			PARAM1:	CLR	R5
2062	010736	012704	014062			MOV	#INBUF,R4
2063	010742	122714	000015			CMPB	#15,(R4) ;CR ?
2064	010746	001420				BEQ	PARERR ;YOU TYPED CR TOO SOON !
2065	010750	121427	000060		1\$:	CMPB	(R4),#60 ;LOW LIMIT ASCII 0

# E04

2066	010754	002415			BLT	PARERR	
2067	010756	121427	000067		CMPB	(R4), #67	;HIGH LIMIT ASCII 7
2068	010762	003012			BGT	PARERR	
2069	010764	142714	000060		BICB	#60, (R4)	;CONVERT TO OCTAL
2070	010770	152405			BISB	(R4)+, R5	;STORE AWAY ITS AN OK CHAR
2071	010772	122714	000015		CMPB	#15, (R4)	;CR ?
2072	010776	001406			BEQ	LIMITS	;NOW CHECK FOR HIGH & LOW LIMIT CONDS
2073	011000	006305			ASL	R5	;ALLOCATE ROOM FOR NEXT CHAR
2074	011002	006305			ASL	R5	
2075	011004	006305			ASL	R5	
2076	011006	000760			BR	1\$	
2077	011010	104413			PARERR:	INSTR	;RETRY
2078	011012	000750			BR	PARAM1	
2079							
2080							;TEST TO SEE IF NUMBER IS WITHIN LIMITS
2081							
2082	011014	020537	011062		LIMITS:	CMP	R5, HILIM
2083	011020	101373			BHI	PARERR	;THE # IS TOO HIGH
2084	011022	020537	011060		CMP	R5, LCLIM	
2085	011026	103770			BLO	PARERR	;THE # IS TOO LOW
2086	011030	133705	011066		BITB	LOBITS, R5	;TEST BY MASKING THE #
2087	011034	001365			BNE	PARERR	
2088							
2089							;STORE NUMBER AT SPECIFIED ADDRESS
2090							
2091	011036	013704	011064		1\$:	MOV	DEVADR, R4
2092	011042	010524				MOV	R5, (R4)+
2093	011044	062705	000002			ADD	#2, R5
2094	011050	105337	011067			DECB	ADRCNT
2095	011054	001372				BNE	1\$
2096	011056	000002				RTI	
2097	011060	000000			LCLIM:	0	
2098	011062	000000			HILIM:	0	
2099	011064	000000			DEVADR:	0	
2100	011066	000000			LOBITS:	0	
2101		011067			ADRCNT=	LOBITS+1	
2102							
2103							;ASCII STRING INPUT ROUTINE
2104							
2105	011070	017637	000000	011104	.INSTR:	MOV	3(SP), MSG
2106	011076	062716	000002			ADD	#2, (SP)
2107	011102	104403			.INST1:	TYPE	
2108	011104	000000			.MSG:	0	
2109	011106	012704	014062			MOV	#INBUF, R4
2110	011112	012703	000007			MOV	#7, R3
2111	011116	105777	167726		1\$:	TSTB	3TKCSR
2112	011122	100375				BPL	1\$
2113	011124	117714	167722			MOVB	3TKDBR, (R4)
2114	011130	142714	000200			BICB	#200, (R4)
2115	011134	122427	000015			CMPB	(R4)+, #15
2116	011140	001413				BEQ	INSTR2
2117	011142	105777	167706		2\$:	TSTB	3TPCSR
2118	011146	100375				BPL	2\$
2119	011150	117777	167676	167700		MOVB	3TKDBR, 3TPDBR
2120	011156	005303				DEC	R3
2121	011160	001356				BNE	1\$

2122	011162	104403		
2123	011164	012243		
2124	011166	000745		
2125	011170	000002		
2126				
2127				
2128				
2129				
2130				
2131				
2132	011172	017605	000000	
2133	011176	122737	000116	014062
2134	011204	001002		
2135	011206	105015		
2136	011210	000406		
2137	011212	122737	000131	014062
2138	011220	001005		
2139	011222	112715	177777	
2140	011226	062716	000002	
2141	011232	000002		
2142	011234	104413		
2143	011236	000755		
2144	011240	105777	167604	
2145	011244	100005		
2146	011246	005777	167600	
2147	011252	013737	001062	001060
2148	011260	013716	001060	
2149	011264	000002		
2150				
2151				
2152				
2153				
2154				
2155	011266	011646		
2156	011270	162716	000002	
2157	011274	017616	000000	
2158	011300	006316		
2159	011302	042716	177001	
2160	011306	062716	001270	
2161	011312	017616	000000	
2162	011316	000136		
2163				01400
2164				01500
2165				01600
2166				01700
2167				01800
2168				01900
2169				02000
2170	011320	005015	050104	030461
	011345	015	042012	030525
	011372	005015	053520	020122
	011410	003407	005015	047105
	011433	015	052012	051505
	011447	114	047111	035105
	011456	005015	051105	047522
	011473	015	000012	03100

```

.INSTE: TYPE
        MQM      ;?
        BR       .INST1      ;RETRY
INSTR2: RTI

;COMPARE THE FIRST CHARACTER IN THE TELETYPE INPUT
;BUFFER TO THE CHARACTERS "N" AND "Y".
;IF THE CHARACTER IS "N" CLEAR THE FLAG
;IF THE CHARACTER IS "Y" SET THE FLAG

.SETFLG:MOV    2(SP),R5
        CMPB   #'N,INBUF      ;IS IT "N" ?
        BNE   1$
        CLRB   (R5)           ;000
        BR    2$
1$:     CMPB   #'Y,INBUF      ;IS IT "Y" ?
        BNE   3$
        MOVB   #-1,(R5)      ;377
2$:     ADD    #2,(SP)
        RTI
3$:     INSTER                ;RETRY
        BR    .SETFLG
.SCOF2: TSTB   @TKCSR
        BPL   3$
        TST   @TKDBR
        MOV   NEXT,RETURN
3$:     MOV    RETURN,(SP)
        RTI
;TRAP DISPATCH SERVICE
;ARGUMENT OF TRAP IS EXTRACTED
;AND USED AS OFFSET TO OBTAIN POINTER
;TO SELECTED SUBROUTINE

.TRPSR: MOV    (SP),-(SP)      ;GET PC OF RETURN
        SUB    #2,(SP)        ;=PC OF TRAP
        MOV    2(SP),(SP)     ;GET TRP
TRPOK:  ASL    (SP)           ;MULTIPLY TRAP ARG BY 2
        BIC   #177001,(SP)   ;CLEAR UNWANTED BITS
        ADD   #.TRPTAB,(SP)  ;POINTER TO SUBROUTINE ADDRESS
        MOV   2(SP),(SP)     ;SUBROUTINE ADDRESS
        JMP   2(SP)+        ;GO TO SUBROUTINE

;*****
;TTY MESSAGES
;*****

EDP11:  .ASCIZ <15><12>'DP11/DFC11-A TEST ?
EDU11:  .ASCIZ <15><12>'DU11/DFC11-A TEST ?
MPOWER: .ASCIZ <15><12>'PWR FAILED /
MEPASS: .ASCIZ <7><7><15><12>'END PASS CSR: /
MTSTN:  .ASCIZ <15><12>'TEST NO. /
MLINE:  .ASCIZ /LINE: /
MERRPC: .ASCIZ <15><12>'ERROR PC: /
MCKLF:  .ASCIZ <15><12>

```

G04

011476	005015	042012	053105	03200	DUTITLE: .ASCII <15><12><12>/DEVICE UNDER TEST-DU11 &DFC11A /<15><12>
011542	042115	030455	026461	03300	.ASCIZ /MD-11-DZDFA-B /<15><12>
011563	015	005012	042504	03400	DPTITLE: .ASCII <15><12><12>/DEVICE UNDER TEST-DP11 &DFC11A /<15><12>
011627	115	026504	030461	03500	.ASCIZ /MD-11-DZDFA-B /<15><12>
011650	005015	042526	052103	03600	MVECTO: .ASCIZ <15><12>/VECTOR ADDRESS-/
011672	005015	051461	020124	03700	MREGAD: .ASCIZ <15><12>/1ST DEVICE: RECEIVER CONTROL REGISTER ADDRESS-/
011753	015	040412	042522	03800	MMULT: .ASCIZ <15><12>/ARE YOU RUNNING MULTIPLE DEVICES ? (Y OR N)-/
012032	005015	040514	052123	03900	MLASTD: .ASCIZ <15><12>/LAST DEVICE:RECEIVER CONTROL REGISTER ADDRESS-/
012113	015	020012	027056	04000	MCOW: .ASCIZ <15><12>/...SELECT SOMETHING TO RUN QACTREG /
012162	005015	052517	020124	04100	MRANGE: .ASCIZ <15><12>/OUT OF RANGE:RETYPE LAST DEVICE RXCSR ADDRESS-/
012243	040	037440	000040	04200	MQM: .ASCIZ / ? /
012250	005015	020122	000	04300	MR: .ASCIZ <15><12>/R /
				04400	
012255	015	051012	043505	04500	EM1: .ASCIZ <15><12>/REG. ADDRESS /
012276	005015	051124	047101	04600	EM2: .ASCIZ <15><12>/TRANSMITTER /
012316	005015	042522	042503	04700	EM3: .ASCIZ <15><12>/RECEIVER /
012333	015	052412	042516	04800	EM4: .ASCIZ <15><12>/UNEXPECTED INTERRUPT /
012363	105	050130	041505	04900	DH0: .ASCIZ /EXPECTED RECEIVED /
012407	106	044501	042514	05000	DH1: .ASCIZ /FAILED TO INTERRUPT. /
012434	047111	042524	052522	05100	DH2: .ASCIZ /INTERRUPTED UNEXPECTEDLY. /
012466	005015	051124	050101	05200	DH3: .ASCIZ <15><12>/TRAPPED FROM /
012506	005015	054523	052123	05300	MNOCLOCK: .ASCII <15><12>/SYSTEM HAS NO CLOCK!!/
012535	015	050012	052125	05400	MSCOPBAUD: .ASCIZ <15><12>/PUT UP SWITCH 14 FOR SCOPE LOOP /
012600	005015	051511	052040	05500	MCPS: .ASCIZ <15><12>/IS THE AC LINE INPUT 60 CYCLES? (Y OR N) --- /
012663	015	042012	020117	05600	MRBAUD: .ASCIZ <15><12>/DO YOU STILL WISH TO RUN BAUD TEST USING SCOPE LOOP? (
012773	015	052012	042510	05700	MNODEV: .ASCIZ <15><12>/THERE IS NO DEVICE PRESENT AT THE ADDRESS YOU SELECTED-
013112	034461	030062	020060	05800	M19200: .ASCIZ /19200 /
013122	033071	030060	020040	05900	M9600: .ASCIZ /9600 /
013131	064	030070	020060	06000	M4800: .ASCIZ /4800 /
013140	032062	030060	020040	06100	M2400: .ASCIZ /2400 /
013147	061	030062	020060	06200	M1200: .ASCIZ /1200 /
013156	030066	020060	000040	06300	M600: .ASCIZ /600 /
013164	030063	020060	000040	06400	M300: .ASCIZ /300 /
013172	032461	020060	000040	06500	M150: .ASCIZ /150 /
013200	032467	020040	000	06600	M75: .ASCIZ /75 /
013205	015	041412	047101	06700	MWRONGRT: .ASCII <15><12>/CAN'T DETERMINE RATE--PUT UP/<15><12>
013245	123	020127	032061	06800	.ASCIZ /SW 14 IF SCOPE LOOP IS WANTED. /
013306	005015	047531	051125	06900	MBAUDRT: .ASCIZ <15><12>/YOUR BAUD RATE IS /
013334	041040	052501	027104	07000	MWRITE: .ASCIZ /BAUD. IF NOT CORRECT, USE A SCOPE!!!/
013402	005015	047504	054440	07100	MBDRT: .ASCIZ<15><12>/DO YOU WISH TO CHECK BAUD RATE? (Y OR N) --- /
013465	015	054412	052517	07200	MTSTM: .ASCIZ <15><12>/YOU TOOK TOO LONG TO SCOPE OR DFC FAILED TEST. /
013547	015	042012	020117	09100	MBDTEST: .ASCIZ <15><12>/DO YOU WANT TO RUN BAUD RATE TEST? (Y OR N) --- /
				09200	.EVEN
				09600	.ERRTAB:
2171	013632	000000		09700	0
2172	013634	000000		09800	0
2173	013636	000000		09900	0
2174	013640	011473		10000	MCRLF
2175	013642	012363		10100	DH0 ;HALT 1
2176	013644	013720		10200	DT0
2177				10300	
2178	013646	012255		10400	EM1
2179	013650	012363		10500	DH0 ;HALT 2
2180	013652	013732		10600	DT1
2181				10700	
2182	013654	012276		10800	EM2

H04

2183	013656	012407		10900		DH1	:HALT 3
2184	013660	000000		11000		0	
2185				11100			
2186	013662	012316		11200		EM3	
2187	013664	012407		11300		DH1	:HALT 4
2188	013666	000000		11400		0	
2189				11500			
2190	013670	012276		11600		EM2	
2191	013672	012434		11700		DH2	:HALT 5
2192	013674	000000		11800		0	
2193				11900			
2194	013676	012316		12000		EM3	
2195	013700	012434		12100		DH2	:HALT 6
2196	013702	000000		12200		0	
2197				12300			
2198	013704	011473		12400		MCRLF	
2199	013706	013465		12500		MTSTM	:HALT 7
2200	013710	000000		12600		0	
2201				12700			
2202	013712	012333		12800		EM4	
2203	013714	012466		12900		DH3	:HALT 10
2204	013716	013750		13000		DT2	
2205	013720	000002		13100	DT0:	2	
2206	013722	006	004	13200		.BYTE	6,4
2207	013724	001124		13300		SAVRO	
2208	013726	006	002	13400		.BYTE	6.2
2209	013730	001126		13500		SAVRI	
2210				13600			
2211	013732	000003		13700	DT1:	3	
2212	013734	006	010	13800		.BYTE	6.8.
2213	013736	001110		13900		TEMP4	
2214	013740	006	004	14000		.BYTE	6,4
2215	013742	001124		14100		SAVRO	
2216	013744	006	002	14200		.BYTE	6,2
2217	013746	001126		14300		SAVRI	
2218				14400			
2219	013750	000001		14500	DT2:	1	
2220	013752	006	002	14600		.BYTE	6.2
2221	013754	006012		14700		TRP.PC	
2222	013756	000000		14800			
2223		014020		14900	TEMP:	0	
2224	014020	000000		15000	.=. +40		
2225		014062		15100	MDATA:	0	
2226	014062	000000		15200	.=. +40		
2227		014124		15300	INBUF:	0	
2228		000001		15400	.=. +40		
					.END		



















# E05

DZDFA-B MACY11 27(732) 17-SEP-76 16:06 PAGE 60  
 DZDFAB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ADD	978 1934	1593 1987	1751 1997	1763 2093	1766 2106	1768 2140	1786 2160	1787 2074	1806 2075	1808 2158	1810	1828	1835	1902	1931
ASL	1563	1566	1567	1568	1930	1932	2073								
ASR	1564	1744													
BZS	977	1790													
BEJ	787	791	823	903	948	958	973	978	1016	1029	1037	1050	1131	1143	1253
	1270	1323	1326	1344	1351	1501	1509	1516	1604	1678	1715	1735	1753	1778	1793
	1818	1846	1895	1906	1916	1923	1939	1954	1958	1972	2012	2064	2072	2116	
BGE	1110	1186	1207	1253											
BGT	2069														
BHI	880	2093													
BIC	946	971	975	1079	1090	1121	1124	1141	1209	1210	1232	1233	1246	1255	1268
	1305	1340	1354	1559	1589	1724	1738	1748	1757	1933	1996	2159			
BICB	2069	2114													
BIS	874	881	1071	1074	1083	1101	1165	1167	1172	1196	1197	1301	1307	1345	1517
	1675	1680	1691	1692	1747	1749	1750								
BISB	2070														
BIT	790	820	822	957	1028	1059	1500	1520	1677	1714	1792	1845	1852	1894	1903
	1915	1920	1969	1971											
BITB	2086														
BLO	1367	1380	1393	1406	1419	1432	1445	1458	2085						
BLOS	1363	1376	1389	1402	1415	1428	1441	1454							
BLT	2066														
BMI	795	858	1238	1572	1591	1718	1730	1881							
BNE	766	780	817	821	950	956	960	1008	1085	1092	1107	1183	1204	1250	1315
	1336	1338	1342	1521	1628	1696	1698	1707	1723	1741	1780	1839	1853	1856	1895
	1904	1921	1941	1970	2006	2010	2015	2020	2097	2095	2121	2134	2138		
BPL	769	1849	1908	1918	1963	2112	2118	2145							
BR	774	796	825	886	1088	1112	1128	1221	1261	1346	1482	1499	1519	1560	1601
	1614	1642	1646	1683	1802	1847	1851	1910	2076	2078	2124	2136	2143		
CLC	875	1788	1959	2001	2003										
CLR	499	504	505	506	759	760	762	804	810	811	812	859	860	873	940
	941	942	954	985	986	1004	1097	1104	1105	1119	1127	1180	1181	1191	1201
	1202	1212	1214	1228	1230	1247	1248	1257	1259	1299	1332	1333	1478	1548	1557
	1552	1602	1625	1671	1672	1673	1674	1742	1743	1755	1756	1831	1858	1873	2061
CLRB	757	761	805	1670	1857	1874	1925	2016	2135						
CMP	879	947	972	977	1091	1109	1130	1142	1185	1206	1252	1262	1269	1325	1350
	1362	1366	1375	1379	1388	1392	1401	1405	1414	1418	1427	1431	1440	1444	1453
	1457	1484	1490	1626	1627	1644	1722	1734	1740	1752	1855	1922	2082	2094	
CMPB	2063	2065	2067	2071	2115	2133	2137								
COM	758														
COMB	781	819													
DEC	764	959	1084	1237	1335	1337	1697	1706	2005	2019	2120				
DECB	2009	2014	2094												
EMT	462														
HALT	488	1039	1511	1551	1556	1645	1659	1783	1966						
INC	949	955	1022	1106	1108	1182	1184	1203	1205	1249	1251	1341	1533	1580	1582
	1586	1695	1721	1739	1854	1875	1882	1968							
INCB	1682														
JMP	500	507	510	788	793	828	861	896	904	906	992	1009	1019	1030	1040
	1042	1043	1061	1198	1316	1327	1359	1365	1372	1379	1385	1391	1398	1404	1411
	1417	1424	1430	1437	1443	1450	1456	1463	1467	1499	1502	1512	1513	1523	1665
	1785	1796	1825	1840	1886	1887	1975	2162							
JSR	770	773	775	776	838	840	997	1006	1313	1609	1794	1795	1801	1804	1820
MOV	503	751	752	753	754	755	756	763	778	789	792	801	802	803	906
	807	808	809	813	814	815	824	827	839	841	950	851	872	984	995



DZDFA-B MACY11 27(732) 17-SEP-76 16:06 PAGE 62  
DZDFAB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

.REPT	498								
.TITLE	415								
.WORD	580	581	582	583	584	585	586	765	

ERRORS DETECTED: 0  
DEFAULT GLOBALS GENERAT 0

\*.DZDFAB.SEQ/SOL/CRF/ .UM=DZDFAB  
RUN-TIME: 13 21 3 SECONDS  
RUN-TIME RATIO: 108/38=2.8  
CORE USED: 13K (25 PAGES)

