

# DU11

OVERLAY FOR ITEP  
MD-11-DZDUO-B

EP-DZDUO-B-DL-A

NOV 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

MADE IN U.S.A.

This microfiche card contains a grid of frames. The first column contains 15 frames of text, likely technical specifications or code. The second column contains 15 frames of data, possibly tables or charts. The third column contains 15 frames of data, possibly tables or charts. The frames are arranged in a 3x5 grid.

Frame 1	Frame 2	Frame 3
Frame 4	Frame 5	Frame 6
Frame 7	Frame 8	Frame 9
Frame 10	Frame 11	Frame 12
Frame 13	Frame 14	Frame 15
Frame 16	Frame 17	Frame 18
Frame 19	Frame 20	Frame 21
Frame 22	Frame 23	Frame 24
Frame 25	Frame 26	Frame 27
Frame 28	Frame 29	Frame 30
Frame 31	Frame 32	Frame 33
Frame 34	Frame 35	Frame 36
Frame 37	Frame 38	Frame 39
Frame 40	Frame 41	Frame 42
Frame 43	Frame 44	Frame 45
Frame 46	Frame 47	Frame 48
Frame 49	Frame 50	Frame 51
Frame 52	Frame 53	Frame 54
Frame 55	Frame 56	Frame 57
Frame 58	Frame 59	Frame 60
Frame 61	Frame 62	Frame 63
Frame 64	Frame 65	Frame 66
Frame 67	Frame 68	Frame 69
Frame 70	Frame 71	Frame 72
Frame 73	Frame 74	Frame 75
Frame 76	Frame 77	Frame 78
Frame 79	Frame 80	Frame 81
Frame 82	Frame 83	Frame 84
Frame 85	Frame 86	Frame 87
Frame 88	Frame 89	Frame 90
Frame 91	Frame 92	Frame 93
Frame 94	Frame 95	Frame 96
Frame 97	Frame 98	Frame 99
Frame 100	Frame 101	Frame 102

IDENTIFICATION

PRODUCT CODE:            MAINDEC-11-DZDU0-B-D

PRODUCT NAME:            DU11 OVERLAY FOR INTERPROCESSOR TEST PROGRAM

PROGRAM DATE:            OCTOBER 1976

MAINTAINER:              DIAGNOSTICS

AUTHORS:                  R A JONES  
                              JOHN EGOLF  
                              GEORGE BALSLEY

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

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

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

COPYRIGHT (C) 1973, 1976, BY DIGITAL EQUIPMENT CORPORATION

## 1.0 ABSTRACT.

THIS PROGRAM IS DESIGNED AS A MAINTENANCE AID FOR FIELD SERVICE PERSONEL. IT WILL VERIFY THE PROPER OPERATION OF A COMPLETE COMMUNICATION LINK FROM ONE PDP-11 SYSTEM TO ANOTHER OR TO A COMMUNICATION TEST CENTER.

THIS PROGRAM MUST BE USED IN CONJUNCTION WITH THE INTERPROCESSOR TEST PROGRAM(DZITP) ON A PDP-11 SYSTEM WITH A DL-11 INTERFACE.

## 2.0 REQUIREMENTS.

### 2.1 EQUIPMENT

- A. PDP-11 SYSTEM WITH 4K OF CORE.
- B. A DUII COMMUNICATION INTERFACE.

### 2.2 STORAGE.

4K OF CORE

## 3.0 LOADING PROCEDURE

THIS PROGRAM IS IN ABSOLUTE FORMAT.  
 THE ABS LOADER MUST BE USED TO LOAD THE PROGRAM.

## 4.0 OPERATING PROCEDURES.

- A. TWO METHODS OF ENTERING PARAMETERS ARE PROVIDED
  - 1. LOAD ADDRESS 200 AND START TO ENTER PARAMS FROM CONSOLE TTY, PROCEED TO SECTION B.
  - 2. LOAD ADDRESS 200 AND SET SWITCH REGISTER BIT 15 BEFORE STARTING TO ENTER PARAMS FROM CONSOLE SWITCHES, PROCEED TO SECTION C.
 \*THE PROGRAM MAY BE RESTARTED AT LOC 204 (ONCE PARAMETERS HAVE ALREADY BEEN SELECTED)
- B. CONSOLE DIALOGUE PARAMETER INPUT (CURRENT VALUES FOR PARAMETERS ARE FOUND IN OVERLAY)
  - 1. THE PROGRAM WILL TYPEOUT THE NAME OF THE VARIABLE OVERLAY.
    - A. IF YOU WISH TO SETUP JUST THE INDICATED OVERLAY, TYPE A CARAGE RETURN
    - B. IF YOU WISH TO SETUP A DN11, TYPE IN DN.
    - C. IF YOU WISH TO SETUP A DN1188, TYPE IN DMB.

IF DN OR DMB WAS TYPED IN STEP 1 ABOVE THEN THE BUS ADDRESS, VECTOR ETC. REFERED TO IN STEPS 2 THRU 7, PERTAIN TO THE DN11 OR DMB8.

- 2. THE PROGRAM WILL TYPE THE DEFAULT BUS ADDRESS OF THE INTERFACE UNDER TEST.
  - A. TYPE A CAR. RETURN TO USE DEFAULT BUS ADDRESS
  - B. TYPEIN ACTUAL BUS ADDRESS
- 3. THE PROGRAM WILL TYPE OUT THE DEFAULT VECTOR ADDRESS
  - A. TYPE A CAR. RETURN TO USE DEFAULT ADDRESS
  - B. TYPEIN ACTUAL VECTOR ADDRESS
- 4. THE PROGRAM WILL TYPE OUT THE DEFAULT INTERFACE PRIORITY  
 NOTE: 200=PRIO 4, 240=PRIO 5, 300=PRIO 6, ETC.

- A. TYPE A CAR. RETURN TO USE DEFAULT VALUE  
B. TYPE IN ACTUAL VALUE
5. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#1  
IF REQUIRED BY THE ISR. (SEE SECT. 10.0 IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)  
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE  
B. TYPE IN ACTUAL VALUE
6. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#2  
IF REQUIRED BY THE ISR.  
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE  
B. ENTER ACTUAL VALUE
7. THE PROGRAM WILL TYPEOUT THE DEFAULT VALUE OF PARAM#3  
IF REQUIRED BY THE OVERLAY.  
A. TYPE A CAR. RETURN TO USE DEFAULT VALUE  
THE DN-11 WILL USE PARAM #3 AS THE # TO DIAL.  
IF USING A MODEM WITHOUT AUTOMATIC HANDSHAKING,  
THE NUMBER MUST TERMINATE WITH A  
"END-OF-NUMBER" CHARACTER (:).  
B. ENTER ACTUAL VALUE.
8. THE PROGRAM WILL RETURN TO STEP B1 IF THIS SETUP  
WAS FOR DN11 OR DN11BB.
9. THE PROGRAM WILL REQUEST THAT SWITCH REGISTER BE SET.  
A. SETUP SWITCH REGISTER AS SPECIFIED IN STEP D.  
AND TYPE A CAR. RETURN.

NOTE: IF ANY OF THE ABOVE ITEMS 2 THRU 7 WERE CHANGED BY ENTERING  
NEW VALUES, THE NEW VALUE BECOMES THE DEFAULT VALUE FOR SUBSEQUENT  
RESTARTS OF THE PROGRAM.

- C. MANUAL PARAMETER INPUT FROM SWITCH REGISTER
1. THE PROGRAM HALTS FOR ISR (INTERFACE SERVICE ROUTINE) SPECIFICATION  
SMR14=SETUP DN-11B ISR  
SMR13=SETUP DN-11 ISR  
SMR=000000=SETUP VARIABLE ISR
  2. THE FOLLOWING HALTS ARE REPEATED FOR EACH ISR SPECIFIED.  
SETUP SEQUENCE IS: DN11, DN11-BB THEN VARIABLE OVERLAY. (EACH ENTRY SET SWITCHES THEN HIT CONTINUE.)
    - A. HALT FOR BUS ADDRESS OF INTERFACE
    - B. HALT FOR VECTOR ADDRESS OF INTERFACE
    - C. HALT FOR PRIORITY OF INTERFACE
    - D. HALT FOR INTERFACE PARAM #1 (SEE SECT. 10.0 IN OVERLAY LISTING FOR PARAMETER DESCRIPTION)
    - E. HALT FOR INTERFACE PARAM #2 (DN11 AND DNBB PARAMETERS ARE DISCUSSED IN SECT. 10.0 OF THE MONITOR.)
    - F. GO BACK TO STEP A IF THIS SETUP WAS FOR DN OR DNBB.
  3. HALT FOR OPERATIONAL SWITCH SETTINGS. (SEE STEP D.)
    - A. PRESS CONTINUE TO START TESTING

BEFORE ATTEMPTING TO RUN THIS PROGRAM, THE OPERATOR MUST ACCERTAIN THE COMPLETE COMMUNICATION LOOP AND PROCEDURES TO BE USED, INCLUDING THE TYPE OF MODEMS, THE TYPE OF INTERFACE BEING USED AT THE OTHER CPU AND THE MODES OF OPERATION, DATA AND PARAMETERS TO BE USED AT EACH CPU.

THIS WILL REQUIRED VOCAL COMMUNICATION WITH THE OPERATOR AT THE OTHER CPU UNLESS ITS CONFIGURATION AND OPERATION ARE FIXED AS A TEST CENTER.

AFTER DETERMINING THAT THE EQUIPMENTS ARE COMPATIBLE AND AGREEING ON THE MODE AND VARIABLE PARAMETERS TO BE USED, THE SYSTEM WHICH IS TO RECEIVE DATA FIRST SHOULD BE LOADED AND STARTED. IF THE MODEM BEING USED ON THIS SYSTEM HAS AN AUTOMATIC ANSWER FEATURE, IT SHOULD BE ENABLED.

THE SYSTEM WHICH IS TO TRANSMIT FIRST SHOULD THEN BE LOADED AND STARTED AND THE CONNECTION ESTABLISHED EITHER MANUALLY OR AUTOMATICALLY (VIA DN-11).

## D. OPERATIONAL SWITCH SETTINGS.

SW15=1 HALT ON ERROR  
 SW14=1 SINGLE PASS  
     SW14 HAS NO EFFECT IF SW04=0  
 SW13=1 INHIBIT ERROR TIMEOUTS  
 SW12=1 INHIBIT ALL TIMEOUTS EXCEPT ERRORS  
     IF SW12=0 AND SW04=1 END PASS IS TYPED  
     AND TRANSMITTED/RECEIVED DATA IS TYPED.  
 SW11=1 USE PREVIOUSLY SPECIFIED DATA  
 SW10=1 DATA SELECT (WITH SW09)  
 SW09=1 DATA SELECT (WITH SW10)  
     00=1 GET DATA FROM OPERATOR  
     01=1 TEST MESSAGE #1 (8A QUICK BROWN FOX)  
     10=1 TEST MESSAGE #2 (8B NUMERICS)  
     11=1 TEST MESSAGE #3 (8C CONTEST/QUICK BROWN FOX/NUMERICS)  
 SW08=1 TRANSMIT RECEIVED DATA (INTERNAL LOOPBACK MODE)  
 SW07=1 DO NOT TEST RECEIVED DATA  
 SW06=1 MONITOR TRANSMITTED DATA ON CONSOLE TTY.\*  
 SW05=1 MONITOR RECEIVED DATA ON CONSOLE TTY.\*  
     \* IN MANY CASES, NOT ALL DATA WILL APPEAR ON THE CONSOLE  
     TTY. THIS IS ESPECIALLY TRUE WHEN THE COMM INTERFACE IS  
     RUNNING AT A FASTER BAUD THAN THE CONSOLE, BUT EVEN AT EQUAL  
     OR SLOWER BAUDS, ALL CHARACTERS MAY NOT APPEAR ON THE CONSOLE.  
  
 SW04=1 RETURN TO MONITOR FOR END PASS  
     WHEN SW04=0 PROGRAM LOOPS IN THE OVERLAY NEVER RETURNING TO THE MONITOR.  
 SW03=1 INTERNAL LOOPBACK MODE  
 SW02=1 EXTERNAL LOOPBACK MODE  
 SW01=1 ONE-WAY-IN MODE  
 SW00=1 ONE-WAY-OUT MODE

THIS PROGRAM HAS BEEN MODIFIED TO RUN ON A PROCESSOR WITH OR WITHOUT A HARDWARE SWITCH REGISTER. WHEN FIRST EXECUTED THE PROGRAM TESTS THE EXISTENCE OF A HARDWARE SWITCH REGISTER. IF NOT FOUND A SOFTWARE SWITCH REGISTER LOCATION (SWREG=LOC. 176 ) IS DEFAULTED TO. IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SWREG ARE DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(IE) SWR=XXXXXX NEW=

POSSIBLE RESPONSES ARE:

1.      <CR>      IF NO CHANGES ARE TO BE MADE
2.      6 DIGITS 0-7      TO REPRESENT IN OCTAL THE NEW SWITCH REGISTER VALUE ;LAST DIGIT FOLLOWED BY <CR>.
3.      ↑U      TO ALLOW REENTERING VALUE IF ERROR IS COMMITTED KEYING IN SWREG VALUE.

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE CONTENTS OF SWREG DURING PROGRAM EXECUTION. BY STRIKING ↑G (CNTRL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE THE CONTENTS OF SWREG, WHICH IS PROCESSED IN KEY AREAS OF THE PROGRAM CODE (IE) ERROR ROUTINES, AFTER HALTS END OF PASS, AND OTHER APPLICABLE AREAS.

IF OPERATOR SPECIFIED DATA WAS INDICATED, THE PROGRAM WILL TYPE A REQUEST FOR THE DATA. DATA MAY BE ENTERED AS ASCII CHARACTERS OR OCTAL CODE. TYPE IN THE DATA TERMINATED WITH A CR. OCTAL CODE MAY BE ENTERED BY TYPING AN ↑(UP ARROW) FOLLOWED BY THE OCTAL CODE (IN THE RANGE 000 TO 377) SEPERATED BY SPACES AND TERMINATED BY ↑(UP ARROW).  
I.E. ABCD↑ 000 123 377↑ EFG (CAR.RETURN)

A TYPICAL SWITCH SETTING FOR HALF-DUPLEX=003150 THIS SETTING USES INTERNAL LOOPBACK MODE, LOOPS IN OVERLAY, MONITORS TRANSMITTED AND RECEIVED DATA ON THE CONSOLE TTY, AND TESTS RECEIVED DATA USING TEST MESSAGE #3.

A TYPICAL SWITCH SETTING FOR FULL-DUPLEX=003144 THIS SETTING IS THE SAME AS ABOVE EXCEPT IT USES THE EXTERNAL LOOPBACK MODE.

ALL STANDARD MESSAGES (TEST MESSAGES 1-3) ARE PRECEDED BY 2 FILL CHARACTERS(177), AND ARE FOLLOWED BY A CR(015), LF(012), RECEIVE TERMINATING CHARACTER(001), 4 FILLS(177), AND A TRANSMIT TERMINATING CHARACTER(000). DURING TRANSMISSION, WHEN A 000 CHARACTER IS SEEN THE TRANSMISSION IS STOPPED. DURING RECEPTION, WHEN A 001 CHARACTER IS RECEIVED, THE RECEIVER IS SHUT OFF. IF THE MESSAGE WAS INPUTED BY THE OPERATER, THE TERMINATING CHARACTERS ARE ADDED.

## TEST MODES

## INTERNAL LOOPBACK MODE

1. THE OVERLAY WAITS TO RECEIVE A MESSAGE (TERMINATED BY <001>)
2. VERIFIES THE DATA AGAINST THE DATA SELECTED BY SW09 AND SW10 (SW7=0)
3. TRANSMIT THE DATA SELECTED BY SW09 AND SW10 (SW8=0) OR TRANSMIT THE RECEIVED DATA (SW8=1)
4. RETURNS TO MONITOR FOR "END PASS" (SW4=1) OR GO TO STEP 1. (SW4=0)

## EXTERNAL LOOPBACK MODE

1. THE OVERLAY SETS REQUEST TO SEND
2. WAIT FOR CLEAR TO SEND
3. TRANSMITS THE SELECTED DATA
4. RESETS REQUEST TO SEND
5. WAIT FOR MESSAGE TO BE RECEIVED
6. VERIFIES THE DATA (SW07=0)
7. RETURNS TO MONITOR FOR "END PASS". (SW04=1) OR GO TO STEP 1 (SW04=0)

## ONE-WAY-IN MODE

1. THE OVERLAY WAITS FOR MESSAGE TO BE RECEIVED.
2. VERIFIES THE DATA (SW07=0)
3. RETURNS TO MONITOR FOR "END PASS" (SW04=1) OR GO TO STEP 1 (SW04=0)

## ONE-WAY-OUT MODE

1. THE OVERLAY SETS REQUEST TO SEND
2. WAITS FOR CLEAR TO SEND
3. TRANSMITS SELECTED DATA
4. RETURNS TO MONITOR FOR "END PASS". (SW04=1) OR GO TO STEP 1 (SW04=0)

- E. THE OVERLAY IS THEN ENTERED AND A CONNECTION ESTABLISHED EITHER MANUALLY OR AUTOMATICALLY.

IF ONE-WAY-IN OR INTERNAL LOOPBACK MODES ARE SELECTED.  
THE OVERLAY WILL SET DATA TERMINAL READY AND WAIT FOR DATA.

IF ONE-WAY-OUT OR EXTERNAL LOOPBACK MODES WERE SELECTED.  
THE OVERLAY WILL SET DATA TERMINAL READY AND REQUEST TO SEND.  
THE OVERLAY WILL THEN WAIT FOR CLEAR TO SEND BEFORE ATTEMPTING TO TRANSMIT DATA.

THE PROGRAM WILL PRINTOUT A "WAITING FOR CLEAR TO SEND" MESSAGE AND THE CONTENTS OF THE XMIT CSR EVERY 60 SECS. UNTIL CLEAR TO SEND IS ASSERTED.



F. IF SW04=0 THE OVERLAY WILL CONTINUE TO TRANSMIT/RECEIVE DATA.

IF SW04=1 THE OVERLAY WILL RETURN TO THE MONITOR AND TYPE "END PASS".

IF BOTH SW04=1 AND SW14=1, THE PROGRAM WILL REQUEST NEW INTERFACE PARAMS AFTER ONE PASS OF THE SELECTED TEST MODE.

TEST EXECUTION MAY BE INTERRUPTED BY TYPING THE FOLLOWING CHARACTERS ON THE CONSOLE TTY.

LINE FEED = RESTART PROGRAM AT LOCATION 200.

QUESTION MARK = PRINTOUT FIRST 8 WORDS OF INPUT BUFFER. (ASCII)

THEN TYPE EITHER:

#XXXXXX      TO PRINTOUT THE 8 WORDS AT LOC XXXXXX.

#BXXXXXX      TO PRINTOUT THE 16 BYTES AFTER LOC XXXXXX.

#C      TO CONTINUE

PROGRAM MUST BE RESTARTED AT 200 AFTER PRINTING.

CARRIAGE RETURN = RESTART AT REQUEST FOR NEW OPERATIONAL SWITCHES.

## 5.0 PROGRAM AND/OR OPERATOR ACTION

IF THE OPERATOR WISHES TO MANUALLY EXAMINE THE TRANSMIT OR RECEIVE BUFFERS, DO THE FOLLOWING: TO FIND THE STARTING ADDRESS OF THE RECEIVE BUFFER, LOAD ADDRESS 11020 AND EXAMINE. TO FIND THE STARTING ADDRESS OF THE TRANSMIT BUFFER, LOAD ADDRESS 11022 AND EXAMINE.

### 5.1 NORMAL HALTS SEE SECTION 4.

## 6.0 ERRORS

### 6.1 ERROR REPORTING

THE ONLY ERROR REPORT FROM THE CONTROL PROGRAM OCCURS IF THE INTERFACE SPECIFIED IS NOT LOADED.

IF DATA IS RECEIVED AND SWITCH 7 (NO DATA COMPARE) IS RESET, THE DATA WILL BE COMPARED AGAINST THE PRESELECTED DATA AFTER A LINE FEED CHARACTER IS RECEIVED. IF THERE IS A MISMATCH, THE FOLLOWING ERROR REPORT IS PRINTED:

RECEIVED DATA=RRRRRR  
DATA SHOULD BE TTTTTT  
DATA COMPARE ERROR; BAD DATA=BBB GOOD DATA=GGG

WHERE RRRRRR IS THE RECEIVE BUFFER (UP TO 512 CHARACTERS)  
TTTTTT IS THE TRANSMIT BUFFER (UP TO 512 CHARACTERS)  
BBB IS THE BAD DATA CHARACTER  
GGG IS THE GOOD DATA CHARACTER

IF THE INTERFACE DETECTS A DATA ERROR, THE FOLLOWING  
WILL BE PRINTED BEFORE THE DATA IS COMPARED:

THERE WAS A RECEIVER ERROR. RECEIVER DATA REGISTER =XXXXXX

WHERE XXXXXX IS THE CONTENTS OF THE RECEIVER DATA REGISTER  
THE LOW BYTE IS THE DATA, AND THE HIGH BYTE IS THE ERROR BITS.

IF A RECEIVE TERMINATING CHARACTER(DGI) IS NOT DETECTED  
WITHIN 512 CHARACTERS A "BUFFER FULL" PRINTOUT WILL OCCUR.

## 7.0 RESTRICTIONS

THE OPERATION OF THIS PROGRAM REQUIRES COORDINATION BETWEEN  
THE OPERATOR AND THE OPERATOR OF ANOTHER PDP-11 SYSTEM  
UNLESS ONE OF THE SYSTEMS IS ALWAYS OPERATING IN A FIXED  
MODE. THE FOLLOWING TABLE LISTS THE VALID COMBINATIONS:

CPU #1	CPU #2
ONE-WAY-OUT	ONE-WAY-IN
ONE-WAY-IN	ONE-WAY-OUT
EXTERNAL-LOOPBACK	INTERNAL-LOOPBACK
INTERNAL-LOOPBACK	EXTERNAL-LOOPBACK
EXTERNAL-LOOPBACK	EXTERNAL-LOOPBACK (FULL DUPLEX)

WHEN THE COMMUNICATION LINK INVOLVES MODEMS THE FOLLOWING  
RESTRICTIONS APPLY:

IF RUNNING IN FULL DUPLEX MODE BOTH SYSTEMS  
MUST BE IN EXTERNAL LOOP BACK MODE.

BOTH SYSTEMS SHOULD BE RUNNING IDENTICAL ROUTINES.

EXAMPLE:  
SWITCHES 14,13,7,4 SHOULD BE THE SAME  
ON BOTH CPU S

IF PROGRAM IS WAITING IN A SCAN ROUTINE AND TYPES OUT  
A "WAITING MESSAGE", IF AN INCOMING MESSAGE STARTS DURING  
THE TYPE OUT, IT WILL BE LOST BECAUSE THE TYPEOUT PRIORITY  
IS AT LEVEL 7. THIS WILL RESULT IN OVERRUN OR SILO OVER-  
RUN ERRORS, DEPENDING ON THE DEVICE. TO AVOID THIS SITUATION  
RUN WITH SWITCH 13 UP. IF OVERRUN DOES OCCURE DURING A  
TYPEOUT THE PROGRAM SHOULD BE RESTARTED.

IF USING AN ASYNCHRONOUS DEVICE, MODEMS AND THE  
MAYNARD TEST STATION AND INITIALIZE DOES NOT CLEAR THE  
CONNECTION (EXAMPLE THE DJ11) IF THE PROGRAM IS RESTARTED  
IN THE MIDDLE OF A MESSAGE AT LOC 204 OR BY HITTING CR  
AN IMMEDIATE ERROR MESSAGE FROM MAYNARD WILL BE RE-

CEIVED. THIS IS BECAUSE THE TEST STATION IS STILL LOOKING FOR THE REST OF THE INTERRUPTED MESSAGE. TO AVOID THIS ERROR, RESTART PROGRAM ONLY AT THE END OF THE MESSAGE CURRENTLY BEING TRANSMITTED.

## 8.0 MISCELLANEOUS

ITEP WAS CHECKED OUT USING THE FOLLOWING BELL TELEPHONE MODEMS.  
201A (HALF-DUPLEX SYNCHRONOUS 2000 BAUD)  
202C (HALF-DUPLEX ASYNCHRONOUS 1200 BAUD)  
103A (FULL-DUPLEX ASYNCHRONOUS 110 BAUD)

## 9.0 PROGRAM DESCRIPTION

9.1 THE DU11 INTERFACE SERVICE PARAMS ARE SETUP, AS SPECIFIED BY THE OPERATOR, BY THE ITEP CONTROL PROGRAM.

TIME: PROVIDES A MEANS OF MEASURING ELAPSED TIME. IT IS INCREMENTED EVERY SECOND BY A CLOCK INTERRUPT ROUTINE IN ITEP.

9.2 WHEN THE OVERLAY IS FIRST ENTERED BY ITEP AT LOCATION START, THE CONTENTS OF THE SWITCH REGISTER ARE STORED IN REGISTER 0. THE MODE AND DATA SELECTIONS ARE FIXED AT THIS TIME AND CANNOT BE ALTERED WITHOUT RETURNING TO THE CONTROL PROGRAM. THE INTERRUPT VECTORS AND VARIABLES ARE THEN SETUP. THE SELECTED ROUTINE DETERMINED BY THE MODE IS THEN ENTERED

9.3 THE OVERLAY THEN LOOPS IN ROUTINES: \$OWI, IF "ONE WAY IN" MODE WAS SELECTED. \$OWO, IF "ONE WAY OUT" MODE WAS SELECTED. \$ILB, IF "INTERNAL LOOP BACK" MODE WAS SELECTED. \$XLB, IF "EXTERNAL LOOP BACK" WAS SELECTED.

9.31 \$OWI: IN THIS ROUTINE THE RECEIVER IS INITIALIZED AND PROGRAM LOOPS WAITING FOR THE RECEIVER TO FINISH. IF NOTHING IS RECEIVED FOR 60 SECS A "WAITING" MESSAGE IS TYPED. WHEN THE RECEIVER IS DONE, THE PROGRAM CHECKS DATA IF SWITCHES PERMIT, AND TYPES END PASS DEPENDING ON SWITCH SETTINGS.

9.32 \$OWO: THE TRANSMITTER IS INITIALIZED AND PROGRAM LOOPS WAITING FOR TRANSMITTER TO FINISH. A "WAITING" MESSAGE IS TYPED EVERY 60 SECS IF THERE IS NO ACTION. WHEN THE TRANSMITTER IS DONE, THE PROGRAM EITHER LOOPS BACK TO \$OWO OR TYPES END PASS DEPENDING ON SWITCH SETTINGS.

9.33 \$ILB: THE RECEIVER IS INITIALIZED AND PROGRAM LOOPS WAITING FOR RECEIVER TO FINISH, A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF NO ACTION. WHEN RECEIVER IS DONE PROGRAM CHECKS DATA IF SWITCH SETTINGS PERMIT, AND END PASS IS TYPED IF SWITCH SETTINGS PERMIT. THEN THE TRANSMITTER IS INITIALIZED, A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF NO ACTION. WHEN TRANSMITTER IS DONE PROGRAM RETURNS TO START OF ROUTINE. (\$ILB)

9.34 \$XLB: IF IN HALF DUPLEX THE TRANSMITTER IS INITIALIZED, A "WAITING MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO ACTION

WHEN THE TRANSMITTER IS DONE THE RECEIVER IS INITIALIZED  
A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO ACTION.  
WHEN THE RECEIVER IS DONE DATA IS CHECKED IF SWITCH SETTINGS  
PERMIT AND END PASS IS TYPED IF SWITCHES ALLOW. THE PROGRAM NOW  
REPEATS CYCLE STARTING AT \$XL9.  
IF IN FULL DUPLEX THE RECEIVER AND TRANSMITTER ARE INITIALIZED  
A "WAITING" MESSAGE IS TYPED EVERY 60 SEC IF THERE IS NO  
ACTION. WHEN BOTH THE RECEIVER AND TRANSMITTER ARE DONE DATA IS  
CHECKED, END PASS IS TYPED AND PROGRAM LOOPS TO \$XLB DEPENDING  
ON THE SWITCH SETTINGS.

- 9.4 THE RETURN TO MONITOR ROUTINE FOR END PASS AT EOP:  
LOCKS OUT INTERRUPTS AND SAVES THE TRANSMITTER INTERRUPT ENABLE  
BIT AND ALL GENERAL REGISTERS. IT THEN RETURNS TO THE MONITOR  
TO TYPE "END PASS". THE MONITOR CHECKS SW14 IF UP IT RETURNS  
TO ENTER:, OTHERWISE IT RESTARTS THE PROGRAM.
- 9.5 ENTER: IS ENTERED FROM THE MONITOR AFTER TYPEING "END PASS",  
IT RESTORES THE GENERAL REGISTERS AND THE TRANSMITTER CSR  
AS SAVED IN EOP. THE DELAY FLAG IS SET AND PROGRAM RETURNS TO  
THE SCAN ROUTINE (OWO, OWI, ILB, ALB) WHERE IT CAME FROM.
- 9.6 THE INITIALIZE TRANSMIT SUBROUTINE AT STARTX:  
SETS UP THE INTERFACE AND POINTERS NECESSARY TO  
INITIATE A TRANSMIT OPERATION.  
AFTER SETTING "DATA TERMINAL READY" AND "REQUEST TO SEND" A CHECK  
IS MADE ON PARAM2 TO DETERMINE IF HALF DUPLEX OPERATION  
WAS SELECTED BY THE OPERATOR. IF IT WAS, THE  
SUBROUTINE WAITS FOR CLEAR TO SEND.  
A 'WAITING FOR CLEAR TO SEND' PRINTOUT OCCURS  
EVERY 30 SECONDS UNTIL CLEAR TO SEND IS ASSERTED.
- 9.7 THE INITIALIZE RECEIVED SUBROUTINE AT STARTR:  
SETS UP THE INTERFACE AND POINTERS NECESSARY TO  
RECEIVE A MESSAGE.
- 9.8 THE TRANSMIT INTERRUPT SERVICE ROUTINE,  
AT XISR:, IS ENTERED VIA TRANSMIT INTERRUPTS  
FROM THE INTERFACE.  
A TEST IS MADE TO SEE IF THE LAST CHARACTER  
TRANSMITTED WAS A NULL (ALL ZEROS) CHARACTER.  
IF IT WAS: THE TRANSMIT LOGIC IN THE INTERFACE  
IS RESET AND THE TRANSMIT COMPLETE FLAG IS SET.  
AT XISR1: THE NEXT CHARACTER IS TRANSMITTED  
AND PRINTED ON THE TTY IF THE MONITOR TRANSMIT  
SWITCH IS SET.
- 9.9 THE RECEIVE INTERRUPT SERVICE ROUTINE  
AT RISR: IS ENTERED VIA RECEIVER INTERRUPTS  
FROM THE INTERFACE.  
THE RECEIVED CHARACTER IS STORED IN  
THE INPUT BUFFER AND PRINTED ON THE TTY IF  
THE MONITOR RECEIVER SWITCH IS SET.  
IF THE INPUT BUFFER IS FULL, A 'BUFFER FULL'  
PRINTOUT WILL OCCUR. THIS INDICATES THAT A  
LINE FEED CHARACTER WAS NOT RECOGNIZED

IN THE RECEIVED DATA (WITHIN 1000 CHARACTERS).  
IF THE RECEIVED CHARACTER IS A LINE FEED,  
THE RECEIVED LOGIC IS RESET AND THE  
RECEIVE COMPLETE FLAG IS SET.  
IF A 'RECEIVE ERROR' IS DETECTED AT RISR:, THE  
CSR AND DBR WILL BE SAVED AND PRINTED OUT  
AFTER THE COMPLETE MESSAGE HAS BEEN RECEIVED.

- 9.10 THE DATA TEST SUBROUTINE AT TESTD: IS  
ENTERED AFTER A COMPLETE MESSAGE HAS BEEN  
RECEIVED.  
IF A 'RECEIVE ERROR' HAD BEEN DETECTED,  
THE CONTENTS OF THE 'RECEIVE BUFFER' AT THE  
TIME THE ERROR OCCURRED WILL BE PRINTED.  
THE DATA IS COMPARED UNTIL A 'ALL ZEROS'  
CHARACTER IS RECOGNIZED. 'FILL' (ALL ONES)  
CHARACTERS ARE IGNORED. IF A MISMATCH  
IS DETECTED, THE COMPLETE CONTENTS OF THE  
INPUT BUFFER AND GOOD DATA IS PRINTED.

#### 10.0 PARAMETERS FOR THE DU11

PARAM#1 (LOW BYTE) IS LOADED INTO THE TRANSMIT STATUS REGISTER (TXCSR).  
BIT 3      HALF DUPLEX (1), DEFAULT= HALF DUPLEX (1)

PARAM#1 (HIGH BYTE) IS LOADED INTO THE RECEIVER STATUS REGISTER (RXCSR).  
BIT 8      STRIP SYNC (1), DEFAULT= STRIP SYNC (1)

PARAM#2 IS LOADED INTO THE PARAMETER STATUS REGISTER (PARCSR).  
BITS 0-7      SYNC CHARACTER, DEFAULT= 26 (26)  
BIT 8      PARITY SENSE SELECT (1), DEFAULT= (0)  
BIT 9      PARITY ENABLE (1), DEFAULT= (0)  
BITS 10,11      WORD LENGTH SELECT, DEFAULT= 8 BITS (11)  
BITS 12,13      MODE SELECT, DEFAULT= INTERNAL SYNCHRONOUS (11)

PARAM#3 IS NOT USED (177777).

```

580
581
582
583
584      011000
585 011000 052574 000040
586 011004 1F0040
587 011006 000300
588 011010 000240
589 011012 000410
590 011014 036026
591 011018 177777
592 011020 000000
593 011022 000000
594 011024 000077
595 011026 000000
596 011030 000000
597 011032 000000
598 011034 000000
599 011036 011102
600 011040
601 011040      000
602 011041
603 011041      001
604 011042 000000
605 011044 177570
606 011046 177570
607
608
609
610
611      000000
612      100000
613      040000
614      020000
615      020000
616
617 011050 000000
618 011052 000000
619 011054 000000
620 011056 000000
621 011060 000000
622
623 011062 000000
624 011064 000000
625 011066 000000
626 011070 000000
627
628 011072 177560
629 011074 177562
630 011076 177564
631 011100 177566
632
633      000001

```

```

*****
;
;      DUII INTERFACE SERVICE PARAMS
;
*****

```

```

DU11:  .=11000
BA:    .ASCIZ /DU /
RIV:   300
PRIOR: 240
PARAM1: STPSYN!HOXEN
PARAM2: SYNINT!EIGHT!NOPAR!26
PARAM3: 177777
IRDA:  .WORD 0
IXDA:  .WORD 0
SETTLE: .WORD 0
B2016: .WORD 0
TIME:  .WORD 0
TX.TERM: .WORD START
RX.TERM: .WORD 0
FLAG:   .WORD 0
SWR:    177570
DISPLAY:177570

```

```

;ISR NAME
;BUS ADDRESS
;VECTOR ADDRESS
;PRIORITY
;PARAM #1
;PARAM #2
;PARAM #3
;INITIAL READ DATA ADDRESS
;INITIAL XMIT DATA ADDRESS
;LINE SETTLE DELAY FLAG
;ADDR OF BIN TO OCT TYPE ROUTINE
;TIMER
;ADDR OF START OF PROGRAM
;TRANSMITTER TERMINATING CHAR.
;RECEIVER TERMINATING CHAR.

```

```

*****
;
;      CONSTANTS + WORKING STORAGE
;
*****

```

```

STAT=RO
XFLG=100000
RFLG=40000
DSFLG=20000
BIT13=20000
SXCSR: 0
SRCSR: 0
ERCSR: 0
ERDBR: 0
DSSTAT: 0
XCC: 0
XDR: 0
TKS: 177560
TKB: 177562
TPS: 177564
TPB: 177566
FULL.DUPLEX=000001

```

```

;XMIT COMPLETE FLAG
;RCV COMPLETE FLAG
;DATA SET STATUS CHANGE FLAG
;INHIBIT PRINTOUTS
;SAVED XMIT CSR
;SAVED RCV CSR
;RCV CSR SAVED ON ERROR
;RCV DATA REG SAVED ON ERROR
;RCV CSR SAVED ON DS CHANGE
;XMIT CHAR COUNT
;RCV CHAR COUNT
;RCV DATA ADDR.
;XMIT DATA ADDR.

```

```

634
635
636
637 011102 000240
638 011104 017700 177734
639 011110 042700 177400
640 011114 013702 011006
641 011120 012722 013572
642 011124 013722 011010
643 011130 012722 013474
644 011134 013722 011010
645 011140 013704 011004
646 011144 013714 011012
647 011150 013702 011014
648 011154 042702 000001
649 011160 010264 000004
650 011164 052764 000400 000004
651 011172 113764 011012 000004
652 011200 113764 C.1013 000001
653 011206 013764 011014 000002
654 011214 004537 013770
655 011220 011014
656
657
658
659
660
661
662
663 011222 005037 011032
664 011226 005037 013122
665 011232 005037 013126
666 011236 032700 000001
667 011242 001402
668 011244 000137 011420
669 011250 032700 000002
670 011254 001402
671 011256 000137 011312
672 011262 032700 000010
673 011266 001402
674 011270 000137 011516
675 011274 032700 000004
676 011300 001402
677 011302 000137 011746
678 011306 000000
679 011310 000776
680
681
682
683
684
685
686
687
688
689

```

```

*****
DUII-X INTERFACE SERVICE ROUTINE
*****
START:  NOP
        MOV     JSR,   R0      ;SETUP MODE IN R0
        BIC     #177400, R0   ;STRIP JUNK
        MOV     RIV,  R2      ;SETUP
        MOV     BRISR, (R2)+  ;INTERRUPT
        MOV     PRIOR, (R2)+  ;VECTORS
        MOV     BXISR, (R2)+
        MOV     PRIOR, (R2)+
        MOV     BA,    R4      ;SETUP BUS ADDR INDEX
        MOV     PARAM1, PARCSR ;SETUP VARIABLES
        MOV     PARAM2, R2
        BIC     #0001, R2
        MOV     R2,     XCSR(R4) ;IN CSR'S
        BIS     #RESET, TXCSR(R4) ;MASTER RESET
        MOV     PARAM1, TXCSR(R4) ;SET UP VARIABLES
        MOV     PARAM1+1, 1(R4) ;LOAD UPPER BYTE RXCSR
        MOV     PARAM2, PARCSR(R4) ;LOAD PARCSR
        JSR     RS,STRIP      ;GO CALC PARITY
        PARAM2

```

```

*****
ROUTINE USED TO GOTO
SUBROUTINE DEPENDENT
ON MODE SELECTED.
*****

```

```

GO:    CLR     TIME
        CLR     DELAY
        CLR     STOP
        BIT     #0W0, MODE
        BEQ    1$
        JMP    $0W0
1$:    BIT     #0W1, MODE
        BEQ    2$
        JMP    $0W1
2$:    BIT     #1LB, MODE
        BEQ    3$
        JMP    $1LB
3$:    BIT     #XLB, MODE
        BEQ    4$
        JMP    $XLB
4$:    HALT
        BR     .-2

```

```

*****
ROUTINE USED IF "ONE WAY IN" MODE WAS SELECTED.
NOTE THAT WHEN IN THIS MODE HALF DUPLEX IS THE
ONLY MODE AVAILABLE.
"ONE WAY IN" MEANS THAT ONLY THE RECEIVER IS
ENABLED. THE TRANSMITTER IS NEVER "TURNED ON".
*****

```

```

690 ;*****:*****
691
692
693
694 011312 104416 SOWI: KBDIN
695 011314 004737 013400 JSR PC,STARTR
696 011320 032700 040000 15: BIT #RFLG,STAT
697 011324 001013 BNE 25
698 011325 023727 011032 000100 CMP TIME,#100
699 011334 103771 BLO 15
700 011336 011402 MOV #RCSR,R2
701 011340 016403 000004 MOV XCSR(R4),R3
702 011344 104001 HLT 1
703 011346 005037 011032 CLR TIME
704 011352 000762 BR 15
705
706 011354 032777 000200 177462 25: BIT #MODAT,#SWR
707 011362 001002 BNE 35
708 011364 004737 012336 JSR PC,TESTD
709 011370 042700 040000 35: BIC #RFLG,STAT
710 011374 032777 000020 177442 BIT #LOOP,#SWR
711 011402 001405 BEQ 45
712 011404 012737 011416 013124 MOV #45,BACK
713 011412 000137 012176 JMP EOP
714 011416 000735 45: BR SOWI
715
716
717
718
719
720
721
722
723
724

```

```

*****
ROUTINE USED IF "ONE WAY OUT" WAS SELECTED.
NOTE THAT WHEN IN THIS MODE HALF DUPLEX IS THE ONLY
MODE AVAILABLE.
"ONE WAY OUT" MEANS THAT ONLY THE TRANSMITTER IS
ENABLED. THE RECEIVER IS NEVER "TURNED ON."
*****

```

```

725 011420 104416 SOWO: KBDIN
726 011422 004737 013130 JSR PC,STARTX
727 011426 005037 011032 CLR TIME
728 011432 032700 100000 15: BIT #XFLG,STAT
729 011436 001013 BNE 25
730 011440 023727 011032 000100 CMP TIME,#100
731 011446 103771 BLO 15
732 011450 011402 MOV #RCSR,R2
733 011452 016403 000004 MOV XCSR(R4),R3
734 011456 104001 HLT 1
735 011460 005037 011032 CLR TIME
736 011464 000762 BR 15
737 011466 042700 100000 25: BIC #XFLG,STAT
738 011472 032777 000020 177344 BIT #LOOP,#SWR
739 011500 001405 BEQ 35
740 011502 012737 011514 013124 MOV #35,BACK
741 011510 000137 012176 JMP EOP
742 011514 000741 35: BR SOWO
743
744
745

```





```

801
802
803
804
805
806
807
808
809
810
811
812
813
814 011746 104416
815 011750 032737 000010 011012
816 011756 001002
817 011760 004737 013400
818 011764 004737 013130
819 011770 005037 011032
820 011774 032700 100000
821 012000 001016
822 012002 032700 040000
823 012006 001024
824 012010 023727 011032 000100
825 012016 103766
826 012020 011402
827 012022 016403 000004
828 012026 104001
829 012030 005037 011032
830 012034 000757
831 012036 032737 000010 011012
832 012044 001756
833 012046 042700 100000
834 012052 004737 013400
835 012056 000746
836 012060 032737 000010 011012
837 012064 001020
838 012070 032700 100000
839 012074 001013
840 012076 023727 011032 000100
841 012106 103766
842 012110 011402
843 012114 016403 000004
844 012118 104001
845 012122 005037 011032
846 012126 000757
847 012130 042700 100000
848 012134 042700 040000
849 012138 005037 011032
850 012140 032777 000200 176676
851 012146 001002
852 012150 004737 012336
853 012154 032777 000020 176662
854 012162 001671
855 012164 012737 011746 013124
856 012172 000137 012176

```

```

*****
ROUTINE USED IF "EXTERNAL LOOP BACK" WAS SELECTED.
EITHER HALF OR FULL DUPLEX MAY BE SELECTED IN THIS MODE.
"EXTERNAL LOOP BACK" MEANS THAT THE TRANSMITTER IS FIRST
TURNED ON (IF HALF DUPLEX) AND THE WHOLE MESSAGE IS TRANSMITTED;
THEN THE RECEIVER IS ENABLED. AFTER THE WHOLE MESSAGE IS RECEIVED
DATA WILL THEN BE CHECKED IF DESIRED AND END PASS WILL
BE GIVEN IF DESIRED. THEN THE CYCLE IS REPEATED
AS ABOVE. IF RUNNING IN FULL DUPLEX THE PROGRAM
WAITS FOR BOTH THE RECEIVER AND TRANSMITTER TO
FINISH THEN RESTARTS THE RECEIVER AND TRANSMITTER.
*****

```

```

$XLB: KBOIN
      BIT #HALF.DUPLEX,PARAM1
      BNE 18
      JSR PC,STARTR
18:   JSR PC,STARTX
      CLR TIME
28:   BIT #XFLG,STAT
      BNE 38
78:   BIT #AFLG,STAT
      BNE 48
      CMP TIME,#100
      BLO 28
      MOV @RCSR,R2
      MOV XCSR(R4),R3
      HLT 1
      CLR TIME
      BR 28
38:   BIT #HALF.DUPLEX,PARAM1
      BEQ 78
      BIC #XFLG,STAT
      JSR PC,STARTR
      BR 28
48:   BIT #HALF.DUPLEX,PARAM1
      BNE 88
      BIT #XFLG,STAT
      BNE 68
      CMP TIME,#100
      BLO 48
      MOV @RCSR,R2
      MOV XCSR(R4),R3
      HLT 1
      CLR TIME
      BR 48
68:   BIC #XFLG,STAT
88:   BIC #AFLG,STAT
      CLR TIME
      BIT #MODAT,@SWR
      BNE 58
      JSR PC,TESTD
58:   BIT #LOOP,@SWR
      BEQ $XLB
      MOV #XLB,BACK
      JMP EOP

```

```

857
858
859
860
861
862
863 012176
864 012176 104414 000340
865 012202 016437 000004 012334
866 012210 042737 177677 012334
867 012216 042764 000100 000004
868 012224 012766 012264 000002
869 012232 010037 013106
870 012236 010137 013110
871 012242 010237 013112
872 012246 010337 013114
873 012252 010437 013116
874 012256 010537 013120
875 012262 000207
876
877 012264
878 012264 013700 013106
879 012270 013701 013110
880 012274 013702 013112
881 012300 013703 013114
882 012304 013704 013116
883 012310 013705 013120
884 012314 012737 177777 013122
885 012322 053764 012334 000004
886 012330 000177 000570
887 012334 000000
888
889
890
891
892
893
894
895 012336 013746 011056
896 012342 011413
897 012344 032777 020000 176472
898 012352 001007
899 012354 104400 012546
900 012360 044077 176444
901 012364 000746
902 012366 104400 012627
903 012372 013701 011032
904 012376 013702 011020
905 012402 122122
906 012404 001776
907 012406 123741 011040
908 012412 001453
909 012414 122742 000002
910 012420 001005
911 012422 010237 012430
912 012426 104400

```

```

*****
ROUTINE TO RETURN
TO MONITOR FOR
END PASS.
*****

```

```

EOP:
STPS,PRTY7 ;SET PS PRIORITY TO 7
MOV XCSR(R4),@TPIE ;SAVE TX CSR
BIC @TPIE,@TPIE ;CLEAR ALL BUT TX IE
BIC @TIE,XCSR(R4) ;CLEAR TX IE (EVEN IF IT WASN'T SET)
MOV ENTER,2(SP) ;SET FOR RETURN IF SW 14=1
MOV R0,SAVR0 ;SAVE REGISTER 0
MOV R1,SAVR1 ;SAVE REGISTER 1
MOV R2,SAVR2 ;SAVE REGISTER 2
MOV R3,SAVR3 ;SAVE REGISTER 3
MOV R4,SAVR4 ;SAVE REGISTER 4
MOV R5,SAVR5 ;SAVE REGISTER 5
RTS PC ;RETURN TO CONTROL PROGRAM

```

```

ENTER:
MOV SAVR0,R0 ;RESTORE R0
MOV SAVR1,R1 ;RESTORE R1
MOV SAVR2,R2 ;RESTORE R2
MOV SAVR3,R3 ;RESTORE R3
MOV SAVR4,R4 ;RESTORE R4
MOV SAVR5,R5 ;RESTORE R5
MOV @-1,DELAY
BIS @TPIE,XCSR(R4) ;IF ORIGINALLY SET; SET TX IE
JMP @BACK
@TPIE: 000000

```

```

*****
SUBROUTINE TO CHECK
RECEIVER DATA.
*****
TESTD: MOV ER,9,-(SP) ;WAS THERE A RECEIVE ERROR?
BEQ TSTDAT ;BR IF NO
BIT @BIT13,@SWR ;INHIBIT PRINTOUTS?
BNE TSTDAT ;BR IF YES
TYPE MSG0 ;(15)<(12) THERE WAS A RECEIVE ERROR. RBUF=
JSR R0,-2016 ;PRINT CONTENTS OF RBUF
TST -(SP)
TYPE MSG1 ;(15)<(12)
TSTDAT: MOV IXC@,R1 ;SETUP XMIT DATA ADDR
MOV IRCA,R2 ;SETUP RCV DATA ADDR
SCAN4: CMPB (R1)+,(R2)+ ;DATA OK?
BEQ SCAN4 ;BR IF OK
CMPB TX.TERM,-(R1) ;IS IT END OF DATA
BEQ TESTDX ;BR IF YES
CMPB #002,-(R2)
BNE Z$
MOV R2,1$
TYPE

```

```

913 012430 000000 15: .WORD 0
914 012432 000443 BR TESTDX
915 012434 105712 25: TSTB (R2)
916 012436 001441 BEQ TESTDX ;BR IF YES
917 012438 122721 000177 CMPB #177, (R1)+ ;IS IT FILL CHAR?
918 012440 001756 BEQ SCAN4 ;BR IF YES
919 012442 005301 DEC R1 ;BACKUP
920 012444 122722 000177 CMPB #177, (R2)+ ;IS IT FILL?
921 012446 001756 BEQ SCAN4 ;BR IF YES
922 012448 105712 TSTB -(R2) ;BACK UP POINTER
923 012450 123722 011014 CMPB PARAM2, (R2)+
924 012452 000240 BEQ SCAN4 ;BR IF CHAR WAS SYNC
925 012454 000240 SCANS: NOP ;DATA ERROR
926 012456 032777 020000 176346 BIT #BIT13, @SWR ;INHIBIT PRINTOUTS
927 012458 001016 BNE DERR ;BR IF YES
928 012460 104400 012632 TYPE MSG2 ;<15><12>RECEIVED DATA = <15><12>
929 012462 013737 011020 012514 MOV IRDA, RDX ;SETUP DATA ADDRESS
930 012464 104400 TYPE ;PRINT RECEIVED DATA
931 012466 000000 RDX: 0 ;RECEIVED DATA ADDR.
932 012468 104400 012657 TYPE MSG3 ;<15><12>DATA SHOULD BE<15><12>
933 012470 013737 011022 012532 MOV IXDA, .+10 ;SETUP ADDR.
934 012472 104400 TYPE ;PRINT GOOD DATA
935 012474 011022 IXDA
936 012476 111103 DERR: MOVB (R1), R3 ;SETUP XMIT DATA
937 012478 114202 MOVB -(R2), R2 ;SETUP RCV DATA
938 012480 104007 HLT+7 ;DATA ERROR HALT
939 012482 005726 TESTDX: TST (SP)+ ;POP STACK
940 012484 003207 RTS ;RETURN FROM SUB/ROUT
941 012546 005015 044124 051105 MSG0: .ASCIZ <15><12>/THERE WAS A RECEIVER ERROR. REGISTER (SEL 2) =/
(1) 012527 015 000012 MSG1: .ASCIZ <15><12>
(1) 012632 005015 042522 042503 MSG2: .ASCIZ <15><12>/RECEIVED DATA = /<15><12>
(1) 012657 015 042012 052101 MSG3: .ASCIZ <15><12>/DATA SHOULD BE/<15><12>
(1) 012702 005015 046120 040505 MSG4: .ASCIZ <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./
(1) 012751 015 053412 042510 .ASCIZ <15><12>/WHEN CONNECTION COMPLETE; HIT CONTINUE SWITCH./<15><12>
(1) 013034 005015 046120 040505 MSG5: .ASCIZ <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./<15><12>
(1) 013106 000000 .EVEN
942 013110 000000 SAVR0: 0
943 013112 000000 SAVR1: 0
944 013114 000000 SAVR2: 0
945 013116 000000 SAVR3: 0
946 013118 000000 SAVR4: 0
947 013120 000000 SAVR5: 0
948 013122 000000 DELAY: 0
949 013124 000300 BACK: 0
950 013126 000000 STOP: 0
*****
; INITIALIZE TRANSMIT SUBROUTINE
*****
951 013130 005037 011032 STARTX: CLR TIME
952 013134 005737 013122 TST DELAY ;IF SW04=1 & SW14=0 WAIT BEFORE TURNING ON TX
953 013140 001416 BEQ IS ;NO GO AHEAD AND TURN ON TX
954 013142 005037 013470 CLR TEMP1 ;PREPARE FOR DELAY

```

```

961 013146 012737 000007 013472      MOV      #7,TEMP2
962 013154 062737 000001 013470      ADD      #1,TEMP1      ;INCREMENT DELAY....
963 013162 001374      BNE
964 013164 005337 013472      DEC      TEMP2
965 013170 001371      BNE
966 013172 005037 013122      CLR      DELAY      ;ZERO DELAY
967 013176 013737 011070 1S:      MOV      IXDA,XDA      ;SETUP XMIT DATA ADDR.
968 013204 042700 100000      BIC      #XFLC,R0      ;RESET XMIT COMPLETE FLAG
969 013210 052714 000002      BIS      #DTR,#RXCSR    ;SET DTR
970 013214 005737 013126      TST
971 013220 001005      BNE
972 013222 104400 012702      TYPE
973 013226 000000      HALT
974 013230 005137 013126      COM
975 013234 032737 000010 011012 2S:      BIT      #HALF.DUPLEX,PARAM1
976 013242 001403      BEQ
977 013244 032714 010000      4S:      BIT      #10000,#RXCSR    ;IS CARRIER UP
978 013250 001375      BNE      4S      ;BR IF YES
979 013252 052714 000004      3S:      BIS      #RTS,#RXCSR    ;SET RTS
980 013256 032714 020000      CTSW:    BIT      #CTS,#RXCSR; IS CLEAR TO SEND SET?
981 013262 001017      BNE      CTSOK      ;BR IF YES
982 013264 023727 011032 000036 CTSW:    CMP      TIME,#36      ;30 SECS ELAPSED?
983 013272 103771      BLO      CTSW      ;BR IF NO
984 013274 011402      MOV      #RXCSR,R2      ;SETUP RECEIVE CSR
985 013276 016403 000004      MOV      TXCSR(R4),R3    ;SETUP XMIT CSR
986 013302 032777 010000 175534      BIT      #SM12,#SWR      ;INHIBIT PRINTOUTS?
987 013310 001061      BNE      1S      ;BR IF YES
988 013312 104002      HLT+2      ;PRINTOUT "WAITING TO XMIT" MESSAGE
989 013314 005037 011032      1S:      CLR      TIME      ;RESET TIMER
990 013320 001056      BR      CTSW      ;WAIT SOME MORE
991 013322 001037 011032      CTSOK:   CLR      TIME
992 013326 012737 000005 013376      MOV      #5,SYNCRNO      ;SETUP SYNC. COUNTER
993 013334 052764 000020 000004      BIS      #SEND,TXCSR(R4); SET SEND
994 013342 113764 011014 000006 1S:      MOVB    PARAM2,TXDBUF(R4);LOAD A SYNC CHAR
995
996 013350 105764 000004      TSTB    TXCSR(R4)      ;IS XMIT READY(TXDONE=1?)
997 013354 100375      BPL
998 013356 005337 013376      DEC      SYNCRNO      ;DECREMENT COUNTER
999 013362 001367      BNE      1S      ;BR IF NOT ZERO
1000
1001 013364 052764 000100 000004      BIS      #TXINTE,TXCSR(R4);SET XMIT INTERRUPT ENABLE
1002 013372 001240      NOP
1003 013374 001007      RTS      PC      ;EXIT FROM SUBROUTINE
1004 013376 000000      SYNCRNO: 0
1005
1006 ;*****
1007 ;***** INITIALIZE RECEIVER SUBROUTINE *****
1008 ;*****
1009 STARTR:
1010 013400 005737 013126      TST      STOP
1011 013404 001006      BNE      1S      ;FIRST TIME HERE?
1012 013406 052714 000002      BIS      #DTR,#RXCSR    ;SET DTR
1013 013412 104400 013034      TYPE     MSGS      ;MAKE CONNECTION
1014 013416 005137 013126      COM
1015 013422 005037 011032      1S:      CLR      TIME
1016 013426 013737 011020 011066      MOV      IRDA,RDA      ;SETUP RCV DATA ADDR

```

1017	013434	012737	001000	011064	MOV	#1000,RCC	:SETUP RCV CHAR COUNT
1018	013442	042700	040000		BIC	#RFLG,R0	:RESET RCV COMPLETE FLAG
1019	013446	005037	011054		CLR	ERCSR	:RESET ERROR RECORDS
1020	013452	005037	011056		CLR	EROSR	
1021	013456	005764	000002		TST	RXD8UF(R4)	:CLR ANY RXDONE
1022	013462	052714	000122		BIS	#INTEN!SYNSCH!DTR,@RXCSR	:SET INTER ENABLE & SEARCH SYNC &DTR
1023	013466	000207			RTS	PC	:EXIT FROM SUBROUTINE
1024	013470	000000		TEMP1:	0		
1025	013472	000000		TEMP2:	0		

```

1026 ;*****
1027 ; TRANSMIT INTERRUPT SERVICE ROUTINE
1028 ;*****
1029 XISR: NOP
1030 013474 000240 CMPB @XDA, TX.TERM ;FINISHED XMITTING?
1031 013476 127737 175366 011040 BNE XISR1 ;BR IF NO
1032 013504 001010 BNE XISR1 ;BR IF NO
1033 013506 052700 100000 BIS #XFLG, R0 ;SET XMIT COMPLETE FLAG
1034 013512 042764 000120 000004 BIC #TXINTE!SEND, TXCSR(R4) ;RESET XMIT INTR ENABLE
1035 013520 042714 000004 BIC #RTS, @RXCSR ;RESET RTS
1036 013524 000417 BR XISR2 ;
1037 XISR1: MOV @XDA, TXDBUF(R4); XMIT NEXT CHAR.
1038 013534 032777 000100 175302 BIT #100, @SWR ;MONITOR OUTPUT?
1039 013542 001406 BEQ NOXMON ;BR IF NO
1040 013544 105777 175326 TSTB @TPS ;IS TTY AVAILABLE
1041 013550 100003 BPL NOXMON ;BR IF NO
1042 013552 117777 175312 175320 MOV @XDA, @TPB; TYPE THE CHAR
1043 013560 NOXMON:
1044 013560 005237 011070 INC XDA ;INCREMENT ADDRESS
1045 013564 005037 011032 XISR2: CLR TIME ;RESET TIMER
1046 013570 000002 RTI ;RETURN FROM INTERRUPT
1047 ;*****
1048 ; RECEIVE INTERRUPT SERVICE ROUTINE
1049 ;*****
1050 RISR: NOP
1051 013572 000240 MOV @RXCSR, R2 ;SAVE RXCSR CONTENTS
1052 013574 011402 MOV TXCSR(R4), R3 ;SAVE TXCSR CONTENTS
1053 013576 016403 000004 MOV RXDBUF(R4), TEMP1 ;STORE CHAR AND ERROR FLAGS IF ANY
1054 013602 016437 000002 013470 013472 MOVB TEMP1, TEMP2
1055 013610 113737 013470 013472 BICB PARITY, TEMP2 ;STRIP PARITY
1056 013616 143737 014122 013472 MOVB TEMP1, @ROA ;MOVE CHAR TO INBUF
1057 013624 113777 013470 175234 BIT #40, @SWR ;MONITOR INPUT?
1058 013632 032777 000040 175204 BEQ NORMON ;BR IF NO
1059 013640 001406 BEQ NORMON ;BR IF NO
1060 013642 105777 175230 TSTB @TPS ;IS TTY AVAILABLE?
1061 013644 100003 BPL NORMON ;BR IF NO
1062 013650 113777 013472 175222 MOVB TEMP2, @IPB; TYPE THE CHAR
1063 013656 NORMON:
1064 013658 005237 011066 INC ROA ;BUMP POINTER
1065 013662 105077 175200 CLRB @ROA ;CLEAR NEXT CHAR POSITION
1066 013666 005337 011064 DEC RCC ;DECREMENT CHAR. COUNTER
1067 013672 001005 BNE IS ;BR IF BUFFER NOT FULL
1068 013674 042714 000120 BIC @INTEN!SYNSCH, @RXCSR ;RESET INTERRUPT FLAG
1069 013700 104006 HLT+6 ;RECEIVER BUFFER FULL
1070 013702 004737 013400 JSR PC, STARTR ;INITIALIZE RECEIVER
1071 013706 123737 011041 013472 IS: CMPB RX.TERM, TEMP2 ;IS IT LINE FEED?
1072 013714 001004 BNE RISR1 ;BR IF NO
1073 013716 042714 000120 BIC @INTEN!SYNSCH, @RXCSR ;DISABLE INTERRUPT AND SEARCH SYNC
1074 013722 052700 040020 BIS #RFLG, R0 ;SET RCVR COMPLETE FLAG
1075 013726 005737 013470 RISR1: TST TEMP1 ;IS THERE A DATA ERROR
1076 013732 100005 BPL RISR2 ;BR IF NO
1077 013734 010237 011054 MOV R2, @RXCSR ;SAVE RXCSR
1078 013740 013737 013470 011056 MOV TEMP1, @ROBR ;SAVE RXDBUF
1079 013746 005702 RISR2: TST R2 ;IS THERE A DATA SET STATUS CHANGE
1080 013750 100004 BPL RISR3 ;BR IF NO
1081 013752 010237 011060 MOV R2, @SSTAT ;SAVE STATUS
1082 013756 052700 020000 BIS #OSFLG, R0 ;SET FLAG

```

```

1082 013752
1083 013762 005037 011032
1084 013766 000002
1085
1086
1087
1088 013770 013537 013470
1089 013774 042737 170777 013470
1090 014002 022737 007000 013470
1091 014010 001005
1092 014012 012737 000400 014122
1093 014020 000137 014120
1094 014024 022737 005000 013470
1095 014032 001005
1096 014034 012737 000200 014122
1097 014042 000137 014120
1098 014046 022737 003000 013470
1099 014054 001005
1100 014056 012737 000100 014122
1101 014064 000137 014120
1102 014070 022737 001000 013470
1103 014076 001005
1104 014100 012737 000040 014122
1105 014106 000137 014120
1106 014112 012737 000200 014122
1107 014120 000205
1108 014122 000200
1109 000001

```

```

RISR3: CLR TIME ;RESET TIMER
RTI ;RETURN FROM INTERRUPT
*****
: STRIP PARITY SUBROUTINE
*****
STRIP: MOV @ (R5)+,TEMP1 ;PICK UP CONTENTS
BIC @C<7000>,TEMP1 ;STRIP OUT JUNK
CMP #EIGHT,PAREN,TEMP1 ;IS IT EIGHT BITS/CHAR ?
BNE .+14 ;NO
MOV #BIT8,PARITY ;YES
JMP OUT
CMP #SEVEN,PAREN,TEMP1 ;IS IT 7 BITS/CHAR ?
BNE .+14 ;NO
MOV #BIT7,PARITY ;YES
JMP OUT
CMP #SIX,PAREN,TEMP1 ;IS IT 6 BITS/CHAR ?
BNE .+14 ;NO
MOV #BIT6,PARITY ;YES
JMP OUT
CMP #FIVE,PAREN,TEMP1 ;IS IT 5 BITS/CHAR ?
BNE .+14 ;NO
MOV #BIT5,PARITY ;YES
JMP OUT
MOV #BIT7,PARITY ;DEFAULT VALUE
OUT: RTS
PARITY:BIT7 ;DEFAULT VALUE
.END

```



BA	011004	586#	645				
BACK	013124	712*	740*	775*	855*	886	950#
BIT#	= 002000	580#					
BIT0	= 000001	580#					
BIT1	= 000002	580#					
BIT10	= 002000	580#					
BIT11	= 004000	580#					
BIT12	= 010000	580#					
BIT13	= 020000	580#	615#	897	927		
BIT14	= 040000	580#					
BIT15	= 100000	580#					
BIT2	= 000004	580#					
BIT3	= 000010	580#					
BIT4	= 000020	580#					
BIT5	= 000040	580#	1104				
BIT6	= 000100	580#	1100				
BIT7	= 000200	580#	1096	1106	1108		
BIT8	= 000400	580#	1092				
BIT9	= 001000	580#					
BREAK	= 000001	580#					
B2016	011030	596#	900				
CARGET	= 010000	580#					
CLK	= 020000	580#					
CTS	= 020000	580#	980				
CTSOK	013322	981	991#				
CTSM	013256	980#	983	990			
CTSMN	013214	982#					
DELAY	013122	664*	884*	949#	958	966#	
DEERR	012534	928	937#				
DISPLA	011046	606#					
DNA	= 100000	580#					
DNAINT	= 000040	580#					
DSC	= 100000	580#					
DSFLG	= 020000	580#	614#	1081			
DSINTE	= 000040	580#					
DSR	= 001000	580#					
DSSTAT	011060	621#	1080*				
DTR	= 000002	580#	969	1012	1022		
DUI1	011000	585#					
EIGHT	= 006000	580#	590	1090			
ENTER	012264	868	877#				
EOP	012176	713	741	776	856	863#	
ERCSR	011054	619#	1019#	1076*			
EROBR	011056	620#	895	1020*	1077*		
EVEPAR	= 001400	580#					
EVPAR	= 000400	580#					
FIVE	= 000000	580#	1102				
FLAG	011042	604#					
FRMEAR	= 020000	580#					
FULL.0	= 000001	633#					
GO	011222	663#					
HALF.0	= 000010	580#	815	831	836	975	
HDXEN	= 000010	580#	589				
ILB	= 000010	580#	672				
IROR	011020	552#	779	904	930	1016	
ISYM00	= 000000	580#					

M02

IXDA	011022	593#	780	903	934	936	967		
KBDIN =	104416	580#	694	725	757	814			
LOOP =	000020	580#	710	738	773	853			
NEXT =	010000	580#							
MINT =	004000	580#							
MRESET =	000400	580#	650						
MSG0	012546	899	943#						
MSG1	012627	902	943#						
MSG2	012632	929	943#						
MSG3	012657	933	943#						
MSG4	012702	943#	972						
MSG5	013034	943#	1013						
MTDATA =	040000	580#							
NOOP =	000200	580#	706	769	850				
NOPAR =	000000	580#	590						
NORMON	013556	1058	1060	1062#					
NO-MON	013560	1039	1041	1043#					
OOOPAR =	001000	580#							
OUT	014120	1093	1097	1101	1105	1107#			
OVRUN =	040000	580#							
OWI =	000002	580#	669						
OWO =	000001	580#	666						
PARAM1	011012	589#	646	651	652	815	831	836	975
PARAM2	011014	590#	647	653	655	924	994		
PARAM3	011016	591#							
PARCSR =	000002	580#	653#						
PAREN =	001000	580#	1090	1094	1098	1102			
PARER =	010000	580#							
PARITY	014122	1055	1092#	1096#	1100#	1104#	1106#	1108#	
PRIOR	011010	583#	642	644					
PRTY0 =	000000	580#							
PRTY1 =	000040	580#							
PRTY2 =	000100	580#							
PRTY3 =	000140	580#							
PRTY4 =	000200	580#							
PRTY5 =	000240	580#							
PRTY6 =	000300	580#							
PRTY7 =	000340	580#	864						
QTPIE	012334	865#	866#	885	887#				
RCC	011064	624#	1017#	1065#					
RDA	011066	625#	1016#	1056#	1063#	1064#			
RDAK	012514	930#	932#						
REACT =	004000	580#							
RFLG =	040000	613#	696	709	760	772	822	848	1018
RING =	040000	580#							1073
RINTEN =	000100	580#	1022	1067	1072				
RISR	013572	641	1050#						
RISR1	013726	1071	1074#						
RISR2	013746	1075	1078#						
RISR3	013762	1079	1082#						
RIV	011006	587#	640						
RTS =	000004	580#	979	1034					
RWAIT =	104410	530#							
RXBUF =	000002	580#	1021	1053					
RXLINE =	000200	580#							
RERR =	100000	580#							

# N02

DUL1 ITP OVERLAY      MACY11 27(1006)      29-OCT-76 14:37 PAGE 28  
 DZDU08.P11      05-AUG-76 09:24      CROSS REFERENCE TABLE -- USER SYMBOLS

RX.TER	011041	602#	1070																		
SAVR0	013106	869#	878	943#																	
SAVR1	013110	870#	879	944#																	
SAVR2	013112	871#	880	945#																	
SAVR3	013114	872#	881	946#																	
SAVR4	013116	873#	882	947#																	
SAVRS	013120	874#	883	948#																	
SCAN4	012402	905#	906	919	922	925															
SCANS	012466	926#																			
SEND =	000020	580#	993	1033																	
SETTLE	011024	594#																			
SEVEN =	004000	580#	1094																		
SIX =	002000	580#	1098																		
SRC5R	011052	618#																			
SRO =	002000	580#																			
START	011102	599	637#																		
STARTR	013400	695	758	817	834	1009#	1069														
STARTX	013130	726	789	818	957#																
STD =	000010	580#																			
STOP	013126	665#	951#	970	974#	1010	1014*														
STPS =	104414	580#	864																		
STPSYN =	000400	580#	589																		
STRIP	013770	654	1088#																		
SAR	011044	605#	638	706	710	738	769	773	777	850	853	897	927	986							
		1038	1057																		
		580#	986																		
SW12 =	010000	617#																			
SXCSR	011050	992#	998#	1004#																	
SYMCNO	013376	580#																			
SYNEXT =	020000	580#																			
SYNINT =	030000	580#	590																		
SYNSCH =	000020	580#	1022	1067	1072																
SYSTST =	014000	580#																			
TEMP1	0.3470	960#	962#	1024#	1053#	1054	1056	1074	1077	1089#	1089#	1090	1094	1098							
		1102																			
TEMP2	013472	961#	964#	1025#	1054#	1055#	1061	1070													
TEST0	012336	708	771	852	895#																
TESTDX	012542	908	914	917	940#																
TIE =	000100	580#	866	867																	
TIME	011032	597#	663#	698	703#	727#	730	735#	759#	762	767#	788#	792	797#							
		819#	824	829#	840	845#	849#	957#	982	989#	991#	1015#	1045#	1083#							
TKB	011074	629#																			
TKS	011072	623#																			
TPB	011100	631#	1042#	1061#																	
TPS	011076	630#	1040	1059																	
TS:DAT	012372	896	898	903#																	
TXCSR =	000004	500#	650#	651#	985	993#	996	1001#	1033#	1052											
TXDEF =	000006	500#	954#	1037#																	
TXDOXE =	000200	500#																			
TXINTE =	000100	500#	1001	1033																	
TX.TER	011040	600#	907	1030																	
TYPE =	104400	500#	895	902	912	929	931	933	935	972	1013										
USER =	000000	500#																			
VCID =	000001	500#																			
XCC	011062	625#																			
XCSR =	000004	500#	649#	701	733	765		827	843	865	867#	885#									
XCR	011070	626#	781#	967#	1030	1037	10	1044#													

XFLG	= 100000	612#	728	737	790	799	820	833	838	847	968	1032
XISR	013474	643	1029#									
XISR1	013526	1031	1037#									
XISR2	013564	1035	1045#									
XLB	= 000004	580#	675									
XWAIT	= 104412	580#										
SILB	011516	674	757#	800								
SOWI	011312	671	694#	714								
SOWO	011420	668	725#	742								
SXLB	011746	677	814#	854	855							
.	= 014124	584#	679	783	934#	963	965	997	1091	1095	1099	1103

DUII ITEP OVERLAY MACY11 27(1006) 29-OCT-76 14:37 PAGE 31  
DZDUOB.P11 05-AUG-76 09:24 CROSS REFERENCE TABLE -- MACRO NAMES

BOX	18	581	608	634	953	1006	1026	1047	1085
DCPARM	18								
DHDOC1	18								
DHPARM	18								
DJPARM	18								
DLPARM	18								
DPPARM	18								
DQDOC1	18								
DUPARM	18								
DUPPAR	18	560							
DVDOC1	18								
DVPARM	18								
DZPARM	18								
HELLO	18								
HLT	5808	702	734	766	796	828	844	939	988
SEQUAT	18	580							1058
SINTF	18	580							
SITEP	18	656							
SSERV	18	623							

. ABS. 014124 000

ERRORS DETECTED: 0  
DEFAULT GLOSALS GENERATED: 0

. DZDUOB.SEB/SOL/CRF/ML:TOC=ITEP1.MAC,DZDUOB.P11  
RUN-TIME: 10 13 .9 SECONDS  
RUN-TIME RATIO: 44/26=1.6  
CORE USED: 16K (31 PAGES)

