

DJ11

OVERLAY FOR ITEP
MD-11-DZDJ-D-B

EP-DZDJ-D-B-DL-A

NOV 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

MADE IN U.S.A.

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDJD-B-D
PRODUCT NAME: DJ11 OVERLAY FOR INTERPROCESSOR TEST PROGRAM
PROGRAM DATE: OCTOBER 1976
MAINTAINER: DIAGNOSTICS
AUTHORS: R A JONES
JOHN EGOLF
FRY BRISMAN

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSIDERED 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 DJ11 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 DN10B, TYPE IN DNB.

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

- 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
SWR14=SETUP DN-11B ISR
SWR13=SETUP DN-11 ISR
SWR=00000=SETUP VARIABLE ISR
 2. THE FOLLOWING HALTS ARE REPEATED FOR EACH ISR SPECIFIED.
SETUP SEQUENCE IS: DN11, DN11-88 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 DN88 PARAMETERS ARE DISCUSSED IN SECT. 10.0 OF THE MONITOR.)
 - F. GO BACK TO STEP A IF THIS SETUP WAS FOR DN OR DN8.
 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 MODEN 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 TYPEOUTS
SW12=1 INHIBIT ALL TYPEOUTS 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 (9A QUICK BROWN FOX)
10=1 TEST MESSAGE #2 (9B NUMERICS)
11=1 TEST MESSAGE #3 (9C 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 (SMREG=LOC. 176) IS DEFAULTED TO. IF THIS IS THE CASE, UPON EXECUTION THE CONTENTS OF THE SMREG ARE DUMPED IN OCTAL ON THE CONSOLE TTY AND ANY CHANGES ARE REQUESTED

(IE) SMR=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 SMREG VALUE.

BUILT INTO THE PROGRAM IS THE ABILITY TO DYNAMICALLY CHANGE THE CONTENTS OF SMREG DURING PROGRAM EXECUTION. BY STRIKING ↑G (CNTRL G) ON CONSOLE TTY THE OPERATOR SETS A REQUEST FLAG TO CHANGE THE CONTENTS OF SMREG, 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 (CHR.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 83.

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 SM09 AND SM10(SM7=0)
3. TRANSMIT THE DATA SELECTED BY SM09 AND SM10 (SM8=0) OR TRANSMIT THE RECEIVED DATA (SM8=1)
4. RETURNS TO MONITOR FOR "END PASS" (SM4=1) OR GO TO STEP 1. (SM4=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 (SM7=0)
7. RETURNS TO MONITOR FOR "END PASS". (SM4=1) OR GO TO STEP 1(SM4=0)

ONE-WAY-IN MODE

1. THE OVERLAY WAITS FOR MESSAGE TO BE RECEIVED.
2. VERIFIES THE DATA(SM7=0)
3. RETURNS TO MONITOR FOR "END PASS"(SM4=1) OR GO TO STEP 1 (SM4=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". (SM4=1) OR GO TO STEP 1 (SM4=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 SMD4=0 THE OVERLAY WILL CONTINUE TO TRANSMIT/RECEIVE DATA.

IF SMD4=1 THE OVERLAY WILL RETURN TO THE MONITOR AND TYPE "END PASS".

IF BOTH SMD4=1 AND SM14=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:

#DXXXXXX 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=RRRRR
DATA SHOULD BE TTTTT
DATA COMPARE ERROR; BAD DATA=BBB GOOD DATA=GGG

WHERE RRRRR IS THE RECEIVE BUFFER (UP TO 512 CHARACTERS)
TTTTT 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<DDI> 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
RESTRICTION 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 DJ11 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: SOWI, IF "ONE WAY IN" MODE WAS SELECTED, SOMO, IF "ONE WAY OUT" MODE WAS SELECTED, SILB, IF "INTERNAL LOOP BACK" MODE WAS SELECTED, SXLB, IF "EXTERNAL LOOP BACK" WAS SELECTED.

9.31 SOWI: 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 SOMO: 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 SOMO OR TYPES END PASS DEPENDING ON SWITCH SETTINGS.

9.33 SILB: 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. (SILB)

9.34 SXLB: 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 \$XLB.
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(OHD,ONI,ILB,XLB) 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 DSR 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 DJ11
PARAM#1 IS THE LINE NUMBER IN OCTAL, DEFAULT= LINE 0
PARAM#2 BIT 0 IS FULL DUPLEX (1), DEFAULT= FULL DUPLEX (1)
PARAM#3 IS NOT USED (177777)

```

571
572
573
574
575
576 011000 045104 000040
577 011004 160010
578 011006 000300
579 011010 000240
580 011012 000000
581 011014 000001
582 011016 177777
583 011020 000000
584 011022 000000
585 011024 000000
586 011026 000000
587 011030 000000
588 011032 000000
589 011034 000000
590 011036 011102
591 011040
592 011040 000
593 011041
594 011041 001
595 011042 000000
596 011044 177570
597 011046 177570
600
601
602
603
604
605
606
607
608 011050 000000
609 011052 000000
610 011054 000000
611 011056 000000
612 011060 000000
613
614 011062 000000
615 011064 000000
616 011066 000000
617 011070 000000
618
619 011072 177560
620 011074 177562
621 011076 177564
622 011100 177566
623
624 000001
  
```

```

*****
: DJ11 INTERFACE SERVICE PARAMS
*****
  
```

```

DJ11:  =11000
BA:    .ASCIZ  /DJ /
RIV:   300
PRIOR: 240
PARAM1: 0
PARAM2: 1
PARAM3: 177777
IRDA:  .WORD  0
IXDA:  .WORD  0
SETTLE: .WORD  0
B2016: .WORD  0
TIME:  .WORD  0
TX. TERM: .WORD  START
RX. TERM: .BYTE  000
FLAG:    .BYTE  001
SMR:    .WORD  0
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=RD
XFLG=10000
RFLG=40000
DSFLG=20000
BIT13=20000
SXCSR: 0
SACSR: 0
ERCSR: 0
ERDSR: 0
DSSTAT: 0
XCC: 0
RCC: 0
ROA: 0
XDA: 0
TKS: 177560
TKB: 177562
TPS: 177564
TPB: 177566
  
```

```

: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.
  
```

FULL.DUPLEX=000001

691 011336 001402
 701 011340 000137 011374
 711 011344 002700 000010
 721 011348 001402
 731 011352 000137 011600
 741 011356 002700 000004
 751 011360 001402
 761 011364 000137 012030
 771 011370 000000
 781 011372 000776
 791
 801
 811
 821
 831
 841
 851
 861
 871
 881
 891
 901
 911
 921
 931
 941
 951
 961
 971
 981
 991
 1001
 1011
 1021
 1031
 1041
 1051
 1061
 1071
 1081
 1091
 1101
 1111
 1121
 1131
 1141
 1151
 1161
 1171
 1181
 1191
 1201
 1211
 1221
 1231
 1241
 1251
 1261
 1271
 1281
 1291
 1301
 1311
 1321
 1331
 1341
 1351
 1361
 1371
 1381
 1391
 1401
 1411
 1421
 1431
 1441
 1451
 1461
 1471
 1481
 1491
 1501
 1511
 1521
 1531
 1541
 1551
 1561
 1571
 1581
 1591
 1601
 1611
 1621
 1631
 1641
 1651
 1661
 1671
 1681
 1691
 1701
 1711
 1721
 1731
 1741
 1751
 1761
 1771
 1781
 1791
 1801
 1811
 1821
 1831
 1841
 1851
 1861
 1871
 1881
 1891
 1901
 1911
 1921
 1931
 1941
 1951
 1961
 1971
 1981
 1991
 2001
 2011
 2021
 2031
 2041
 2051
 2061
 2071
 2081
 2091
 2101
 2111
 2121
 2131
 2141
 2151
 2161
 2171
 2181
 2191
 2201
 2211
 2221
 2231
 2241
 2251
 2261
 2271
 2281
 2291
 2301
 2311
 2321
 2331
 2341
 2351
 2361
 2371
 2381
 2391
 2401
 2411
 2421
 2431
 2441
 2451
 2461
 2471
 2481
 2491
 2501
 2511
 2521
 2531
 2541
 2551
 2561
 2571
 2581
 2591
 2601
 2611
 2621
 2631
 2641
 2651
 2661
 2671
 2681
 2691
 2701
 2711
 2721
 2731
 2741
 2751
 2761
 2771
 2781
 2791
 2801
 2811
 2821
 2831
 2841
 2851
 2861
 2871
 2881
 2891
 2901
 2911
 2921
 2931
 2941
 2951
 2961
 2971
 2981
 2991
 3001
 3011
 3021
 3031
 3041
 3051
 3061
 3071
 3081
 3091
 3101
 3111
 3121
 3131
 3141
 3151
 3161
 3171
 3181
 3191
 3201
 3211
 3221
 3231
 3241
 3251
 3261
 3271
 3281
 3291
 3301
 3311
 3321
 3331
 3341
 3351
 3361
 3371
 3381
 3391
 3401
 3411
 3421
 3431
 3441
 3451
 3461
 3471
 3481
 3491
 3501
 3511
 3521
 3531
 3541
 3551
 3561
 3571
 3581
 3591
 3601
 3611
 3621
 3631
 3641
 3651
 3661
 3671
 3681
 3691
 3701
 3711
 3721
 3731
 3741
 3751
 3761
 3771
 3781
 3791
 3801
 3811
 3821
 3831
 3841
 3851
 3861
 3871
 3881
 3891
 3901
 3911
 3921
 3931
 3941
 3951
 3961
 3971
 3981
 3991
 4001
 4011
 4021
 4031
 4041
 4051
 4061
 4071
 4081
 4091
 4101
 4111
 4121
 4131
 4141
 4151
 4161
 4171
 4181
 4191
 4201
 4211
 4221
 4231
 4241
 4251
 4261
 4271
 4281
 4291
 4301
 4311
 4321
 4331
 4341
 4351
 4361
 4371
 4381
 4391
 4401
 4411
 4421
 4431
 4441
 4451
 4461
 4471
 4481
 4491
 4501
 4511
 4521
 4531
 4541
 4551
 4561
 4571
 4581
 4591
 4601
 4611
 4621
 4631
 4641
 4651
 4661
 4671
 4681
 4691
 4701
 4711
 4721
 4731
 4741
 4751
 4761
 4771
 4781
 4791
 4801
 4811
 4821
 4831
 4841
 4851
 4861
 4871
 4881
 4891
 4901
 4911
 4921
 4931
 4941
 4951
 4961
 4971
 4981
 4991
 5001
 5011
 5021
 5031
 5041
 5051
 5061
 5071
 5081
 5091
 5101
 5111
 5121
 5131
 5141
 5151
 5161
 5171
 5181
 5191
 5201
 5211
 5221
 5231
 5241
 5251
 5261
 5271
 5281
 5291
 5301
 5311
 5321
 5331
 5341
 5351
 5361
 5371
 5381
 5391
 5401
 5411
 5421
 5431
 5441
 5451
 5461
 5471
 5481
 5491
 5501
 5511
 5521
 5531
 5541
 5551
 5561
 5571
 5581
 5591
 5601
 5611
 5621
 5631
 5641
 5651
 5661
 5671
 5681
 5691
 5701
 5711
 5721
 5731
 5741
 5751
 5761
 5771
 5781
 5791
 5801
 5811
 5821
 5831
 5841
 5851
 5861
 5871
 5881
 5891
 5901
 5911
 5921
 5931
 5941
 5951
 5961
 5971
 5981
 5991
 6001
 6011
 6021
 6031
 6041
 6051
 6061
 6071
 6081
 6091
 6101
 6111
 6121
 6131
 6141
 6151
 6161
 6171
 6181
 6191
 6201
 6211
 6221
 6231
 6241
 6251
 6261
 6271
 6281
 6291
 6301
 6311
 6321
 6331
 6341
 6351
 6361
 6371
 6381
 6391
 6401
 6411
 6421
 6431
 6441
 6451
 6461
 6471
 6481
 6491
 6501
 6511
 6521
 6531
 6541
 6551
 6561
 6571
 6581
 6591
 6601
 6611
 6621
 6631
 6641
 6651
 6661
 6671
 6681
 6691
 6701
 6711
 6721
 6731
 6741
 6751
 6761
 6771
 6781
 6791
 6801
 6811
 6821
 6831
 6841
 6851
 6861
 6871
 6881
 6891
 6901
 6911
 6921
 6931
 6941
 6951
 6961
 6971
 6981
 6991
 7001
 7011
 7021
 7031
 7041
 7051
 7061
 7071
 7081
 7091
 7101
 7111
 7121
 7131
 7141
 7151
 7161
 7171
 7181
 7191
 7201
 7211
 7221
 7231
 7241
 7251
 7261
 7271
 7281
 7291
 7301
 7311
 7321
 7331
 7341
 7351
 7361
 7371
 7381
 7391
 7401
 7411
 7421
 7431
 7441
 7451
 7461
 7471
 7481
 7491
 7501
 7511
 7521
 7531
 7541
 7551
 7561
 7571
 7581
 7591
 7601
 7611
 7621
 7631
 7641
 7651
 7661
 7671
 7681
 7691
 7701
 7711
 7721
 7731
 7741
 7751
 7761
 7771
 7781
 7791
 7801
 7811
 7821
 7831
 7841
 7851
 7861
 7871
 7881
 7891
 7901
 7911
 7921
 7931
 7941
 7951
 7961
 7971
 7981
 7991
 8001
 8011
 8021
 8031
 8041
 8051
 8061
 8071
 8081
 8091
 8101
 8111
 8121
 8131
 8141
 8151
 8161
 8171
 8181
 8191
 8201
 8211
 8221
 8231
 8241
 8251
 8261
 8271
 8281
 8291
 8301
 8311
 8321
 8331
 8341
 8351
 8361
 8371
 8381
 8391
 8401
 8411
 8421
 8431
 8441
 8451
 8461
 8471
 8481
 8491
 8501
 8511
 8521
 8531
 8541
 8551
 8561
 8571
 8581
 8591
 8601
 8611
 8621
 8631
 8641
 8651
 8661
 8671
 8681
 8691
 8701
 8711
 8721
 8731
 8741
 8751
 8761
 8771
 8781
 8791
 8801
 8811
 8821
 8831
 8841
 8851
 8861
 8871
 8881
 8891
 8901
 8911
 8921
 8931
 8941
 8951
 8961
 8971
 8981
 8991
 9001
 9011
 9021
 9031
 9041
 9051
 9061
 9071
 9081
 9091
 9101
 9111
 9121
 9131
 9141
 9151
 9161
 9171
 9181
 9191
 9201
 9211
 9221
 9231
 9241
 9251
 9261
 9271
 9281
 9291
 9301
 9311
 9321
 9331
 9341
 9351
 9361
 9371
 9381
 9391
 9401
 9411
 9421
 9431
 9441
 9451
 9461
 9471
 9481
 9491
 9501
 9511
 9521
 9531
 9541
 9551
 9561
 9571
 9581
 9591
 9601
 9611
 9621
 9631
 9641
 9651
 9661
 9671
 9681
 9691
 9701
 9711
 9721
 9731
 9741
 9751
 9761
 9771
 9781
 9791
 9801
 9811
 9821
 9831
 9841
 9851
 9861
 9871
 9881
 9891
 9901
 9911
 9921
 9931
 9941
 9951
 9961
 9971
 9981
 9991
 1001
 10011
 10021
 10031
 10041
 10051
 10061
 10071
 10081
 10091
 10101
 10111
 10121
 10131
 10141
 10151
 10161
 10171
 10181
 10191
 10201
 10211
 10221
 10231
 10241
 10251
 10261
 10271
 10281
 10291
 10301
 10311
 10321
 10331
 10341
 10351
 10361
 10371
 10381
 10391
 10401
 10411
 10421
 10431
 10441
 10451
 10461
 10471
 10481
 10491
 10501
 10511
 10521
 10531
 10541
 10551
 10561
 10571
 10581
 10591
 10601
 10611
 10621
 10631
 10641
 10651
 10661
 10671
 10681
 10691
 10701
 10711
 10721
 10731
 10741
 10751
 10761
 10771
 10781
 10791
 10801
 10811
 10821
 10831
 10841
 10851
 10861
 10871
 10881
 10891
 10901
 10911
 10921
 10931
 10941
 10951
 10961
 10971
 10981
 10991
 11001
 11011
 11021
 11031
 11041
 11051
 11061
 11071
 11081
 11091
 11101
 11111
 11121
 11131
 11141
 11151
 11161
 11171
 11181
 11191
 11201
 11211
 11221
 11231
 11241
 11251
 11261
 11271
 11281
 11291
 11301
 11311
 11321
 11331
 11341
 11351
 11361
 11371
 11381
 11391
 11401
 11411
 11421
 11431
 11441
 11451
 11461
 11471
 11481
 11491
 11501
 11511
 11521
 11531
 11541
 11551
 11561
 11571
 11581
 11591
 11601
 11611
 11621
 11631
 11641
 11651
 11661
 11671
 11681
 11691
 11701
 11711
 11721
 11731
 11741
 11751
 11761
 11771
 11781
 11791
 11801
 11811
 11821
 11831
 11841
 11851
 11861
 11871
 11881
 11891
 11901
 11911
 11921
 11931
 11941
 11951
 11961
 11971
 11981
 11991
 12001
 12011
 12021
 12031
 12041
 12051
 12061
 12071
 12081
 12091
 12101
 12111
 12121
 12131
 12141
 12151
 12161
 12171
 12181
 12191
 12201
 12211
 12221
 12231
 12241
 12251
 12261
 12271
 12281
 12291
 12301
 12311
 12321
 12331
 12341
 12351
 12361
 12371
 12381
 12391
 12401
 12411
 12421
 12431
 12441
 12451
 12461
 12471
 12481
 12491
 12501
 12511
 12521
 12531
 12541
 12551
 12561
 12571
 12581
 12591
 12601
 12611
 12621
 12631
 12641
 12651
 12661
 12671
 12681
 12691
 12701
 12711
 12721
 12731
 12741
 12751
 12761
 12771
 12781
 12791
 12801
 12811
 12821
 12831
 12841
 12851
 12861
 12871
 12881
 12891
 12901
 12911
 12921
 12931
 12941
 12951
 12961
 12971
 12981
 12991
 13001
 13011
 13021
 13031
 13041
 13051
 13061
 13071
 13081
 13091
 13101
 13111
 13121
 13131
 13141
 13151
 13161
 13171
 13181
 13191
 13201
 13211
 13221
 13231
 13241
 13251
 13261
 13271
 13281
 13291
 13301
 13311
 13321
 13331
 13341
 13351
 13361
 13371
 13381
 13391
 13401
 13411
 13421
 13431
 13441
 13451
 13461
 13471
 13481
 13491
 13501
 13511
 13521
 13531
 13541
 13551
 13561
 13571
 13581
 13591
 13601
 13611
 13621
 13631
 13641
 13651
 13661
 13671
 13681
 136

737	011504	004737	313202			JSR	PC, STARTX
738	011510	005037	011032			CLR	TIME
739	011514	032700	100000	15:		DIT	#XFLG, STAT
740	011520	001013				BNE	25
741	011526	023727	011032	000100		CHP	TIME, #100
742	011530	103771				BLO	15
743	011536	011402				MOV	#RCSR, R2
744	011539	016403	000004			MOV	XCSR(R4), R3
745	011540	104001				HLT	1
746	011542	005037	011032			CLR	TIME
747	011546	000762				BR	15
748	011550	042700	100000	25:		BIC	#XFLG, STAT
749	011554	032777	000020	177262		DIT	#LOOP, #SMR
750	011558	001405				BEG	35
751	011564	012737	011576	013176		MOV	#35, BACK
752	011572	000137	012260			JMP	EOP
753	011576	000741		35:		BR	SOWO

757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811

011600 104416
011602 004737 013566
011605 005037 011032
011612 032700 040000
011616 001013
011620 023727 011032 000100
011625 103771
011630 011402
011632 016403 000004
011635 104001
011640 005037 011032
011644 000762
011646 032777 000200 177170 25:
011654 001002
011656 004737 012420
011662 042700 040000 35:
011666 032777 000020 177150
011674 001405
011676 012737 011710 013176
011704 000137 012250
011710 032777 000400 177126 45:
011716 001416
011720 013702 011020
011724 013702 011022
011730 010337 011070
011734 112223
011736 001376
011740 112743 000177
011744 005203
011746 112723 000177
011752 105023
011754 005037 011032 75:
011760 004737 013202
011764 032700 100000 55:
011770 001013
011772 023727 011032 000100
012000 103771
012002 011402
012004 016403 000004
012010 104001
012012 005037 011032
012016 000762
012020 042700 100000 65:
012024 000137 011600

ROUTINE USED IF INTERNAL LOOP BACK WAS SELECTED.
NOTE THAT WHEN IN THIS MODE; HALF DUPLEX IS THE
ONLY MODE AVAILABLE.
"INTERNAL LOOP BACK" MEANS THAT THE RECEIVER IS "TURNED ON"
AND A COMPLETE MESSAGE IS RECEIVED. IF DATA IS TO BE CHECKED
IT IS: IF "END PASS" IS DESIRED; IT IS GIVEN.
THEN THE TRANSMITTER IS ENABLED. AFTER THE WHOLE MESSAGE
IS TRANSMITTED; THE CYCLE IS REPETED AS ABOVE.

SILB: KBDIN
JSR PC, STARTR
CLR TIME
15: BIT #AFLG, STAT
BNE 25
CMP TIME, #100
BLO 15
MOV #ACSR, R2
MOV XCSR(A4), R3
HLT 1
CLR TIME
BR 15
25: BIT #MODAT, #SMR
BNE 35
JSR PC, TESTD
35: BIC #AFLG, STAT
BIT #LOOP, #SMR
BEQ 45
MOV #45, BACK
JMP EOP
45: BIT #400, #SMR
BEQ 75
MOV IXDA, R2
MOV IXDA, R3
MOV R3, XDA
MOVB (R2)+, (R3)+
BNE -3
MOVB #177, -(R3)
INC R3
MOVB #177, (R3)+
CLRB (R3)+
75: CLR TIME
JSR PC, STARTX
55: BIT #AFLG, STAT
BNE 65
CMP TIME, #100
BLO 55
MOV #ACSR, R2
MOV XCSR(A4), R3
HLT 1
CLR TIME
BR 55
65: BIC #AFLG, STAT
JMP SILB

;USE EXTERNAL DATA?
;BR IF NO
;SET POINTER
;SET POINTER
;SETUP XMIT DATA ADDR
;MOVE INPUT TO OUTPUT
;LOOP IF NOT ZERO CHAR
;INSERT A FILL CHAR
;RAMP ADDRESS
;INSERT ANOTHER FILL
;INSERT ZERO CHAR

012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047

012030	104416		
012032	032737	000001	011014
012040	001402		
012042	004737	013566	
012044	004737	013202	
012046	005037	011032	
012048	032700	100000	
012050	001016		
012052	032700	040000	
012054	001024		
012072	023727	011032	000100
012100	103766		
012102	011402		
012104	016403	000004	
012110	104001		
012112	005037	011032	
012114	000756		
012116	032737	000001	011014
012118	001356		
012120	042700	100000	
012134	004737	013566	
012140	000746		
012142	032737	000001	011014
012144	001420		
012146	032700	100000	
012148	001013		
012150	023727	011032	000100
012166	103766		
012170	011402		
012172	016403	000004	
012174	104001		
012200	005037	011032	
012202	000756		
012204	042700	100000	
012206	042700	040000	
012216	005037	011032	
012220	032777	000200	176614
012222	001002		
012224	004737	012420	
012226	032777	000020	176600
012244	001671		
012246	012737	012030	013176
012254	000137	012260	

```

*****
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.
*****

```

5XLB:	KBDIN
	BIT
	BEQ
	JSR
15:	JSR
	CLR
25:	BIT
	BNE
75:	BIT
	BNE
	CHP
	BLO
	NOV
	NOV
	HLT
	CLR
	BR
35:	BIT
	BNE
	BIC
	JSR
	BR
45:	BIT
	BEQ
	BIT
	BNE
	CHP
	BLO
	NOV
	NOV
	HLT
	CLR
	BR
55:	BIC
55:	BIC
	CLR
	BIT
	BNE
	JSR
55:	BIT
	BEQ
	NOV
	JMP

```

#FULL.DUPLEX,PARAM2
15
PC,STARTR
PC,STARTX
TIME
#XFLG,STAT
35
#RFLG,STAT
45
TIME,#100
25
#ACSR,R2
XCSR(R4),R3
1
TIME
25
#FULL.DUPLEX,PARAM2
75
#XFLG,STAT
PC,STARTR
25
#FULL.DUPLEX,PARAM2
85
#XFLG,STAT
65
TIME,#100
45
#ACSR,R2
XCSR(R4),R3
1
TIME
45
#XFLG,STAT
#RFLG,STAT
TIME
#MODAT,#SMR
55
PC,TESTD
#LOOP,#SMR
5XLB
#5XLB,BACK
EOP

```

```

868
869
870
871
872
873
874 012260
875 012260 104414 000340
876 012264 016437 000004 012416
877 012272 042737 137777 012416
878 012300 042764 040000 000004
879 012306 012766 012346 000002
880 012314 010037 013160
881 012320 010137 013162
882 012324 010237 013164
883 012330 010337 013166
884 012334 010437 013170
885 012340 010537 013172
886 012344 000207
887
888
889 012346
890 012346 013700 013160
891 012352 013701 013162
892 012358 013702 013164
893 012364 013703 013166
894 012366 013704 013170
895 012372 013705 013172
896 012376 012737 177777 013174
897 012404 053764 012416 000004
898 012412 000177 000560
899 012416 000000
900
901
902
903
904
905
906
907
908
909
910 012420 013746 011056
911 012424 001413
912 012426 022777 020000 176410
913 012434 001007
914 012436 104400 012620
915 012442 004077 176362
916 012446 005746
917 012450 104400 012701
918 012454 013701 011022
919 012458 013702 011020
920 012464 122122
921 012466 001776
922 012470 123741 011040
923 012474 001447
924 012476 122742 000002
925 012482 001006
926 012484 001006
927 012486 010237 012512
928 012510 104400
    
```

```

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

```

EOP:
STPS,PTY7
MOV XCSR(R4),QTPIE ;SET PS PRIORITY TO 7
BIC #1C(TIE),QTPIE ;SAVE TX CSR
BIC #TIE,XCSR(R4) ;CLEAR ALL BUT TX IE.
MOV BENTER,2(SP) ;CLEAR TX IE (EVEN IF IT WASN'T SET)
MOV R0,SAVR0 ;SET FOR RETURN IF SW.14=1
MOV R1,SAVR1 ;SAVE REGISTER 0
MOV R2,SAVR2 ;SAVE REGISTER 1
MOV R3,SAVR3 ;SAVE REGISTER 2
MOV R4,SAVR4 ;SAVE REGISTER 3
MOV R5,SAVR5 ;SAVE REGISTER 4
RTS PC ;SAVE REGISTER 5
;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 QTPIE,XCSR(R4) ;IF ORGINALLY SET; SET TX IE
JMP @BACK
QTPIE: 000000
    
```

```

*****
SUBROUTINE TO CHECK
RECEIVER DATA.
*****
TESTD: MOV ENDR, -(SP) ;WAS THERE A RECEIVE ERROR?
BEQ TSTDAT ;BR IF NO
BIT #BIT13,ASMR ;INHIBIT PRINTOUTS?
BNE TSTDAT ;BR IF YES
TYPE MSG0 ;<15><12>THERE WAS A RECEIVE ERROR. RBUF=
JSR @202016 ;PRINT CONTENTS OF RBUF
TST -(SP)
TYPE MSG1 ;<15><12>
TSTDAT: MOV IXDA, R1 ;SETUP XMIT DATA ADDR
MOV IRDA, R2 ;SETUP RCV DATA ADDR
SCANH: CFPB (R1)+,(R2)+ ;DATA OK?
BEQ SCANH ;BR IF OK
CFPB TX_TERM,-(R1) ;IS IT END OF DATA
BEQ TESTDX ;BR IF YES
CFPB @002,-(R2)
BNE @25
MOV R2,18
TYPE
    
```



```
012512 000000 15: .WORD 0
012514 000437 BR TESTDX
012516 105712 25: TSTB (R2)
012520 001435 BEQ TESTDX ;BR IF YES
012522 122721 000177 CHPB #177, (R1)+ ;IS IT FILL CHAR?
012524 001756 BEQ SCAN4 ;BR IF YES
012530 005301 DEC R1 ;BACKUP
012532 122722 000177 CHPB #177, (R2)+ ;IS IT FILL?
012534 001752 BEQ SCAN4 ;BR IF YES
012540 000240 NOP ;DATA ERROR
012542 032777 020000 176274 SCANS: BIT #BIT13, 2SMR ;INHIBIT PRINTOUTS
012550 001016 BNE DERR ;BR IF YES
012552 104400 012704 TYPE MSG2 ;<15><12>RECEIVED DATA = <15><12>
012554 013737 011020 012566 MOV IXDA, RDAX ;SETUP DATA ADDRESS
012556 104400 TYPE ;PRINT RECEIVED DATA
012560 000000 RDAX: 0 ;RECEIVED DATA ADDR.
012570 104400 012731 TYPE MSG3 ;<15><12>DATA SHOULD BE<15><12>
012574 013737 011022 012604 MOV IXDA, .+10 ;SETUP ADDR.
012576 104400 TYPE ;PRINT GOOD DATA
012580 011022 IXDA
012582 111103 DERR: MOVB (R1), R3 ;SETUP XMIT DATA
012584 114202 MOVB -(R2), R2 ;SETUP RCV DATA
012586 104007 HLT+7 ;DATA ERROR HALT
012588 005725 TESTDX: TST (SP)+ ;POP STACK
012590 000207 RTS PC ;RETURN FROM SUB/ROUT

(1) 012620 005015 044124 051105 MSG0: .ASCIZ <15><12>/THERE WAS A RECEIVER ERROR. REGISTER (SEL 2) =/
(1) 012701 015 000012 MSG1: .ASCIZ <15><12>
(1) 012704 005015 042522 042503 MSG2: .ASCIZ <15><12>/RECEIVED DATA = /<15><12>
(1) 012731 015 042012 052101 MSG3: .ASCIZ <15><12>/DATA SHOULD BE/<15><12>
(1) 012754 005015 046120 040505 MSG4: .ASCII <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./
(1) 013023 015 053412 042510 MSG5: .ASCIZ <15><12>/WHEN CONNECTION COMPLETE; HIT CONTINUE SWITCH./<15><12>
(1) 013106 005015 046120 040505 MSG5: .ASCIZ <15><12>/PLEASE MAKE CONNECTION (DIAL NUMBER)./<15><12>
(1) 013160 000000 .EVEN
013162 000000 SAVR0: 0
013164 000000 SAVR1: 0
013166 000000 SAVR2: 0
013170 000000 SAVR3: 0
013172 000000 SAVR4: 0
013174 000000 SAVR5: 0
013176 000000 DELAY: 0
013200 000000 BACK: 0
STOP: 0
```

1

```
*****  
; TRANSMITTER INITIALIZATION SUBROUTINE  
*****  
961 013202 005737 013174 STARTX: TST DELAY ; IF SM04=1 & SM14=0 DELAY  
962 013206 001416 BEQ IS ; NO DELAY START TRANSMITTER  
963 013210 005037 013554 CLR TEMP1 ; PREPARE FOR DELAY  
964 013214 012737 000007 013556 MOV #7,TEMP2  
965 013218 012737 000001 013554 ADD #1,TEMP1 ; INCREMENT DELAY  
966 013222 001374 BNE -6  
967 013226 005337 013556 DEC TEMP2  
968 013230 001371 BNE -14  
969 013234 005037 013174 CLR DELAY  
970 013238 042700 100000 15: BIC #XFLG,STAT ; SET UP XMIT DATA ADD  
971 013242 013737 011022 011070 MOV IXDA,XDA ; FIRST TIME HERE?  
972 013246 005737 013200 TST STOP ; NO  
973 013250 001021 BNE 25 ; MAKE CONNECTION  
974 013254 104400 012754 TYPE ,MSG4  
975 013258 000000 HALT  
976 013262 005137 013200 COM STOP ; COMPLEMENT STOP  
977 013266 005037 013554 CLR TEMP1 ; YES PREPARE FOR DELAY  
978 013270 012737 000030 013556 MOV #14*2,TEMP2  
979 013274 012737 000001 013554 ADD #1,TEMP1 ; INCREMENT DELAY  
980 013278 005337 013556 DEC TEMP2  
981 013282 001371 BNE -14  
982 013286 042714 002000 25: BIC #RMBK,ARCSR ; SELECT TCR  
983 013290 013764 013560 000004 MOV TCRTHP,TCR(R4) ; SET LINE # IN TCR  
984 013294 012737 000001 011014 BIT #FULL.DUPLEX,PARAM2 ; FULL DUPLEX?  
985 013298 001414 BEQ 35 ; BR IF NO  
986 013302 012700 000004 BIT #XLB,MODE ; XLB MODE?  
987 013306 001411 BEQ 35 ; BR IF NO  
988 013310 012737 177777 013564 MOV #-1,TRNFLG ; SET FLAG  
989 013314 012714 040400 BIS #TIE+HTSE,ARCSR ; SET INTERRUPT ENABLE  
990 013318 000001 WAIT  
991 013322 005737 013562 TST SNCFLG ; FIRST CHAR RECEIVED YET?  
992 013326 001375 BNE -4 ; BR IF NO  
993 013330 012714 040400 35: BIS #TIE+HTSE,ARCSR ; SET INTERRUPT ENABLE,SCAN ENABLE  
994 013334 000207 RTS PC  
1000  
1001 013406 127737 175456 011040 XISR: CNPB XDA,TX.TERM ; IS CHAR TRANSMITTER TERMINATION CHAR  
1002 013410 001005 BNE XISR1 ; BR IF NO  
1003 013414 012700 100000 BIS #XFLG,STAT ; SET XMIT DONE FLAG  
1004 013418 042714 040400 BIC #TIE+HTSE,ARCSR ; CLEAR ENABLES  
1005 013422 000444 BR XISR3  
1006 013426 005737 011012 XISR1: TST PARAM1 ; IS IT LINE 0  
1007 013430 001015 BNE XISR2 ; BR IF NO  
1008 013434 105764 000007 TSTB XBUF+1(R4) ; DID SCAN STOP AT LINE 0  
1009 013438 001412 BEQ XISR2 ; BR IF YES  
1010 013442 016437 000006 013552 MOV XBUF(R4),ERROR1 ; SAVE XBUF  
1011 013446 011402 MOV ARCSR,R2 ; SET UP R2 WITH SCR CONTENTS  
1012 013450 005003 CLR R3  
1013 013454 104010 HLT 10 ; ERROR WRONG LINE  
1014 013458 104400 014135 TYPE ,SCANE ; TYPE ERROR MESSAGE  
1015 013462 000000 HALT  
1016 013466 000776 BR -2
```


1017	013470	117764	175374	000006	XISR2:	MOVB	2XDA,XBUF(R4)	: TRANSMIT DATA
1018	013476	032777	000100	175340		BIT	#100,2SWR	: MONITOR TX DATA?
1019	013504	001406				BEQ	NOXMON	: BR IF NO
1020	013506	105777	175364			TSTB	2TPS	: TTY READY?
1021	013512	100003				BPL	NOXMON	: BR IF NO
1022	013514	117777	175350	175356		MOVB	2XDA,2TPB	: TYPE CHAR
1023	013522	005237	011070		NOXMON:	INC	XDA	: INC TXBUF POINTER
1024	013522	005737	013564			TST	TRNFLG	
1025	013522	001402				BEQ	XISR3	
1026	013534	042714	040400			BIC	#TIE+MTSE,2RCSR	
1027	013540	005037	011032		XISR3:	CLR	TIME	
1028	013544	005037	013564			CLR	TRNFLG	
1029	013550	000002				RTI		
1030	013550	000000			ERROR1:	0		
1031	013550	000000			TEMP1:	0		
1032	013550	000000			TEMP2:	0		
1033	013550	000000			TCRTHP:	0		
1034	013550	000000			SNCFLG:	0		
1035	013564	000000			TRNFLG:	0		

1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091

013566 005737 013200
013572 001005
013574 104400 012754
013580 005137 013200
013584 000000
013586 032737 000001 011014 1S:
013588 001410
013590 032700 000004
013592 001405
013594 005037 013554
013596 005237 013554
013598 001375
013600 042700 040000
013602 013737 011020 011066 2S:
013604 012737 001000 011064
013606 012737 177777 013562
013608 005037 011054
013610 005037 011056
013612 005764 000002 3S:
013614 100775
013616 002714 000101
013618 000207

013710 000240
013712 105714
013714 100403
013716 011402
013720 005003
013722 104010
013724 016401 000002 1S:
013726 042701 000200
013728 005701
013730 100403
013732 011402
013734 005003
013736 104010
013738 032701 070000 2S:
013740 001404
013742 011437 011054
013744 010137 011056
013746 110177 175076 3S:
013748 032777 000040 175046
013750 001405
013752 105777 175072
013754 100002
013756 110177 175066
013758 005237 011066
013760 105077 175044
013762 005337 011064
013764 001007
013766 000005
013768 005002

```
*****  
; RECEIVER INITIALIZATION SUBROUTINE  
*****  
STARTR: TST      STOP      ;FIRST TIME HERE?  
         BNE      15        ;BR IF NO  
         TYPE     MSG4      ;TYPE "MAKE CONNECTION"  
         COM      STOP      ;COMPLEMENT STOP  
         HALT  
1S:     BIT      @FULL.DUPLEX,PARAM2;FULL DUPLEX?  
         BEQ     25        ;BR IF NO  
         BIT     @XLB,MODE   ;XLB MODE?  
         BEQ     25        ;BR IF NO  
         CLR     TEMP1      ;START DELAY  
         INC     TEMP1  
         BNE     .-4  
2S:     BIC     @RFLG,STAT  
         MOV     IRDA,ADA    ;SET UP RECEIVER DATA ADD  
         MOV     @1000,RCC  ;SET UP BUFFER LIMIT  
         MOV     @-1,SNCF LG  
         CLR     ERCSR      ;CLEAR ERROR RECORDS  
         CLR     ERDR  
3S:     TST     @RBUF(R4)   ;CLEAR SILO  
         BMI     3S        ;KEEP CLEARING UNTIL BIT 15 CLEAR  
         BIS     @RCEN+RIE,@RCSR ;SET INTERRUPT ENABLE,RECEIVER ENABLE  
         RTS     PC  
  
RISR:   NOP  
         TSTB   @RCSR      ;DID RECEIVER DONE SET?  
         BMI     15        ;BR IF YES  
         MOV    @RCSR,R2   ;SAVE CSR  
         CLR    R3  
         HALT 10  
1S:     MOV     @RBUF(R4),R1 ;ERROR RECEIVER INTERRUPTED BUT DONE NOT SET  
         BIC     @R200,R1  ;GET CHAR  
         TST    R1        ;STRIP A BIT  
         BMI     25        ;IS CHAR PRESENT SET  
         MOV    @RCSR,R2  ;BR IF YES  
         CLR    R3        ;SAVE CSR  
         HALT 10  
2S:     BIT     @@RBUF+FRME+PARE,R1 ;ERROR CHAR PRESENT NOT SET  
         BEQ    3S        ;CHECK FOR RECEIVER ERRORS  
         MOV    @RCSR,ERCSR ;BR IF NO ERRORS  
         MOV    R1,ERDR    ;SAVE CSR  
         MOV    R1,ADA     ;SAVE RBUF  
         BIT    @@R15,@SMR ;STORE CHAR  
         BEQ    NORMON     ;MONITOR RXDATA?  
         TSTB  @TPS        ;BR IF NO  
         BEQ    3S        ;IS TTY READY?  
         MOV    R1,@TPB    ;BR IF NO  
         INC   @RBUF,PC    ;TYPE CHAR  
         CLR   @RBUF,PC   ;INC RBUF POINTER  
         DEC   @RCC        ;CLEAR NEXT POSITION  
         BNE  15          ;DEC CHAR COUNT  
         BEQ  15          ;BUFFER FULL YET?  
         CLR   R2         ;STOP THE SHOW,BUFFER OVERFLOWED
```


1092	014034	005003				CLR	R3	
1093	014036	104000				HLT	0	
1094	014040	104006				HLT+6		;RECEIVER BUFFER FULL
1095	014042	000000				HALT		
1096	014044	000776				BR	-2	
1097	014046	123701	011041	15:		CHPB	RX.TERM,R1	;IS CHAR 001?
1098	014052	001004				BNE	RISR1	;BR IF NO
1099	014054	042714	000100			BIC	#RIE,#RCSR	;CLEAR RECEIVER INTERRUPT ENABLE
1100	014050	052700	040000			BIS	#RFLG,STAT	;SET R DONE FLAG
1101	014064	005037	011032	RISR1:		CLR	TIME	
1102	014070	005037	013562			CLR	SNCFLG	
1103	014074	000002				RTI		;GO HOME
1104	014076	005015	042522	042503	MFULL:	.ASCIZ<15><12>/RECEIVER BUFFER FULL ERROR!!/		
	014135	015	042412	051122	SCAN:	.ASCIZ<15><12>/ERROR!TRANSMITTER SCAN STOPPED ON WRONG LINE/		
					.EVEN			
					.END			
		000001						

BA	011004	577#	636						
BACK	013176	723#	751#	786#	866#	897	958#		
BIT0	000001	571#							
BIT1	000002	571#							
BIT10	002000	571#							
BIT11	004000	571#							
BIT12	010000	571#							
BIT13	020000	571#	606#	908	935				
BIT14	040000	571#							
BIT15	100000	571#							
BIT2	000004	571#							
BIT3	000010	571#							
BIT4	000020	571#							
BIT5	000040	571#	1081						
BIT6	000100	571#							
BIT7	000200	571#							
BIT8	000400	571#							
BIT9	001000	571#							
BSY	000020	571#	645						
B2016	011030	587#	911						
DELAY	013174	666#	675#	895#	957#	965	973#		
DEAR	012606	936	945#						
DISPLA	011046	597#							
DJ11	011000	576#							
DSFLG	020000	571#	605#						
DSSTAT	011060	612#							
ENTER	012346	879	888#						
EOP	012260	724	752	787	867	874#			
ERCSR	011054	610#	1056#	1078#					
ERRDR	011056	611#	906	1057#	1079#				
ERROR1	013552	1010#	1030#						
FLAG	011042	595#							
FRME	020000	571#	1076						
FULL.D	000001	624#	647	826	842	847	989	1045	
GO	011304	674#							
HO	000002	571#	649						
ILB	000010	571#	683						
IXDA	011020	583#	790	915	938	1053			
IXDA	011022	584#	791	914	942	944	975		
KSDIN	104416	571#	705	736	768	825			
LOOP	000030	571#	721	749	784	864			
MFULL	014076	1104#							
NOSC	000010	571#	644						
NOSC	012620	910	951#						
NOSC	012701	913	951#						
NOSC	012704	937	951#						
NOSC	012731	941	951#						
NOSC	012754	951#	978	1042					
NOSC	013106	951#							
NOSC	000400	571#	994	998	1004	1026			
NOSC	000200	571#	717	780	861				
NOSC	014012	1082	1084	1086#					
NOSC	013522	1019	1021	1023#					
ORUN	040000	571#	1076						
OMI	000002	571#	680						
OMO	000001	571#	677						

BOX	18	572	599	625	961	1036													
DCPARN	18																		
DNDOCI	18																		
DNPARN	18																		
DJPARN	18	560																	
DL PARN	18																		
DP PARN	18																		
DNDOCI	18																		
DNPARN	18																		
DL PARN	18																		
DP PARN	18																		
DNDOCI	18																		
DNPARN	18																		
DZPARN	18																		
HELLO	18																		
MLT	5718	652	713	745	777	807	839	855	947	1013	1069	1075	1093	1094					
SECURT	18	571																	
SINTF	18	571																	
SITEP	18	667																	
SSERV	18	614																	

. ABS. 014214 000

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

DZDJDB.SQB/SOL/CRF/ML:TOC=ITEP1.MAC,DZDJDB.P11
RUN-TIME: 10 13 .8 SECONDS
RUN-TIME RATIO: 49/26=1.8
CORE USED: 16K (31 PAGES)