

BB11-D

DMA INTERFACE
MD-11-DZKDA-A

EP-DZKDA-A-DL
COPYRIGHT 1974
FICHE 1 OF 1

JUN 1978
digital
MADE IN USA

This microfiche card contains 50 frames of technical data, arranged in a 10x5 grid. The frames contain various types of information, including:

- Textual data and code listings.
- Tables with multiple columns and rows of numerical or alphanumeric values.
- Waveform plots and graphs.
- Diagrams and flowcharts.
- Small tables and lists.

The data is presented in a structured, grid-like format typical of microfiche storage.



IDENTIFICATION

PRODUCT CODE: MAINDEC 11-D2KDA-A-D
PRODUCT NAME: RB11-D DMA INTERFACE
DATE CREATED: APRIL 19, 1974
MAINTAINED BY: DIAGNOSTIC ENGINEERING
AUTHOR: MIKE STURAK

COPYRIGHT (C) 1974, BY DIGITAL EQUIPMENT CORPORATION, MAYNARD,
MASS. 01754.

" THE MATERIAL IN THIS DOCUMENT IS FOR INFORMATION PURPOSES
ONLY, AND IS SUBJECT TO CHANGE WITHOUT NOTICE. DIGITAL EQ-
UIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OF
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY IT. DIGITAL
EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ER-
RORS WHICH MAY APPEAR IN THE DOCUMENT. "

1. ABSTRACT
2. REQUIREMENTS
 - 2.1 EQUIPMENT
 - 2.2 STORAGE
 - 2.3 CONFIGURATIONS
 - 2.3.1 STARTING ADDRESS 200 (DIAGNOSTIC TEST)
 - 2.3.2 STARTING ADDRESS 202 (CONVERSATION EXERCISE)
 - 2.4 PRELIMINARY PROGRAMS
3. LOADING PROCEDURE
4. STARTING PROCEDURE
5. PROGRAM AND/OR OPERATOR ACTION
 - 5.1 DATA SWITCH SETTINGS
 - 5.2 MISCELLANEOUS PRINTOUTS
 - 5.3 NON RECOVERABLE PROGRAM HALTS
 - 5.4 RESTRICTIONS
 - 5.5 LOOP ON TEST OK
6. ERRORS
 - 6.1 OVERVIEW
 - 6.2 PRINTOUT FORMAT
 - 6.3 RECOVERY
 - 6.4 NO ERROR TIMEOUT (PARAMETER RECOVERY)
7. "TRAP" TRAP DISCUSSION
 - 7.1 ERROR
 - 7.2 SCOPE
 - 7.3 SUBSCOPE
 - 7.4 FUNCTIONAL DIAGRAM
8. DISCUSSION
 - 8.1 CSR BIT DEFINITIONS
9. TEST DESCRIPTIONS
 - 9.1 DIAGNOSTIC TEST DESCRIPTIONS
 - 9.1.1 TEST 0
 - 9.1.2 TEST 2
 - 9.1.3 TEST 3
 - 9.1.4 TEST 5
 - 9.1.5 TEST 7
 - 9.1.6 TEST 8
 - 9.1.7 TEST 9
 - 9.1.8 TEST 10
 - 9.1.9 TEST 11
 - 9.1.10 TEST 12
 - 9.1.11 TEST 13
 - 9.1.12 TEST 14
 - 9.1.13 TEST 15
 - 9.1.14 TEST 16
 - 9.1.15 TEST 17
 - 9.1.16 TEST 18
 - 9.1.17 TEST 19
 - 9.2 SPECIAL CONVERSATION EXERCISE DESCRIPTION (SA 202)
10. LISTING

1. ABSTRACT

THIS PROGRAM CONTAINS VARIOUS TESTS DESIGNED TO COMPLETELY DIAGNOSE THE BB11-D DIRECT MEMORY ACCESS DEVICE WHEN STRUCTURED IN "LOOP BACK" MODE (DIAGNOSTIC CABLE FROM LOCATION D₁ (M9100) J₁ TO EF01 (M1902) AND STARTED AT PROGRAM LOCATION 200.

WHEN THE BB11-D (KIT) IS IN "LOOP BACK" CONFIGURATION, THE FOLLOWING SIGNALS ARE EQUIVALENT.

(CSR)	SIGNAL NAME	(CSR)	SIGNAL NAME
0	LAST XFER	12	XOF (EXTERNAL OVERFLOW)
2	FNCT	3	STATUS
11	DA OUT	10	INPUT (DA IN=NOT IN)
	DATA ACCEPTED	7	READY

THE FOLLOWING TESTS ARE PERFORMED UPON THE BB11-D, AND ARE LISTED IN THE ORDER IN WHICH THEY ARE EXECUTED (SA 200).

- TEST 0 - " INIT "
- TEST 1 - CSR BITS) 2 (FNCT), AND 3 (STAT)
- TEST 2 - CSR BITS) 0 (IN), AND 7 (RDY)
- TEST 3 - " DATA AVAILABLE OUT "
- TEST 4 - DATA BUFFER REGISTER WRAP AROUND
- TEST 5 - DATA OUTPUT (PART 1)
ONE NBR (OUT) W/O WORD COUNT OVERFLOW
- TEST 6 - DATA OUTPUT (PART 2)
ONE NBR OUTPUT WITH WORD COUNT OVERFLOW
- TEST 10 - INPUT
- TEST 11 - OUTPUT/INPUT (SIMULTANEOUS)
- TEST 12 - DATA VERIFICATION (PART 1)
- TEST 13 - DATA VERIFICATION (PART 2)
- TEST 14 - INTERRUPT ENABLE (PART 1) NO INTERRUPT EXPECTED
- TEST 15 - INTERRUPT ENABLE (PART 2) FROM WORD COUNT OVERFLOW
- TEST 16 - INTERRUPT ENABLE (PART 3) FROM SPECIAL CONDITION
- TEST 17 - INTERRUPT ENABLE (PART 4) PRIORITY LEVEL OF KIT
- TEST 18 - INTERRUPT ENABLE (PART 5) PRIORITY LEVEL 5 VERIFICATION.
- TEST 19 - LAST TRANSFER - EXTERNAL OVERFLOW

THIS PROGRAM ALSO CONTAINS A SPECIAL CONVERSATION EXERCISE DESIGNED TO ASYNCHRONOUSLY TRANSFER DATA (OUT/IN) WHEN STARTED AT PROGRAM LOCATION 202.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11
BB11-D (KIT)
DIAGNOSTIC "LOOP BACK" CABLE

2.2 STORAGE

THIS PROGRAM REQUIRES 4K.

2.3 CONFIGURATION

2.3.1 CONFIGURATION FOR STARTING ADDRESS 200 (DIAGNOSTIC)

CONNECT THE DIAGNOSTIC "LOOP BACK" CABLE FROM BB11-D
LOGIC LOCATION D1 (M9100) J1 TO EF0₁ (M1502). DISCONNECT
ANY CABLE FROM D1 (M9100) J2.

2.3.2 CONFIGURATIONS FOR STARTING ADDRESS 202 (CONVERSATION EXERCISE)

- (A). THE SAME AS PARAGRAPH 2, 3, 1, OR
- (B). CONNECT TO LOGIC LOCATION OF "THIS" BB11-D, D1
(M9100) J2, THE CABLE FROM THE "OTHER" BB11-D
EF01 (M1502). DISCONNECT ANY CABLE FROM D1
(M9100) J1 OF ALL BB11-D'S.

2.4 PRELIMINARY PROGRAMS

ALL DIAGNOSTICS FOR THE MAINTENANCE OF THE PDP-11 SHOULD HAVE
BEEN RUN IN THEIR ENTIRETY SOME TIME PRIOR TO THE EXECUTION
OF THIS DIAGNOSTIC.

3. LOADING PROCEDURE

THIS PROGRAM IS IN .ABS FORMAT. FOLLOW THE DIRECTIONS FOR THE
PARTICULAR ABSOLUTE LOADER BEING USED.

4. STARTING PROCEDURE

STARTING ADDRESS 200 IS THE BB11-D DIAGNOSTIC.
STARTING ADDRESS 202 IS THE SPECIAL CONVERSATION EXERCISE.

- (A) PLACE WITHIN PROGRAM LOCATION 1200 THE OCTAL EQUIVALENT
OF THE BB11-D DEVICE ADDRESS (IF NOT 160000).
- (B) CONFIGURE THE DATA SWITCHES TO REPRESENT THE OCTAL
EQUIVALENT OF THE STARTING ADDRESS 200, OR 202.
- (C) PRESS "LOAD ADDRESS". PRESS "START".

THE PROGRAM IS NOW RUNNING.

5. PROGRAM AND/OR OPERATOR ACTION

5.1 DATA SWITCH SETTINGS

- 15 = 0 - HALT UPON DETECTION OF ERROR
- 14 = 0 - SCOPE LOOP ON ERROR (LATCHED BY FIRST ERROR)
- 13 = 1 - HALT AT END OF EACH PROGRAM PASS (SA 200), OR
 = HALT AT END OF RECEIVE DATA COMPARE (SA 202)
- 12 = 0 - PRINT ERROR INFORMATION AT DETECTION
- 11 = 1 - BURST MODE TRANSMISSIONS (SA 202)

5.2 MISCELLANEOUS PRINTOUTS

THE PROGRAM WILL PRINT ON THE TELEPRINTER THE INTRODUCTORY MESSAGE "TOGGLE INTO PROGRAM LOCATION XXXX THE BB11-D DEVICE ADDRESS 00 IF NOT 100000" AFTER THE FIRST PROGRAM START 200.

THE PROGRAM WILL INFORM THE ATTENDANT OF THE BB11-D VECTOR ADDRESS BY PRINTING ON THE TELEPRINTER THE MESSAGE "BB11-D VECTOR ADDRESS IS XXXX" UPON FIRST DETECTION OF A BB11-D DEVICE INTERRUPT AFTER START 200, OR START 202.

THE PROGRAM WILL PRINT THE MESSAGE "WAITING FOR OTHER KIT" (STARTING ADDRESS 202) EACH TIME THE "OTHER" KIT BECOMES, OR FAILS TO BECOME READY; (STATUS BIT 03 = 1 INDICATES THE KIT IS READY).

THE "OTHER" KIT IS NOT READY WHEN THE DATA SWITCH 13=1, OR 15=1, OR WHEN A BB11-D ERROR HAS BEEN DETECTED AND DATA SWITCH 12=0.

5.3 NON RECOVERABLE PROGRAM HALTS

PROGRAM LOCATION	DESCRIPTION
2	UNASSIGNED TRAP
6	STACK POINTER OVERFLOW
12	RESERVED INSTRUCTION TRAP
16	TRACE TRAP
26	POWER FAILURE
32	ENT TRAP
66	UNEXPECTED TELEPRINTER INTERRUPT

5.4 RESTRICTIONS (APPLICABLE ONLY FOR STARTING ADDRESS 202)

THE PROGRAM WILL CHANGE WAITING FOR CSR BIT 7 (READY) IF THE HARDWARE CONFIGURATION IS THAT DESCRIBED WITHIN PARAGRAPH 2.3.2 (A), AND DATA SWITCH 11 (BURST MODE) IS POSITIONED TO REPRESENT A BINARY 1. THIS IS ALSO TRUE IF THE HARDWARE CONFIGURATION IS THAT DESCRIBED WITHIN PARAGRAPH 2.3.2 (B), AND DATA SWITCH 11 OF BOTH POP-11'S ARE CONFIGURED TO REPRESENT A BINARY 1.

6. ERRORS

6.1 OVERVIEW

THE FOLLOWING ERRORS ARE LISTED IN THE ORDER IN WHICH THEY MAY BE DETECTED.

E1	MC NOT 0 AFTER INIT
E2	BAR NOT 0 AFTER INIT
E3	DBR NOT 0 AFTER INIT
E4	CSR NOT = X6 1 0 AFTER INIT
E10	CSR NOT = X2 0 4
E12	CSR NOT = X2 1 0
E20	CSR NOT = X 1 1
E21	MC NOT = 177776
E22	BAR NOT = 0
E23	CSR NOT = X2 1 0
E30	CSR NOT = 126410
E31	MC NOT = 177776
E34	DBR NOT = 0
E32	BAR NOT = 0
E33	CSR NOT = 122610
E35	CSR NOT = 610
E38	THE CONTENTS OF THE DBR NOT = CONTENTS OF RB
E37	CSR NOT = 126412
E60	MC NOT = 177777
E61	BAR NOT = #BUFFER*2
E62	CSR NOT = 127410
E63	MC NOT = 0
E64	BAR NOT = #BUFFER*2
E65	CSR NOT = 122610
E70	CSR NOT = 11
E72	DBR NOT = #BUFFER*2
E74	MC NOT = 177777
E100	CSR NOT = 210
E102	BAR NOT = #BUFFER*4
E110	CSR NOT = 210
E112	DATA TRANSMITTED NOT = DATA RECEIVED
E116	CSR NOT = 210
E120	DATA TRANSMITTED NOT = DATA RECEIVED
E131	AN UNEXPECTED BB11-D INTERRUPT OCCURED
E130	CSR NOT = X3 1 0
E132	CSR NOT = X2 1 0
E140	NO BB11-D INTERRUPT OCCURED
E142	CSR NOT = X3 1 0
E143	NO BB11-D INTERRUPT OCCURED
E144	CSR NOT = 1 2 2 X1 1 0
E146	NO BB11-D INTERRUPT OCCURED
E152	KIT NOT ON PL- 5 (UNEXPECTED INTERRUPT)
E160	CSR NOT = 127410
E162	CSR NOT = 110210
E164	CSR NOT = 210
E202	OUTPUT MUST HAVE ADDRESSED " NEX "
E206	INPUT MUST HAVE ADDRESSED " NEX "
E210	DATA ERROR (DATA "IN" NOT = DATA EXPECTED)
*E376	UNEXPECTED BB11-D INTERRUPT (MAY OCCUR ANYTIME)

NOTE: CSR BIT "X" IS CSR BIT 00 (LAST DIRECTION) MAY REPRESENT A BINARY 1 OR 0.

6.2 ERROR TYPEOUT FORMAT

E (ERRORX) SCOPE (PCSCOPE) PASS (PASSLSB) ERROR (ERRORS)

BUFFER (BADDRESS) EXPECTED (GOOD) ACTUAL (BAD)

THE FOLLOWING INFORMATION IS PRINTED ON DETECTION OF A BB11-D ERROR IF DATA SWITCH 12 REPRESENTS A BINARY 0,

E ERRORX - THE E# CORRESPONDING TO AN ERROR DESCRIBED WITHIN THE PRECEEDING ERROR ABSTRACT,

SCOPE PCSCOPE - THE OCTAL EQUIVALENT TO THE FIRST ADDRESS OF THE LATCHED SCOPE LOOP,

PASS PASSLSB - PROGRAM PASS COUNTER FOR SA 280 (1-65534(10)),

ERROR ERRORS - ERROR COUNTER WITHIN A SCOPE LOOP (1-65534(10)).

THE FOLLOWING INFORMATION IS AMENDED TO THE ABOVE PRINTOUT ON DETECTION OF A BB11-D DATA ERROR,

BUFFER BADDRESS- THE OCTAL ADDRESS CONTAINING THE DATA IN ERROR

EXPECTED GOOD - THIS OCTAL EQUIVALENT REPRESENTS THE DATA EXPECTED,

ACTUAL BAD - THIS OCTAL EQUIVALENT REPRESENTS THE ACTUAL DATA (RESULT OF DATA TEST),

6.3 ERROR RECOVERY

DATA SWITCH	BINARY REPRESENTATION
-------------	-----------------------

12	0
----	---

UPON THIS PROGRAMS DETECTION OF A BB11-D IRREGULARITY, AN ERROR MESSAGE WILL ENSUE ON THE TELEPRINTER (IF DATA SWITCH 12 = 0)...

15	0
----	---

...THEN THE PDP WILL HALT (IF DATA SWITCH 15 = 0)...

14	0
----	---

...THE FIRST BB11-D ERROR DETECTED LATCHES THIS PROGRAM INTO A SCOPE LOOP (THE FIRST ADDRESS OF THAT LOOP BEING THE CONTENTS OF PROGRAM LOCATION PCSCOPE) EVEN IF THE ERROR IS INTERMITTANT AS WOULD BE INDICATED BY THE ERROR COUNTER INCREMENTING, (THE CONTENTS OF PROGRAM LOCATION ERRORS).

THE OPERATOR MUST PLACE THE DATA SWITCH 14 IN THE POSITION REPRESENTING A " 1 " WHICH WOULD ABORT THE LATCHED SCOPE LOOP AND CONTINUE THIS PROGRAM (AS THOUGH NO ERROR WAS EVER DETECTED).

6.4 NO TYPEOUT (ERROR PARAMETER RECOVERY)

IF AN ERROR HAS OCCURED, BUT THE STATE OF DATA SWITCH 12 IS REPRESENTING A BINARY 1 (DISABELING ANY ERROR TYPEOUT), OR THE SYSTEM CONFIGURATION HAS NO "HARD COPY" OUTPUT DEVICE FOR THE DEPOSIT OF ANY ERROR INFORMATION, THEN PLACE DATA SWITCH 15 TO REPRESENT A BINARY 0 (HALTING THE PDP AT THE DETECTION OF A BB11-D ERROR) AND EXAMINE FOR THE CONTENTS OF THE PROGRAM LOCATIONS DESCRIBED WITHIN PARAGRAPH 6.2 (ERROR TYPEOUT FORMAT) FOR THAT SPECIFIC INFORMATION RELATING TO THAT ERROR CONDITION.

IN SUMMATION, EXAMINING FOR THE CONTENTS OF PROGRAM LOCATION "ERRORX" WOULD REVEAL THE E₀, AND OF "PCSCOPE", THE FIRST ADDRESS OF THE LATCHED SCOPE LOOP, AND ETC., ETC., ETC.,

7. "TRAP" TRAP DISCUSSION

THE FOLLOWING PARAGRAPHS (THRU 7.3) ARE ATTEMPTED AS A SIMPLIFIED FUNCTIONAL OUTLINE OF THIS PROGRAMS ERROR, AND AUTOMATIC SCOPE LOOPING PROCEDURES,

7.1

THE MNUMONICS "ERROR", "SCOPE" AND "SUBSCOPE" ARE "TRAP" TRAPS WHICH VECTOR THROUGH PROGRAM LOCATION 34 TO TRAPROUTINE "ERRORCATCHER" WHERE THE INTERROGATION OF THE IDENTIFIER "X" (CODE 104400*X,) WHERE X = 0 TO 377 OCTAL, PROCEEDS TO DETERMINE WHICH SUB-TRAP-ROUTINE IS EXECUTED. THE GENERATED ASSEMBLY CODE FOR THE MNUMONIC "SCOPE" IS 104400 ("TRAP" TRAP), FOR THE MNUMONIC "SUBSCOPE" IS 104777, AND FOR THE MNUMONIC "ERROR" IS 104400*X, WHERE X = 1 TO 376.

- (A) IF THE CONTENTS OF PROGRAM LOCATION ERRORX IS EQUIVALENT TO THE OCTAL VALUE OF:
 - (1) 0, GOTO SUB-TRAP-ROUTINE "XSCOPE" (PARAGRAPH 7.2), OR
 - (2) 377, GOTO SUB-TRAP-ROUTINE "XSSCOPE" (PARAGRAPH 7.3),
- (B) THE CONTENTS OF PROGRAM LOCATION "ERRORX" IS EQUIVALENT TO THE OCTAL VALUE BETWEEN 1 AND 376. THE CONTENTS OF PROGRAM LOCATION "PCSSCOPE" IS EQUIVALENT TO THE ADDRESS OF THE PROGRAM LOCATION+2 CONTAINING THE MNUMONIC "SCOPE", OR "SUBSCOPE" THEREFORE THE CONTENTS OF THE PROGRAM LOCATION "ERRORS" IS INCREMENTED.
- (C) IF THE CONTENTS OF PROGRAM LOCATION "ERRORX" IS EQUIVALENT OF 200 THRU 277, CSR BIT 2 (FNCT) IS CLEARED (SA 202).
- (D) IF THE DATA SWITCH 12 = 1, GOTO PARAGRAPH (E), OTHERWISE AN APPROPRIATE ERROR MESSAGE IS PRINTED (REFER TO PARAGRAPH 6.2 FOR "ERROR TYPEOUT FORMAT "),
- (E) IF DATA SWITCH 15 = 0, THE PDP IS HALTED.
- (F) RTI (FROM ERROR+X "TRAP" TRAP)

7.2 SCOPE (SUB-TRAP-ROUTINE "XSCOPE")

- (G) IF THE CONTENTS OF PROGRAM LOCATION ERRORS IS NOT EQUIVALENT TO 0, GOTO PARAGRAPH (J).
- (H) NO ERRORS HAVE BEEN DETECTED, OR DATA SWITCH 14 = 1 ABORTING THE SCOPE LOOP ALREADY IN PROGRESS. REPLACE THE CONTENTS OF PROGRAM LOCATION ERRORS WITH 0, AND REPLACE THE CONTENTS OF PROGRAM LOCATION PCSSCOPE WITH THE CONTENTS OF THE CONTENTS OF THE STACK POINTER (RETURN PC) FOR LATER USE AS THE FIRST ADDRESS OF A SCOPE LOOP IF AN ERROR IS DETECTED.
- (I) RTI (FROM SCOPE "SUB-TRAP" TRAP)
- (J) THIS IS A SCOPE LOOP. IF THE DATA SWITCH 14 = 1, GOTO "NOSCOPE", (PARAGRAPH H).
- (K) REPLACE THE CONTENTS OF THE CONTENTS OF THE STACK POINTER WITH THE CONTENTS OF PROGRAM LOCATION "PCSSCOPE" (1ST LOCATION OF THE SCOPE LOOP).
- (L) RTI (TO LATCHED SCOPE LOOP)

7.3 SUBSCOPE (SUB-TRAP-ROUTINE "XSSCOPE")

IF THE CONTENTS OF PROGRAM LOCATION "ERRORS" = 0, AN RTI IS PERFORMED.

SOME TYPE OF BB11-D ERROR EXISTS, IF THE RETURN PC (THE CONTENTS OF THE CONTENTS OF THE STACK POINTER) IS EQUIVALENT TO THE CONTENTS OF PROGRAM LOCATION PCSSCOPE, THEN THIS BB11-D ERROR IS THE ERROR WHICH IS THE ADJITATION OF THE SCOPE LOOP, AND NOT SOME PREVIOUS ERROR WHICH OCCURED WHILE ALREADY WITHIN THE SCOPE LOOP, THEREFORE GOTO PARAGRAPH (J) "SCOPING", OTHERWARDS AN RTI IS PERFORMED.

7.4 FUNCTIONAL DIAGRAM

THIS PARAGRAPH IS INTENDED TO RENDER AN OVERVIEW OF THE FUNCTIONAL DESCRIPTION OF PARAGRAPHS 7.1 THRU 7.3 .

" T " INDICATES THE START OF A PARTICULAR BB11-D TEST,
" E " SIGNIFIES AN ERROR "TRAP" TRAP.

TA1	SCOPE	; NOTE #1
	<<<< SUBTEST 1 OF A >>>>	
E11	ERROR+X SUBSCOPE	; NOTE #2
	<<<< SUBTEST 2 OF A >>>>	
E21	ERROR+X SUBSCOPE	; NOTE #2
	<<<< SUBTEST 3 OF A >>>>	
E31	ERROR+X	
TB1	SCOPE	; NOTE #3
	<<<< SUBTEST 1 OF B >>>>	
E41	ERROR+X SCOPE	; NOTE #4
	<<<< SUBTEST 2 OF B >>>>	
E51	ERROR+X	
	...ETC..	

NOTE #1: THIS IS THE INITIAL ENTRY INTO PROGRAM CODE "XSCOPE" WHICH SETS THE ADDRESS OF "TA+2" (RETURN PC) INTO PROGRAM LOCATION "PCSSCOPE" FOR SCOPE LOOPING FOR THIS TEST.

NOTE #2: IF A BB11-D ERROR EXISTS, SUB-SCOPE TO PROGRAM LOCATION TA+2.

NOTE #3: IF A BB11-D ERROR EXISTS, SCOPE LOOP TO PROGRAM LOCATION TA+2. OTHERWISE, THIS IS THE INITIAL ENTRY INTO PROGRAM CODE "XSCOPE" ...FOR THIS TEST.

NOTE #4: IF A BB11-D ERROR EXISTS, SCOPE LOOP TO PROGRAM LOCATION TB+2.

8. DISCUSSION

8.1 CSR BIT ALLOCATIONS

THE CONTROL/STATUS REGISTER (CSR) IS THE ONLY BYTE ADDRESSABLE REGISTER. WRITING INTO THE LOW BYTE (OR INTO THE ENTIRE WORD) WILL AUTOMATICALLY CLEAR "SPECIAL CONDITION", AND "NEX" AND/OR "XOF" IF PREVIOUSLY SET.

CSR	MNEMONIC	
0	IN	R/W
1	OUT	R/W
2	FNCT	R/W
3	STAT	R
4	RESERVED	
5	RESERVED	
6	END	R/W
7	READY	R
8	LAST DIRECTION	R
9	LAST TRANSFER	R
10	INPUT	R/WRITE 0
11	DATA AVAILABLE (OUT)	R/WRITE 0
12	XOF (EXTERNAL OVFL)	R/CLEARED BY WRITING LOW BYTE OF CSR
13	INPUT DEMAND	R
14	NEX	R/CLEARED BY WRITING LOW BYTE OF CSR
15	SPECIAL CONDITION	R/CLEARED BY WRITING LOW BYTE OF CSR

PROGRAMMING NOTE: CSR BITS 10 (INPUT) AND 11 (DATA AVAILABLE OUT) MAY BE CLEARED BY WRITING A 0 INTO THAT PARTICULAR BIT. THEREFORE, WHENEVER ADDRESSING THE CSR, AND THE STATE OF CSR BITS 10, AND 11 ARE IMPORTANT, WRITE THESE BITS TO A 1. (WRITING THESE BITS TO A 1 DOES NOT SET THEM, IT ONLY ASSURES THAT THEY WON'T BE CLEARED).

9. TEST DESCRIPTIONS

9.1 DIAGNOSTIC TEST DESCRIPTIONS

THE FOLLOWING PARAGRAPHS ARE ATTEMPTED TO SUMMARIZE EACH DIAGNOSTIC TEST. THE E'S WITHIN THE <> ARE INTENDED TO INDICATE WHICH ERRORS MAY BE DETECTED.

9.1.1 TEST 0 (T0) - "INIT"

THIS PROGRAM ISSUES THE PDP-11 "RESET" INSTRUCTION, THEN TESTS THE BB11-D WORD COUNT REGISTER TO = 0, THE BUS ADDRESS REGISTER TO = 0, THE DATA BUFFER REGISTER TO = 177777, AND FINALLY THE CSR TO = X2 1 0 (WHERE X=BIT 0 LAST DIRECTION = 1 OR 0).

"RESET" WILL BE ISSUED WITHIN THE SCOPE LOOP.

<E1, E2, E3, E4>

9.1.2 TEST 2 (T2) - CSR BITS 2 (FNCT), AND 3 (STAT)

THIS PROGRAM FLIPS BIT 2 (FNCT) OF THE CSR TESTING BIT 3 (STAT) TO BE THE COMPLEMENT OF THE "FNCT" BIT.

<E10, E12>

9.1.3 TEST 3 (T3) - CSR BITS 0 (IN), AND 7 (RDY)

THE BB11-D IS "READY" WHEN "IN", AND "OUT" BOTH = 0, AND THERE IS NO "DATA AVAILABLE (OUT)".

THIS PROGRAM FLIPS THE CSR BIT 0 (IN), AND TESTS THE CSR BIT 7 (RDY) TO = 0.

NO INTERRUPTS OF DMA'S ARE EXPECTED, THEREFORE THE WORD COUNT REGISTER AND BUS ADDRESS REGISTER SHOULD REMAIN UNCHANGED.

<E20, E21, E22, E23>

9.1.4 TEST 5 (T5) - DATA AVAILABLE OUT

THIS PROGRAM WRITES INTO THE BB11-D'S DATA BUFFER REGISTER WHICH SHOULD RAISE "DATA AVAILABLE OUT".

THE COMMAND STATUS REGISTER SHOULD = 126410,
THE WORD COUNT REGISTER SHOULD STILL = 177776,
THE DATA BUFFER REGISTER SHOULD = 0,
THE BUS ADDRESS REGISTER SHOULD STILL = 0.

15 SPECIAL CONDITION (1 BECAUSE "INPUT DEMAND")
13 INPUT DEMAND (1 BECAUSE "INPUT"=1, AND "IN"=0)
11 DATA AVAILABLE OUT (1 BECAUSE OF WRITING INTO THE DBR)
10 INPUT (1 BECAUSE "DATA AVAILABLE OUT"
0 LAST DIRECTION (1=OUT)
7 READY (0 BECAUSE "IN"=0, "OUT"=0, AND DATA AVAILABLE =1)
3 STATUS (1 BECAUSE FNCT=0)

<E30, E31, E34, E32, E33, E35>

9.1.5 TEST 7 DATA BUFFER (CABLE WRAP AROUND)

WITH THE DIAGNOSTIC "WRAP AROUND" CABLE CONNECTED FROM BB11-D LOGICAL LOCATION D1 (M9100) J1 TO EF01 (M1502), WRITING INTO THE DATA BUFFER WILL PLACE THAT DATA ONTO THE DATA LINES (OUT), THEREFORE, THE CONTENTS OF THE DATA BUFFER SHOULD BE EQUIVALENT TO THE DATA OUT.

<E50>

9.1.6 TEST 8 OUTPUT (PART 1)

THIS TEST ENABLES FOR A 2 WORD DATA TRANSFER OUT, BUT BECAUSE DATA IS NOT ACCEPTED, THERE SHOULD BE NO WORD COUNT OVERFLOW.

THE CSR SHOULD = 126412;
THE WORD COUNT REGISTER SHOULD = 177777,
THE BUS ADDRESS REGISTER SHOULD = #BUFFER+2.

<E57, E60, E61>

9.1.7 TEST 9 OUTPUT (PART 2)

THIS TEST ENABLES FOR A 1 WORD DATA TRANSFER OUT EXPECTING A WORD COUNT OVERFLOW.

THE CSR SHOULD = 127410;
THE WORD COUNT REGISTER SHOULD HAVE OVERFLOWED TO 0,
THE BUS ADDRESS REGISTER SHOULD = #BUFFER+2.

THIS TEST THEN CLEARS CSR BIT 11 (DATA AVAILABLE OUT), WHICH CLEARS CSR BIT 9 (LAST TRANSFER), AND BECAUSE CSR BIT 2 (IN), AND 1 (OUT) WERE BOTH PREVIOUSLY = 0, CSR BIT 7 (READY) BECOMES SET.

THE CSR SHOULD = 122610;

<E65>

9.1.8 TEST 10 INPUT

THIS TEST ENABLES BIT 0 (IN) THEN RAISES DATA AVAILABLE OUT BY WRITING INTO THE BB11-D DATA BUFFER REGISTER, ONE DMA TRANSFER SHOULD OCCUR.

THE CSR SHOULD = 11;
THE BUS ADDRESS REGISTER SHOULD = #BUFFER+2,
THE WORD COUNT REGISTER SHOULD = 177777.

<E70, E72, E74>

TEST 11 OUTPUT/INPUT (SIMULTANEOUSLY)

THIS TEST ENABLES BOTH CSR BITS 0 (IN), AND 1 (OUT), TWO DMA TRANSFERS SHOULD OCCUR (1 OUT, AND 1 IN),

THE CSR SHOULD = 210V
THE BUS ADDRESS REGISTER SHOULD = 0BUFFER+4,

<E100, E102>

9.1.10 TEST 12 DATA TEST (PART 1)

THIS TEST ENABLES FOR 34 DMA TRANSFERS (17 OUT, AND 17 IN) THEN COMPARES DATA TO THE FOLLOWING EXPECTED PATTERN!

(FIRST)	0	1	2	4	10
	2	4	100	200	400
	1000	2000	4000	10000	20000
	40000	100000	(LAST WORD)		

<E110, E112>

9.1.11 TEST 13 DATA TEST (PART 2)

THIS TEST ENABLES FOR 64 DMA TRANSFERS (32 OUT, AND 32 IN) COMPARING DATA WITH THE EXPECTED WORST CASE PATTERN OF A COMPLEMENTING FLOATING " 1 ".

(FIRST)	-2, 1	-3, 2	-5, 4
	-11, 10	-21, 20	-41, 40
	-101, 100	-201, 200	-401, 400
	-1001, 1000	-2001, 2000	-4001, 4000
	-10001, 10000	-20001, 20000	-40001, 40000
	-100001, 100000	(LAST WORD)	

<E116, E120>

9.1.12 TEST 14 INTERRUPT ENABLE (PART 1)

THIS TEST FLIPS CSR BIT 6 (INTERRUPT ENABLE) EXPECTING NO DEVICE INTERRUPT.

<E131, E138, E132>

9.1.13 TEST 15 INTERRUPT ENABLE (PART 2)

THIS TEST ENABLES THE BB11-D EXPECTING A DEVICE INTERRUPT (FROM SPECIAL CONDITION WORD COUNT OVERFLOW) AT ANY PRIORITY LEVEL FOR NOW.

<E148, E142>

9.1.14 TEST 16 INTERRUPT ENABLE (PART 3)

THIS TEST ENABLES THE BB11-D EXPECTING A DEVICE INTERRUPT FROM SPECIAL CONDITION INPUT DEMAND AT ANY PRIORITY LEVEL FOR NOW.

<E143, E144>

9.1.15 TEST 17 BB11-D INTERRUPT ENABLE PRIORITY LEVEL (PART 4)

THIS TEST ENABLES THE BB11-D EXPECTING A DEVICE INTERRUPT FROM SPECIAL CONDITION AT PRIORITY LEVELS 5, 6, OR 7 FOR NOW.

<E146>

9.1.16 TEST 18 INTERRUPT ENABLE (PART 5)

THIS TEST ENABLES THE BB11-D NOT EXPECTING A DEVICE INTERRUPT AT PRIORITY LEVELS 6, OR 7 BECAUSE THE POP PRIORITY IS 5.

<E152>

9.1.17 TEST 19 EXTERNAL OVERFLOW (XOF)

THIS IS A TEST FOR EXTERNAL OVERFLOW (XOF) CSR BIT 12. A ONE WORD DMA OUTPUT IS ENABLED EXPECTING A BB11-D WORD COUNT OVERFLOW AND LAST TRANSFER, THEN, WITHOUT CLEARING LAST TRANSFER, A ONE WORD DMA DATA INPUT IS ENABLED EXPECTING EXTERNAL OVERFLOW.

<E168, E162, E164>

9.2 SPECIAL CONVERSATION EXERCISE DESCRIPTION (STARTING ADDRESS 202)

BASICALLY, EACH KIT, WITH INTERRUPT ENABLE, WILL TRANSMIT TO THE "OTHER" KIT. UPON THE DETECTION OF "INPUT DEMAND", EACH KIT WILL DISABLE "OUT", AND ENABLE "IN", THUS INPUTTING DATA. IF AFTER 20US BETWEEN DMA INPUT TRANSFERS (TWICE THE MAXIMUM TRANSFER TIME OF 10US) THE INPUT TRANSFERS CEASE, EACH KIT WILL DISABLE "IN", AND RESUME TRANSMISSIONS TO THE "OTHER" KIT AT THE POINT AT WHICH THEY WERE SUSPENDED.

THE DATA INPUTTED IS COMPARED TO THE DATA EXPECTED AT THE DETECTION OF "EXTERNAL OVERFLOW". ANY ERROR MESSAGE, IF APPLICABLE, WILL ENSUE.

THIS ASYNCHRONOUS OPERATION MAY BE DEFERRED SOMEWHAT BY PLACING DATA SWITCH 11 TO REPRESENT A BINARY 1 THUS CAUSING A "BURST" TYPE OF OPERATION.

BASICALLY, EACH KIT, WITH INTERRUPT ENABLE WILL TRANSMIT TO THE "OTHER" KIT. UPON THE DETECTION OF "INPUT DEMAND", THE BINARY STATE OF DATA SWITCH 11 IS TESTED, AND IF = 1, "THIS" KIT WILL NOT DISABLE "OUT", BUT WILL CONTINUE TRANSMITTING UNTIL A WORD COUNT OVERFLOW OCCURS, THEN INPUT DEMAND IS RECOGNIZED AND APPROXIMATELY 1 DMA "IN" OCCURS.

IN SUMMARY, THE PDP WHOSE DATA SWITCH 11 REPRESENTS A BINARY 1 WILL IGNORE "INPUT DEMAND'S UNTIL ALL "N" WORDS ARE TRANSMITTED TO THE "OTHER" KIT, THEN 1 WORD "IN" IS ACCEPTED (APPROXIMATELY 1 OR 1.5X DEPENDING UPON THE LENGTH OF THE PROGRAM SEQUENCE IN WHICH THE "OTHER" KIT IS COMPARING ALL OF THOSE "N" DATA WORDS RECEIVED). THEREFORE, "THIS" KIT WOULD TRANSFER "N" DATA WORDS, "N" TIMES, IN WHICH "N" WORD COUNT OVERFLOWS WOULD OCCUR BEFORE "THIS" KIT DETECTED 1 EXTERNAL OVERFLOW (XOF) CAUSED BY A WORD COUNT OVERFLOW OF THE "OTHER" KIT. THE VALUE OF "N" IN THIS PROGRAM IS 32.

NOTE: REFERENCE PARAGRAPH 9.4 FOR OPERATIONAL DATA SWITCH 11 RESTRICTION.

10. LISTING

1
2
3

TITLE MAINDEC-11-DZKDA-A
ENABLE ABS
ENABLE AMA
.NLIST SEQ,LOC

1 TEST 0 = " INIT "

001434 E1 ; WC NOT 0 AFTER INIT
001440 E2 ; BAR NOT 0 AFTER INIT
001462 E3 ; DBR NOT 0 AFTER INIT
001474 E4 ; CSR NOT = X0 3 0 AFTER INIT

1 TEST 2 = CSR BITS: 2 (FNCT), AND 3 (STAT)

001514 E10 ; CSR NOT = X2 0 4
001532 E12 ; CSR NOT = X2 1 0

1 TEST 3 = CSR BITS: 0 (IN), AND 7 (RDY)

001572 E20 ; CSR NOT = X 1 1
001606 E21 ; WC NOT = 177776
001620 E22 ; BAR NOT = 0
001640 E23 ; CSR NOT = X2 1 0

1 TEST 5 = " DATA AVAILABLE OUT "

001702 E30 ; CSR NOT = 126410
001716 E31 ; WC NOT = 177776
001730 E34 ; DBR NOT = 0
001742 E32 ; BAR NOT = 0
001764 E33 ; CSR NOT = 122610
002006 E35 ; CSR NOT = 610

1 TEST 7 = DATA BUFFER REGISTER WRAP AROUND

002042 E50 ; THE CONTENTS OF THE DBR NOT = CONTENTS OF R0

1 TEST 8 = DATA OUTPUT (PART 1)
1 ONE NPR (OUT) W/O WORD COUNT OVERFLOW

002122 E57 ; CSR NOT = 126412
002136 E60 ; WC NOT = 177777
002152 E61 ; BAR NOT = #BUFFER+2

1 TEST 9 = DATA OUTPUT (PART 2)
1 ONE NPR OUTPUT WITH WORD COUNT OVERFLOW

002220 E62 ; CSR NOT = 127410
002232 E63 ; WC NOT = 0
002246 E64 ; BAR NOT = #BUFFER+2
002270 E65 ; CSR NOT = 122610

```
      | TEST 10 - INPUT
002336      E70      | CSR NOT = 11
002352      E72      | OBR NOT = #BUFFER+2
002366      E74      | WC NOT = 177777

      | TEST 11 - OUTPUT/INPUT
002436      E100     | CSR NOT = 210
002452      E102     | BAR NOT = #BUFFER+4

      | TEST 12 - DATA VERIFICATION TEST (PART 1)
002554      E110     | CSR NOT = 210
002602      E112     | DATA TRANSMITTED NOT = DATA RECIEVED
                   | THE CONTENTS OF BUFFER=X NOT = THE CONTENTS OF BLFFER=X+2

      | TEST 13 - DATA VERIFICATION (PART 2)
002712      E116     | CSR NOT = 210
002740      E120     | DATA TRANSMITTED NOT = DATA RECIEVED
                   | THE CONTENTS OF BUFFER=X NOT = THE CONTENTS OF BLFFER=X+2

      | TEST 14 - INTERRUPT ENABLE (NONE)
002774      E131     | AN UNEXPECTED BB11-D INTERRUPT OCCURED
003006      E130     | CSR NOT = X3 1 0
003026      E132     | CSR NOT = X2 1 0

      | TEST 15 - INTERRUPT ENABLE (EXPECTING AN INTERRUPT FROM WCO)
003112      E140     | NO INTERRUPT OCCURED
003124      E142     | CSR NOT = X3 1 0

      | TEST 16 - INTERRUPT ENABLE (FROM SPECIAL CONDITION)
003200      E143     | NO BB11-D INTERRUPT OCCURED
003212      E144     | CSR NOT = 1 2 2 X1 1 0

      | TEST 17 - PRIORITY LEVEL OF KIT
003276      E146     | NO INTERRUPT OCCURED

      | TEST 18 - PRIORITY LEVEL 5 VERIFICATION
003364      E152     | KIT NOT ON PL# 5 (UNEXPECTED INTERRUPT)

      | TEST 19 - LAST TRANSFER - EXTERNAL OVERFLOW (XOF)
003432      E160     | CSR NOT = 127410
003462      E162     | CSR NOT = 110210
003502      E164     | CSR NOT = 210
```

; STARTING ADDRESS 2P2
; CONVERSATION BETWEEN 2 OR MORE BB11-D'S

003774 E202 ; SPECIAL CONDITION = 1, MUST BE " NEX "

004176 E206 ; SPECIAL CONDITION = 1 (NOT " XOF ")
; MUST BE " NEX "

004250 E210 ; DATA ERROR

110 .LIST SEQ,LOC

120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154

DATA SWITCH DEFINITIONS

```
100000 SW15=100000  
040000 SW14=40000  
020000 SW13=20000  
010000 SW12=10000  
004000 SW11=4000  
ISW10=2000  
ISW9=1000  
SW8=400  
ISW7=200  
ISW6=100  
ISW5=40  
ISW4=20  
ISW3=10  
ISW2=4  
ISW1=2  
ISW0=1
```

DATA SWITCH OPERATIONAL DEFINITIONS

```
SW15 = 0 - HALT ON ERROR  
SW14 = 0 - SCOPE LOOP ON ERROR (LATCHED BY FIRST ERROR)  
SW13 = 1 - HALT AT THE END OF EACH PROGRAM PASS  
SW12 = 0 - PRINT ERROR INFORMATION AT DETECTION
```

SPECIAL DATA SWITCH FOR STARTING ADDRESS 202

BURST MODE

THIS BB11-D WILL NOT HONOR " INPUT DEMAND " UNTIL ALL DATA HAS BEEN TRANSMITTED

```
SW11 = 1 - BURST MODE TRANSMISSION
```

```

156          000000          .00
157
158 000000 000002          .02          ; UNASSIGNED TRAP
159 000002 000000          HALT          ;
160 000004 000006          .02          ; SP OVERFLOW, BUS ERROR TRAP
161 000006 000000          HALT          ;
162 000010 000012          .02          ; RESERVED INSTRUCTION TRAP
163 000012 000000          HALT          ;
164 000014 000016          .02          ; TRACE TRAP
165 000016 000000          HALT          ;
166
167          ; MACRO LOAD THE INTERRUPT VECTOR ADDRESSES (20-777) WITH THE " IOT " TRAP
168
169          ;REPT 174
170
171          .02
172          4          ; IOT (TRAPPED TO PREVIOUS ADDRESS)
173
174          ;ENDR
175
176          000020          .020
177
178 000020 004730          KITCATCHER          ; IOT TRAP (KIT INTERRUPTS VIA IOT)
179 000022 000340          340
180 000024 000026          .02          ; POWER FAIL TRAP
181 000026 000000          HALT          ;
182 000030 000032          .02          ; EMT TRAP
183 000032 000000          HALT          ;
184
185          000034          .034
186
187 000034 004342          ERRORCATCHER          ; TRAP TRAP
188 000036 000340          340
189
190          000064          .064
191
192 000064 000066          .02
193 000066 000000          HALT
194
195          000200          .0200
196
197 000200 000401          BR 15
198 000202 000402          BR 25
199
200 000204 000137 001254          151      JMP S200
201 000210 000137 003534          251      JMP S202
202
203          ; ,=774
204
205          ;          KITCATCHER          ; KIT 11/D VECTOR ADDRESS IF NO JUMPERS CUT
206          ;          340          ; ON THE M7021 MODULE

```

```

200          001200          .=1200
209
210          ; BB11-D REGISTER ADDRESSES (ONLY THE CSR IS BYTE ADDRESSABLE)
211
212 001200 160000 WCI 160000 ; WORD COUNT REGISTER
213 001202 160002 BAR1 160002 ; BUS ADDRESS REGISTER
214 001204 160004 CSR1 160004 ; CONTROL-STATUS REGISTER
215 001206 160006 CSR11 160006 ; HIGH BYTE ADDRESS OF CSR
216 001210 160006 DBR1 160006 ; DATA BUFFER REGISTER
217
218          ; XWC
219          ; XBAR
220          ; XCSR
221          ; XCSR1
222          ; XDBR
223
224          001200          .=1200
225 001200 000000 .WORD 0
226
227          001220          .=1220
228
229          ; PDP-11 SOFTWARE REGISTER ADDRESSES
230
231 001220 177566 TPB1 177566 ; TELEPRINTER BUFFER
232 001222 177564 TPS1 177564 ; TELEPRINTER STATUS REGISTER
233 001224 177562 TKB1 177562 ; KEYBOARD BUFFER
234 001226 177560 TKS1 177560 ; KEYBOARD STATUS REGISTER
235
236
237          ; REGISTER DEFINITIONS
238
239          000007          PC=X7
240          000006          SP=X6
241          000005          R5=X5
242          000004          R4=X4
243          000003          R3=X3
244          000002          R2=X2
245          000001          R1=X1 ; WORK1 T7, T12, T13
246          000000          R0=X0 ; WORK1 T12, T13
247
248          ; STANDARD ADDRESS DEFINITIONS
249
250          177776          PSW=177776 ; PROCESSOR STATUS WORD
251          177570          SWR=177570 ; SWITCH REGISTER
252
253          ; *TRAP* TRAP EQUALITIES
254
255          104400          ERROR=TRAP ; ERROR=TRAP 1 TO TRAP 376
256          104400          SCOPE=TRAP ; SCOPE=TRAP 0
257          104777          SUBSCOPE=TRAP+377 ; SUBSCOPE=TRAP 377

```

```

299 001230 000000      ERRORS:          0          ; IF >0 SOME TYPE OF KIT11/D ERROR EXISTS
260 001232 000000      ERRORXI         0          ; CONTAINS THE KIT11/D ERROR #
261                                     ; IRQ = 100000+X, PRINT KIT VECTOR ADDRESS AT DETECTION OF INTERRUPT
262                                     ; IRQ = X-100000, A KIT INTERRUPT IS EXPECTED
263 001234 000000      IRQ:              0          ; = 1 IF A KIT INTERRUPT IS EXPECTED
264 001236 000000      IROOK:          0          ; THE ADRS FOR RETURN FROM AN EXPECTED KIT INTERRUPT
265 001240 000110      LINECOUNT:    110       ; 72 CHARACTER TELEPRINTER LINE
266 001242 000000      PASSLSB:         0          ; PASS COUNTER LSB TO 17777 OCTAL
267 001244 000000      PCSCOPE:         0          ; CONTAINS THE FIRST ADDRESS OF THE " SCOPE " LOOP
268 001246 000000      PCBSCOPE:         0          ; ADDRESS+2 OF THE COMMAND " SUBSCOPE " OR " SCOPE "
269
270                                     ;          WORD COUNT REGISTER = 16XXX0
271
272                                     ; THIS 16 BIT REGISTER IS TO BE LOADED WITH THE 2'S COMPLEMENT OF THE
273                                     ; NUMBER OF WORDS TO BE TRANSFERRED, CAUTION, UNLESS THE KIT IS MODIFIED
274                                     ; IT CAN ADDRESS ONLY UP TO 32K WORDS, THEREFORE, BIT 15 SHOULD ALWAYS
275                                     ; BE LOADED WITH A "ONE" TO KEEP THE TOTAL NUMBER OF TRANSFERS TO 32K
276                                     ; OR LESS. (THIS REGISTER IS NOT BYTE ADDRESSABLE).
277
278                                     ;          BUS ADDRESS REGISTER = 16XXX2
279
280                                     ; THIS 15 BIT REGISTER IS TO BE LOADED WITH THE ADDRESS THAT SPECIFIES
281                                     ; THE MEMORY LOCATION "TO" OR "FROM" WHICH THE FIRST DATA WORD IS TO
282                                     ; BE TRANSFERRED. THIS REGISTER IS INCREMENTED AFTER EACH TRANSFER.
283                                     ; CAUTION, IF THIS REGISTER INCREMENTS TO ITS 32K ADDRESS LIMIT AND THE
284                                     ; WORD COUNT REGISTER HAS NOT INCREMENTED TO ZERO AND THERE HAS BEEN NO
285                                     ; WORD COUNT OVERFLOW, THE ADDRESS WILL "WRAP AROUND" TO LOCATION 0 ON
286                                     ; THE NEXT TRANSFER. (THIS REGISTER IS NOT BYTE ADDRESSABLE).
287
288                                     ;          CONTROL/STATUS REGISTER = 16XXX4
289
290                                     ; THIS 16 BIT REGISTER IS THE ONLY REGISTER WHICH IS BYTE ADDRESSABLE.
291
292                                     ; WRITING INTO THE LOW BYTE WILL CLEAR " NEX ", AND " XOF ".

```

```

294                                     ; THIS ERROR IS FROM AN UNEXPECTED KIT VECTOR ADDRESS INTERRUPT
295                                     ; ENTRY TO HERE MAY BE FROM ANYWHERE WITHIN THIS PROGRAM
296
297 001250 104776
298 001252 104777
299
300                                     ; STARTING ADDRESS 200
301
302 001254 012737 100000 001234 9200; MOV #SW19, IRQ           ; IRQ = 100000
303 001262 005037 001242                                     CLR PASSLSB
304 001264 012706 001200 CONTINUE; MOV #1200, SP       ; PRIME THE STACK POINTER = 1200
305 001272 112737 000340 177776 MOV #340, RSH          ; SET THE STATUS PRIORITY=7
306 001300 005037 001230                                     CLR ERRORS          ; PRIME FOR ANY ERROR COUNT
307 001304 012737 000110 001240 MOV #110, LINECOUNT ; INITIALIZE LINECOUNT FOR 72 CHARACTERS
308 001312 012737 001250 001236 MOV #E376, IRQOK     ; FOR INVALID KIT VECTOR INTERRUPT
309
310 001320 005737 001200                                     TST WC
311 001324 001020                                     BNE XS200
312 001326 004537 005006                                     JSR R5, PRINT
313 001332 005627                                     MHELP1
314 001334 012737 001200 004726 MOV #WC, VA          ; *TOGGLE INTO PROGRAM
315 001342 004537 005100                                     JSR R5, TYPEOCTAL  ; LOCATION 1200 THE
316 001346 004726                                     VA                  ; BB11-D DEVICE ADDRESS
317 001350 004537 005006                                     JSR R5, PRINT      ; IF NOT 160000
318 001354 005670                                     MHELP2
319 001356 000000                                     HALT
320 001360 012737 160000 001200 MOV #160000, WC
321
322 001366 013700 001200 XS200; MOV WC, RB
323 001372 005720                                     TST (RB)+
324 001374 010037 001202 MOV RB, BAR          ; " ADD #2, (RB) "
325                                     ; BAR ADDRESS = (WC) + 2
326
327 001402 010037 001204                                     TST (RB)+
328 001406 105720 MOV RB, CSR          ; CSR ADDRESS = (WC) + 4
329 001410 010037 001206 TSTB (RB)+
330 001414 105720 MOV RB, CSR3         ; CSR (HIGH BYTE) = (WC) + 5
331 001416 010037 001210 TSTB (RB)+
                                     ; DBR ADDRESS = (WC) + 6
                                     MOV RB, DBR

```

```

333          )          TEST 0 = " INIT "
334
335          ) THE STATUS OF THE KIT'S REGISTERS AFTER INITIALIZE SHOULD BE:
336
337          ) WORD COUNT = 0, BUS ADDRESS REGISTER = 0,
338
339          ) DATA BUFFER R = 177777, AND CSR = X2 1 0
340
341          ) (NOTE: " RESET " WILL BE ISSUED WITHIN THE SCOPE LOOP).
342
343          001422 104400          T01          SCOPE          ) TO PRIME FIRST " SCOPE " LOOP EVER
344
345          001424 000005          RESET          ) ISSUED NOW AND IN SCOPE LOOP
346
347          001426 005777 177546          TST 0 WC          ) TEST WORD COUNT = 0
348          001432 001401          BEQ ,+4
349          001434 104401          E11          ERROR+1          ) THE WORD COUNT NOT = 0 AFTER " INIT "
350          001436 104777          SUBSCOPE          ) LOOP
351
352          001440 005777 177536          TST 0 BAR          ) TEST BUS ADDRESS R = 0
353          001444 001401          BEQ ,+4
354          001446 104402          E21          ERROR+2          ) THE BUS ADDRESS R NOT = 0 AFTER " INIT "
355          001450 104777          SUBSCOPE          ) LOOP
356
357          001452 022777 177777 177530          CMP 0=1, 0 DBR          ) TEST THE DATA BUFFER R = 177777
358          001460 001401          BEQ ,+4
359          001462 104403          E31          ERROR+3          ) THE DATA BUFFER R NOT = -1 AFTER " INIT "
360          001464 104777          SUBSCOPE
361
362          001466 004537 005412          JSR R5, XCSR
363          001472 000210          210
364          001474 104404          E41          ERROR+4

```

```
366          ; TEST 2 - CSR BITS: 2 (FNCT), AND 3 (STAT)
367
368          ; THE STATE OF BIT 3 IS THE COMPLEMENT OF BIT 2
369
370 001476 104400 T21 SCOPE
371
372 001500 052777 000004 177476 R1S #4, 0 CSR          ; SETTING BIT 2 SHOULD CLEAR BIT 3
373
374 001506 004537 005412 JSR R5, XCSR
375 001512 000204
376 001514 104410 E10, 204 ERROR+10
377 001516 104400 SCOPE
378
379          ; CLEARING BIT 2 SHOULD SET BIT 3
380
381 001520 005077 177460 CLR 0 CSR
382
383 001524 004537 005412 JSR R5, XCSR
384 001530 000210
385 001532 104412 E12, 210 ERROR+12
```

```

387           ; TEST 3 - CSR BITS, 0 (IN), 7 (RDY)
388
389           ; SETTING BIT 0 (IN) OF THE CSR SHOULD CLEAR BIT 7 (RDY)
390
391           ; NO INTERRUPTS OR CHA'S ARE EXPECTED
392
393 00I534 104400      T31      SCOPE
394
395 00I536 005077 177442      CLR 0 CSR           ; THE CSR IS DEFAULT TO 612
396
397 00I542 012777 177776 177430      MOV 0=2, 0 WC       ; SET THE WORD COUNT R = -2
398 00I550 005077 177426      CLR 0 BAR           ; SET THE BUS ADDRESS R = 2
399 00I554 005277 177424      INC 0 CSR           ; SET BIT 0 (IN) OF THE CSR = 1
400
401 00I560 004737 000406      JSR PC, STALL      ; 10.5 US TIME PER XFER
402
403 00I564 004537 005412      JSR R5, XCSR
404 00I570 000011
405 00I572 104420      E20:      ERROR=20
406 00I574 104777      SUBSCOPE           ; LOOP
407
408 00I576 027727 177376 177776      CMP 0 WC, 0=2     ; THE WORD COUNT SHOULD STILL = -2
409 00I604 001401
410 00I606 104421      E21:      ERROR=21
411 00I610 104777      SUBSCOPE           ; LOOP
412
413 00I612 005777 177364      TST 0 BAR         ; THE BUS ADDRESS R SHOULD STILL = 0
414 00I616 001401
415 00I620 104422      E22:      ERROR=22
416 00I622 104400      SCOPE            ; LOOP
417
418           ; NOW CLEAR BIT 0 (IN) OF THE CSR, THEREBY SETTING BIT 7 (RDY) = 1
419
420           ; BIT 0 (LAST DIRECTION) SHOULD PREVIOUSLY BE = 1
421
422 00I624 042777 000001 177352      BIC #1, 0 CSR     ; CLEARING BIT 0 SHOULD SET BIT 7
423
424 00I632 004537 005412      JSR R5, XCSR
425 00I636 000210
426 00I640 104423      E23:      ERROR=23
427

```

```

429          ;          TEST 9 - DATA AVAILABLE OUT
430
431          ; WRITING INTO THE KITS DATA BUFFER REGISTER SHOULD RAISE " DATA AVAILABLE ".
432
433          ; THE CSR SHOULD = 126410
434
435          ; 15 SPECIAL CONDITION ( 1 HERE BY INPUT DEMAND)
436          ; 13 INPUT DEMAND ( 1 HERE BECAUSE INPUT=1, AND IN=0)
437          ; 11 DATA AVAILABLE OUT (BECAUSE OF THE COMMAND CLR = DBR)
438          ; 10 INPUT ( 1 HERE BY DATA AVAILABLE OUT)
439          ; 8 LAST DIRECTION (=OUTPUT)
440          ; 7 READY ( 0 BECAUSE IN=0, OUT=0, AND DATA AVAILABLE = 1)
441          ; 3 STAT (BECAUSE BIT 2 FNCT=0)
442
443          ; NO INTERRUPTS OR DMA CYCLES ARE EXPECTED TO OCCUR
444
445 001642 104400      T51      SCOPE
446
447 001644 005077 177334      CLR = CSR          ; THE CSR IS DEFAULT TO X2 1 0
448
449 001650 012777 177776 177322      MOV #=2, = WC      ; SET THE WORD COUNT R = 177776
450 001656 005077 177320      CLR = BAR          ; SET THE BUS ADDRESS R = 0
451 001662 005077 177322      CLR = DBR          ; WRITE 0'S INTO THE DATA BUFFER R
452
453 001666 004737 005406      JSR PC, STALL      ; 10.5 US TIME PER TRANSFER
454
455 001672 027727 177300 126410      CMP = CSR, #126410
456 001700 001401          BEQ ,+4
457 001702 104430      E30:      ERROR=30
458 001704 104777          SUBSCOPE          ; LOOP
459
460 001706 027727 177266 177776      CMP = WC, #=2
461 001714 001401          BEQ ,+4
462 001716 104431      E31:      ERROR=31
463 001720 104777          SUBSCOPE          ; THE WORD COUNT R NOT STILL = -2
464                                     ; LOOP
465 001722 005777 177262          TST = DBR
466 001726 001401          BEQ ,+4
467 001730 104434      E34:      ERROR=34
468 001732 104777          SUBSCOPE          ; THE DATA BUFFER REGISTER NOT = 0
469
470 001734 005777 177242          TST = BAR
471 001740 001401          BEQ ,+4
472 001742 104432      E32:      ERROR=32
473 001744 104400          SCOPE          ; THE BUS ADDRESS R NOT STILL = 0
                                     ; LOOP

```

```
475 ; BECAUSE " IN ", AND " OUT " WERE BOTH PREVIOUSLY = 0,  
476 ; CLEARING BIT 11 (DATA AVAILABLE OUT), SETS BIT 7 (READY)  
477  
478 ; THE CSR SHOULD = 122610  
479  
480 ; 15 SPECIAL CONDITION ( 1 BECAUSE INPUT DEMAND=1)  
481 ; 13 INPUT DEMAND ( 1 BECAUSE INPUT=1, AND IN=0)  
482 ; 10 INPUT ( 1 BECAUSE OF DATA AVAILABLE OUT)  
483 ; 8 LAST DIRECTION (=OUTPUT)  
484 ; 7 READY ( 1 BECAUSE IN=0, OUT=0, AND DATA AVAILABLE OUT=0)  
485 ; 3 STAT ( 1 BECAUSE BIT 2 FNCT=0)  
486  
487 001746 142777 000010 177232 BICB #10, @ CSR1 ; CLEAR BIT 11 (DATA AVAILABLE OUT)  
488  
489 001754 027727 177224 122610 CMP @ CSR, #122610  
490 001762 001401 BEQ ,+4  
491 001764 104433 E331 ERROR+33 ; CSR NOT = 122610  
492 001766 104400 SCOPE  
493  
494 ; CLEARING BIT 10 (INPUT) WILL CLEAN BIT 13 (INPUT DEMAND), AND  
495 ; BIT 15 (SPECIAL CONDITION)  
496  
497 001770 142777 000004 177210 BICB #4, @ CSR1 ; CLEAR BIT 10 (INPUT)  
498  
499 001776 027727 177202 000610 CMP @ CSR, #610  
500 002004 001401 BEQ ,+4  
501 002006 104433 E351 ERROR+35 ; CSR NOT = 610  
502 002010 104400 SCOPE
```

```

524                                     )           TEST 7 - DBR
525                                     )
526                                     ) WITH THE DIAGNOSTIC INPUT-OUTPUT CABLE CONNECTED, WRITING INTO THE
527                                     ) DATA BUFFER REGISTER WILL PLACE ON THE DATA LINES OUT OF THE KIT THE
528                                     ) SAME DATA, THEREFORE READING FROM THE DATA BUFFER R SHOULD READ THE
529                                     ) SAME DATA WRITTEN,
530                                     )
531                                     ) THE CONTENTS OF R1 WILL CONTAIN THE DATA TO BE WRITTEN
532                                     )
533                                     ) 1
534                                     ) 2
535                                     ) 4
536                                     ) 10
537                                     ) 20
538                                     ) 40
539                                     ) 100
540                                     ) 200
541                                     ) 400
542                                     ) 1000
543                                     ) 2000
544                                     ) 4000
545                                     ) 10000
546                                     ) 20000
547                                     ) 40000
548                                     ) 100000
549
550                                     ) THE DATA BUFFER HAS PREVIOUSLY BEEN TESTED IN TEST 5 FOR ALL 0'S
551
552 002012 005077 177166 T7I CLR @ CSR ) THE CSR DEFAULTS TO 010
553
554 002016 012701 000001      MOV #1, R1 ) R1 IS A WORK REGISTER
555
556 002022 104400 T7L: SCOPE ) SETS (PCSCOPE) = ,+2
557
558 002024 010177 177160      MOV R1, @ CSR )WRITE INTO THE KITS DATA BUFFER R
559
560 002030 004737 005406      JSR PC, STALL
561
562 002034 020177 177150      CMP R1, @ CSR
563 002040 001401      BEQ ,+4
564 002042 104450 E50: ERROR+50 ) THE DATA READ NOT = TO THE DATA WRITTEN
565 002044 104777 SUBSCOPE ) SCOPE LOOP IF (ERRORS) > 0
566
567 002046 006301      ASL R1 ) SHIFT THE CONTENTS OF R1 FOR NEXT BIT
568 002050 001364      BNE T7L ) UNTIL (R1) = 0

```

```

550 ; TEST 8 (OUTPUT PART 1) ; 1=NPR (OUT) W/O WORD COUNT CVERFLOW
551
552 ; NO INTERRUPTS ARE EXPECTED
553
554 002052 104400 T0, SCOPE
555
556 002054 009077 177124 CLR 0 CSR ; CSR DEFAULT TO X2 1 2
557
558 002060 012777 177776 177112 MOV #2, 0 WC ; SET THE WC R = 177776
559 002066 012777 009752 177106 MOV #BUFFER, 0 BAR ; SET THE BJS ADDRESS R = ADDRESS1 BUFFER
560 002074 032777 000002 177102 RIS #2, 0 CSR ; SET BIT 1 (OUT)
561
562 002102 004737 009406 JSR PC, STALL ; 10.5 US TIME PER TRANSFER
563 002106 004737 009406 JSR PC, STALL
564
565
566 ; CSR SHOULD = 126412
567
568 ; 15 SPECIAL CONDITION ( 1 HERE FROM INPUT DEMAND)
569 ; 13 INPUT DEMAND ( 1 BECAUSE INPUT=1, AND IN 00)
570 ; 11 DATA AVAILABLE OUT ( 1 BECAUSE OUT=1)
571 ; 10 INPUT (BECAUSE DA OUT SET DA IN)
572 ; 8 LAST DIRECTION (=OUTPUT)
573 ; 3 STAT ( 1 BECAUSE FNCT=0)
574 ; 1 OUT
575
576 ; WC SHOULD = -1
577 ; BAR SHOULD = #BUFFER+2
578
579 002112 027727 177066 126412 CMP 0 CSR, 0126412
580 002120 001401 BEQ ,+4
581 002122 104457 E57I ERROR+57 ; CSR NOT = 126412
582 002124 104777 SUBSCOPE
583
584 002126 022777 177777 177044 CMP #1, 0 WC
585 002134 001401 BEQ ,+4
586 002136 104460 E60I ERROR+60 ; WC NOT = 177777
587 002140 104777 SUBSCOPE
588
589 002142 022777 009754 177032 CMP #BUFFER+2, 0 BAR
590 002150 001401 BEQ ,+4
591 002152 104461 E61I ERROR+61 ; BAR NOT = #BUFFER+2

```

```

593 ; TEST 9 (OUTPUT PART 2) ; 1-NPR (OUT) WITH WORD COUNT OVFL0
594
595 ; CSR SHOULD = 127410
596
597 ; 15 SPECIAL CONDITION ( 1 BECAUSE OF INPUT DEMAND)
598 ; 13 INPUT DEMAND ( 1 BECAUSE INPUT=1, AND IN=0)
599 ; 11 DATA AVAILABLE OUT
600 ; 10 INPUT ( 1 BECAUSE OF DATA AVAILABLE OUT)
601 ; 9 LAST XFER ( 1 BECAUSE OF WORD COUNT OVERFLOW)
602 ; 8 LAST DIRECTION ( =OUTPUT)
603 ; 7 READY ( 1 BECAUSE OUT=0 FROM WORD COUNT OVERFLOW)
604 ; 3 STAT ( 1 BECAUSE BIT 2 FNCT = 0)
605
606 ; WC SHOULD = 0
607 ; BAR SHOULD = #BUFFER+2
608
609 002154 104400 T9; SCOPE
610
611 002156 005077 177022 CLR # CSR ; CSR DEFAULTS TO X2 1 0
612
613 002162 012777 177777 177010 MOV #=1, # WC ; SET WC = 177777
614 002170 012777 005752 177004 MOV #BUFFER, # BAR ; SET BAR = #BUFFER
615 002176 052777 000002 177000 RIS #2, # CSR ; SET BIT 2 (OUT) = 1
616
617 002204 004737 009406 JSR PC, STALL ; 10.5 US TIME PER TRANSFER
618
619 002210 027727 176770 127410 CMP # CSR, #127410
620 002216 001401 BEQ ,+4
621 002220 104462 E621 ERROR+62 ; CSR NOT = 127410
622 002222 104777 SUBSCOPE
623
624 002224 005777 176750 TST # WC
625 002230 001401 BEQ ,+4
626 002232 104463 E631 ERROR+63 ; WC NOT = 0
627 002234 104777 SUBSCOPE
628
629 002236 022777 005754 176736 CMP #BUFFER+2, # BAR
630 002244 001401 BEQ ,+4
631 002246 104464 E641 ERROR+64 ; BAR NOT = #BUFFER+2
632 002250 104400 SCOPE

```

```
634 ; CLEARING BIT 11 (DATA AVAILABLE OUT) CLEARS BIT 9 (LAST TRANSFER), AND
635 ; BECAUSE " IN " AND " OUT " WERE BOTH PREVIOUSLY = 0, SETS BIT 7 (READY)
636
637 ; THE CSR SHOULD = 122610
638
639 ; 15 SPECIAL CONDITION ( 1 BECAUSE INPUT DEMAND)
640 ; 13 INPUT DEMAND ( 1 BECAUSE INPUT=1, AND IN=0)
641 ; 10 INPUT ( 1 BECAUSE OF DATA AVAILABLE OUT)
642 ; 8 LAST DIRECTION ( =OUTPUT)
643 ; 7 READY ( 1 BECAUSE IN=0, OUT=0, AND DATA AVAILABLE OUT=0)
644 ; 3 STAT ( 1 BECAUSE BIT 2 FNCT = 0)
645
646 002252 142777 000010 176726 R1CB #10, 0 CSR1 ; CLEAR BIT 11 (DATA AVAILABLE OUT)
647
648 002260 027727 176720 122610 CMP 0 CSR, #122610
649 002266 001401 R00 ,+4
650 002270 104405 E051 ERROR+05 ; CSR NOT = 122610
```

```

652                                     ; TEST 10 (INPUT)
653                                     ;
654                                     ; SET BIT 0 (IN) OF THE CSR THEN RAISE DATA AVAILABLE OUT BY WRITING INTO THE DBR
655
656 002272 104400 T10: SCOPE
657
658 002274 005077 176704 CLR 0 CSR ; CSR = X2 1 2
659
660 002300 012777 005752 176674 MOV #BUFFER, 0 BAR ; BUS ADDRESS R = #BUFFER
661 002306 012777 177776 176664 MOV #2, 0 MC ; WORD COUNT R = 177776
662 002314 005277 176664 INC 0 CSR ; SET BIT 0 (IN)
663 002320 005077 176664 CLR 0 DBR ; RAISE DATA AVAILABLE OUT
664
665 002324 004737 005406 JSR PC, STALL ; 10.5 US TIME PER TRANSFER
666
667 ; THE CSR SHOULD = 11
668
669 002330 004537 005412 JSR R5, XCSR
670 002334 000011
671 002336 104470 E70: ERROR=70
672 002340 104777 SUBSCOPE
673
674 ; THE BUS ADDRESS SHOULD = #BUFFER+2
675
676 002342 022777 005754 176632 CMP #BUFFER+2, 0 BAR
677 002350 001401 BEQ ,+4
678 002352 104472 E72: ERROR=72
679 002354 104777 SUBSCOPE
680
681 ; THE WORD COUNT REGISTER SHOULD = -1
682
683 002356 022777 177777 176614 CMP #-1, 0 MC
684 002364 001401 BEQ ,+4
685 002366 104474 E74: ERROR=74

```

```
687          I      TEST 11 (OUTPUT AND INPUT SIMULTANEOUSLY)
688
689          I SET BITS 0 (IN), AND 1 (OUT) OF THE CSR SIMULTANEOUSLY
690
691 002370 104400          T11:  SCOPE
692
693 002372 005077 176606          CLR 0 CSR          I CSR = X2 1 0
694
695 002376 012777 177776 176574          MOV 0-2, 0 WC          I SET WORD COUNT R = 177776
696 002404 012777 009752 176570          MOV #BUFFER, 0 BAR          I SET BUS ADDRESS R = #BUFFER
697 002412 052777 000003 176564          BIS 03, 0 CSR          I SET OUT, AND IN
698
699 002420 004737 009406          JSR PC, STALL          I 10.5 US TIME PER TRANSFER
700 002424 004737 009406          JSR PC, STALL          I 10.5 US TIME PER TRANSFER
701
702          I THE CSR SHOULD = 210
703
704 002430 004537 009412          JSR R5, XCSR
705 002434 000210          210
706 002436 104500          E100: ERROR=100
707 002440 104777          SUBSCOPE
708
709          I THE BAR SHOULD = #BUFFER+4
710
711 002442 022777 009756 176532          CMP #BUFFER+4, 0 BAR
712 002450 001401          BEQ ,+4
713 002452 104502          E102: ERROR=102
714 002454 104400          SCOPE
715
716          X11:,
002456
```

.NLIST SEQ,LOC

TEST 12 - DATA TEST (PART 1)

WRITE THE DATA PATTERN AS INDICATED BELOW

	BUFFER	WORD#		PATTERN
005752	BUFFER	11	INPUT	0
005756	BUFFER+4	12	INPUT	1
005762	BUFFER+10	13	INPUT	2
005766	BUFFER+14	14	INPUT	4
005772	BUFFER+20	15	INPUT	10
005776	BUFFER+24	16	INPUT	20
006002	BUFFER+30	17	INPUT	40
006006	BUFFER+34	18	INPUT	100
006012	BUFFER+40	19	INPUT	200
006016	BUFFER+44	110	INPUT	400
006022	BUFFER+50	111	INPUT	1000
006026	BUFFER+54	112	INPUT	2000
006032	BUFFER+60	113	INPUT	4000
006036	BUFFER+64	114	INPUT	10000
006042	BUFFER+70	115	INPUT	20000
006046	BUFFER+74	116	INPUT	40000
006052	BUFFER+100	117	INPUT	100000

760
761

002456

.LIST SEQ,LOC
.EX11

```

763 002456 012701 005752      T12I  MOV #BUFFER, R1
764 002462 005000              CLR R0
765 002464 000201              SEC
                                ; FIRST WORD = 0 (AND C BIT)
                                ; SET CARRY BIT IN PROCESSOR STATUS WORD
766
767 002466 010021              T12LI MOV R0, (R1)*
768 002470 012721 000000      MOV #0, (R1)*
                                ; BUFFER, BUFFER+4, BUFFER+6, ETC
                                ; MOV INST SO NOT TO CLEAR C BIT WITHIN PSW
769 002474 006100              ROL R0
770 002476 001373              BNE T12L
771
772 002500 104400              SCOPE
                                ; SETS (PCSCOPE) = ,+2
773
774 002502 005077 176476      CLR @ CSR
                                ; CSR = X2 1 0
775
776 002506 012777 177736 176464  MOV #-42, @ MC
777 002514 012777 005752 176460  MOV #BUFFER, @ BAR
778 002522 052777 000003 176454  BIS #3, @ CSR
                                ; 17 WORDS TO TRANSFER X 2
                                ; SET IN, AND OUT
779
780 002530 012701 177736      MOV #-42, R1
                                ; STALL FOR 17 WORDS (OUT-IN)
781
782 002534 005777 176440      13I  TST @ MC
783 002540 001402              BEQ 25
784 002542 005201              INC R1
785 002544 001373              BNE 13
786
787                                ; ALL 34 WORDS SHOULD HAVE BEEN TRANSFERRED OUT THEN IN BY NOW
788
789                                ; THE CSR SHOULD = 210
790
791 002546 004537 005412      25I  JSR R5, XCSR
792 002550 000210              210
793 002554 104510              E110: ERROR+110
794 002556 104777              SUBSCOPE
795
796                                ; COMPARE THE DATA
797
798                                ; THE CONTENTS OF BUFFER SHOULD = THE CONTENTS OF BUFFER+2
799
800                                ; ...ETC...
801
802 002560 012701 005752      XT12LI MOV #BUFFER, R1
803 002564 004737 005432      JSR PC, COMPARE
804 002570 001004              BNE E112
805 002572 022701 006056      CMP #BUFFER+104, R1
806 002576 001372              BNE XT12L
807 002600 000401              BR OK112
808
809 002602 104512              E112: ERROR+112
810 002604 104400              OK112: SCOPE
811
812                                X12=,

```

.NLIST SEQ,LOC

1 TEST 13 - DATA TEST (PART 2)

1 WRITE THE DATA PATTERN AS INDICATED BELOW

	1	BUFFER	WORDS	PATTERN
005752		BUFFER	11	177776
005756		BUFFER+4	13	1
005752		BUFFER10	15	177775
005766		BUFFER+14	17	2
005772		BUFFER+20	111	177773
005776		BUFFER+24	113	4
006002		BUFFER+30	115	177767
006006		BUFFER+34	117	10
006012		BUFFER+40	121	177757
006016		BUFFER+44	123	20
006022		BUFFER+50	125	177737
006026		BUFFER+54	127	40
006032		BUFFER+60	131	177677
006036		BUFFER+64	133	100
006042		BUFFER+70	135	177577
006046		BUFFER+74	137	200
006052		BUFFER+100	141	177377
006056		BUFFER+104	143	400
006062		BUFFER+110	145	176777
006066		BUFFER+114	147	1000
006072		BUFFER+120	151	175777
006076		BUFFER+124	153	2000
006102		BUFFER+130	155	173777
006106		BUFFER+134	157	4000
006112		BUFFER+140	161	167777
006116		BUFFER+144	163	10000
006122		BUFFER+150	165	157777
006126		BUFFER+154	167	20000
006132		BUFFER+160	171	130000
006136		BUFFER+164	173	40000
006142		BUFFER+170	175	077777
006146		BUFFER+174	177	100000

062
063

002606

.LIST SEQ,LOC
.X12

```

065 002606 012701 005752      T13;  MOV #BUFFER, R1
066 002612 012700 000001      MOV #1, R0
067 002616 005100      1S;   COM R0
068 002620 010021      MOV R0, (R1)+
069 002622 005021      CLR -(R1)+
070 002624 005100      COM R0
071 002626 010021      MOV R0, (R1)+
072 002630 005021      CLR (R1)+
073 002632 006300      ASL R0
074 002634 001370      BNE 1S
075
076 002636 104400      SCOPE
077
078 002640 005077 176340      CLR # CSR
079
080 002644 012777 177700 176326      MOV #=100, # WC      ; 64 WORDS (32 OUT/32 IN)
081 002652 012777 005752 176322      MOV #BUFFER, # BAR
082 002660 052777 000003 176316      RIS #3, # CSR
083
084 002666 012701 177000      MOV #=1000, R1
085
086 002672 005777 176302      2S;   TST # WC
087 002676 001402      BEO 3S
088 002700 005201      INC R1
089 002702 001373      BNE 2S
090
091      ; ALL 32 X 2 WORDS SHOULD HAVE BEEN XFERRED BY NOW
092
093      ; THE CSR SHOULD = 210
094
095 002704 004537 005412      3S;   JSR R0, XCSR
096 002710 000210      210
097 002712 104516      E116; ERROR+116
098 002714 104777      SUBSCOPE
099
100      ; COMPARE DATA
101
102 002716 012701 005752      XT13L; MOV #BUFFER, R1
103 002722 004737 005432      JSR PC, COMPARE
104 002726 001004      BNE E120
105 002730 022701 006152      CMP #BUFFER+200, R1
106 002734 001372      BNE XT13L
107 002736 000401      BR T14
108
109 002740 104520      E120; ERROR+120

```

```
911          |          TEST 14 = INTR ENABLE
912          |
913          |          (NO INTERRUPTS SHOULD OCCUR)
914          |
915 002742 104400      |T14| SCOPE
916          |
917 002744 012737 002774 001236      | MOV #E131, IRQOK      | PC RETURN IF UNEXPECTED IRQ
918          |
919 002752 142737 000340 177776      | BICB #340, PSW      | SET PSW PRIORITY = 0
920 002760 012777 000100 176216      | MOV #100, 0 CSR      | SET BIT 6 (ENB)
921          |
922 002766 004737 005406          | JSR PC, STALL
923          |
924 002772 000401          | BR E131+2
925          |
926 002774 104531      |E131| ERROR+131
927 002776 104777      | SUBSCOPE
928          |
929          | THE CSR SHOULD = X3 1 0
930          |
931 003000 004537 005412          | JSR R5, XCSR
932 003004 000310          | 310
933 003006 104530      |E130| ERROR+130
934 003010 104777      | SUBSCOPE
935          |
936          | CLEAR BIT 6 (ENB) OF THE CSR
937          |
938 003012 042777 000100 176164      | BIC #100, 0 CSR
939          |
940 003020 004537 005412          | JSR R5, XCSR
941 003024 000210          | 210
942 003026 104532      |E132| ERROR+132
```

```
944          ;      TEST 15 - INTR ENABLE PART 1 (FROM WORD COUNT CVERFLOW)
945
946          ; TEST THE KITS (BB11=0) VECTOR ADDRESS
947
948          ; THE VECTOR ADDRESS UP TO THIS TEST IS UNKNOWN
949
950          ; SET IRQ=1 TO EXPECT A KIT INTERRUPT
951
952          ;      (FROM ANY PRIO LEVEL FOR NOW)
953
954          ;      THEN ANY " IOT " TRAP IS VALAD
955
956 003030 104400      T15:  SCOPE
957
958 003032 005077 176146      CLR 0 CSR
959
960 003036 012737 003114 001236      MOV #E140+2, IRQOK      ; SETUP FOR PC RETURN IF INTERRUPT OK
961 003044 142737 000340 177776      BICB #340, PSH      ; PRIORITY LEVEL = 0
962 003052 052737 000001 001234      BIS #1, IRC      ; PRIME FOR EXPECTED KIT INTERRUPT
963 003060 012777 177776 176112      MOV 0=2, 0 WC
964 003066 012777 005752 176106      MOV #BUFFER, 0 BAR
965 003074 012777 000103 176102      MOV #103, 0 CSR      ; SET BIT 6 (ENB)
966
967 003102 004737 005406      JSR PC, STALL
968 003106 004737 005406      JSR PC, STALL
969
970
971          ; AN INTERRUPT SHOULD HAVE OCCURED BY NOW
972
973 003112 104540      E140:  ERROR+140
974 003114 104777      SUBSCOPE
975
976          ; THE CSR SHOULD = 310
977
978 003116 004537 005412      JSR R5, XCSR
979 003122 000310      310
980 003124 104542      E142:  ERROR+142
```

```
982          ;TEST 16 - INTERRUPT ENABLE PART 2 (FROM SPECIAL CONDITION)
983
984 003126 104400          T16;   SCOPE
985
986 003130 012737 003202 001236          MOV #E143+2, IRQOK
987 003136 142737 000340 177776          R1CB #340, PSM
988 003144 052737 000001 001234          BIS #1, IRC
989 003152 005077 176022          CLR # WC
990 003156 005077 176020          CLR # BAR
991 003162 012777 000100 176014          MOV #100, # CSR
992 003170 005077 176014          CLR # DBR
993
994 003174 004737 005406          JSR PC, STALL
995
996          ; AN INTERRUPT SHOULD HAVE OCCURED BY NOW
997
998 003200 104543          E143;  ERROR+143
999 003202 104777          SUBSCOPE
1000
1001          ; THE CSR SHOULD = 126110
1002
1003          ; 15 SPECIAL CONDITION ( 1 BECAUSE OF INPUT DEMAND)
1004          ; 13 INPUT DEMAND ( 1 BECAUSE INPUT = 1, AND IN = 0)
1005          ; 11 DATA AVAILABLE OUT
1006          ; 10 INPUT ( 1 BECAUSE DATA AVAILABLE OUT)
1007          ; 8 LAST DIRECTION = X
1008          ; 7 READY ( 0 BECAUSE IN=0, OUT=0, AND DATA AVAILABLE OUT=1)
1009          ; 3 STAT ( 1 BECAUSE FNCT = 0)
1010
1011 003204 004537 005412          JSR R9, XCSR
1012 003210 126110          126110
1013 003212 104544          E144;  ERROR+144
```

```

1015      |      TEST 17 - PRIORITY LEVEL INTERRUPTS 5 TO 7 EXPECTED
1016
1017      |      (THE KIT SHOULD INTERRUPT IN PL-5 )
1018
1019      003214  104400      T17:  SCOPE
1020
1021      003216  005077  175762      CLR @ CSR
1022
1023      003222  012737  003300  001236      MOV #E146+2, IRQOK      ; FOR PC RETURN IF INTERRUPT OK
1024      003230  112737  000200  177776      MOV @200, PSM          ; SET PRIORITY LEVEL = 4
1025      003236  052737  000001  001234      BIS #1, IRC
1026      003244  012777  177776  175726      MOV #2, @ MC
1027      003252  012777  009752  175722      MOV @BUFFER, @ BAR
1028      003260  012777  000103  175716      MOV #103, @ CSR        ; SET BIT 6 (ENB)
1029
1030      003266  004737  005406      JSR PC, STALL
1031      003272  004737  005406      JSR PC, STALL
1032
1033      | AN INTERRUPT SHOULD HAVE OCCURED BY NOW
1034
1035      003276  104546      E146:  ERROR+146
1036
1037      |      TEST 18 - PRIORITY LEVEL 9 ONLY
1038
1039      | SETTING THE PSM TO @ PRIORITY # 9, NO INTERRUPT SHOULD OCCUR
1040
1041      003300  104400      T18:  SCOPE
1042
1043      003302  005077  175676      CLR @ CSR
1044
1045      003306  012737  003304  001236      MOV #E152, IRQOK      ; FOR PC RETURN IF UNEXPECTED IRQ
1046      003314  112737  000240  177776      MOV @240, PSM          ; SET PRIORITY LEVEL = 9
1047      003322  052737  000001  001234      BIS #1, IRC
1048      003330  012777  177776  175642      MOV #2, @ MC
1049      003336  012777  009752  175636      MOV @BUFFER, @ BAR
1050      003344  012777  000103  175632      MOV #103, @ CSR        ; SET BIT 6 (ENB)
1051
1052      003352  004737  005406      JSR PC, STALL
1053      003356  004737  005406      JSR PC, STALL
1054
1055      003362  000401      BR E152+2              ; OK, NO INTERRUPT OCCURED
1056
1057      | NO INTERRUPTS SHOULD HAVE OCCURED
1058
1059      003364  104552      E152:  ERROR+152

```

```

1061                                     ; TEST 19 - LAST TRANSFER/XOF
1062
1063 003366 104400                       T19: SCOPE
1064
1065 003370 005077 175610                 CLR @ CSR
1066
1067 003374 012777 177777 175576         MOV #=1, @ MC
1068 003402 012777 005752 175572         MOV #BUFFER, @ BAR
1069 003410 052777 000002 175566         BIS #2, @ CSR
1070
1071 003416 004737 005406                 JSR PC, STALL
1072
1073 003422 027727 175556 127410         CMP @ CSR, #127410
1074 003430 001401                         BEQ ,+4
1075 003432 104560                       E160: ERROR+160
1076 003434 104777                         SUBSCOPE
1077
1078                                     ; NOW WITHOUT CLEARING LAST TRANSFER, SET " IN "
1079                                     ; THEREBY RAISING EXTERNAL OVERFLOW (XOF)
1080
1081
1082 003436 012777 177776 175534         MOV #=2, @ MC
1083 003444 005277 175534                 INC @ CSR
1084
1085 003450 004737 005406                 JSR PC, STALL
1086
1087 003454 004537 005412                 JSR R5, XCSR
1088 003460 110210                         110210
1089 003462 104562                       E162: ERROR+162
1090 003464 104400                         SCOPE
1091
1092                                     ; TO CLEAR " XOF " WRITE INTO THE LOW BYTE OF THE CSR
1093
1094 003466 052777 000010 175510         BIS #10, @ CSR ; TRY TO WRITE BIT 3 (STAT)
1095
1096 003474 004537 005412                 JSR R5, XCSR
1097 003500 000210                         210
1098 003502 104564                       E164: ERROR+164
1099
1100                                     ; IF DATA SWITCH 13 = 1 DO NOT HALT AT THE END OF THIS PASS
1101
1102 003504 104400                         SCOPE
1103
1104 003506 032737 020000 177570         BIT #SW13, SWR
1105 003514 001401                         BEQ ,+4
1106 003516 000000                         HALT
1107 003520 005237 001242                 INC PASSLSB
1108 003524 000137 001266                 JMP CONTINUE

```

```

1110          ; STARTING ADDRESS 202
1111
1112          ; CONVERSATION BETWEEN 2 OR MORE BB11-D'S
1113
1114          005752      OUTBUFFER=BUFFER          ; O-UTPUT BUFFER
1115          006092      INBUFFER=OUTBUFFER+100      ; I-NPUT BUFFER
1116          003530      177740      KWC:          -40      ; OUTPUT WC (CONSTANT AND RESTORE)
1117          003532      005752      KBAR:         OUTBUFFER ; OUTPUT BAR (CONSTANT AND RESTORE)
1118
1119          003534      112737      000340      177776      S202:   MOVB #340, PSH
1120
1121          003542      012701      005752          MOV  #BUFFER, R1
1122          003546      012700      000001          MOV  #1, R0
1123          003552      005100          15:         COM  R0
1124          003554      010021          MOV  R0, (R1)+
1125          003556      005100          COM  R0
1126          003560      010021          MOV  R0, (R1)+
1127          003562      006300          ASL  R0
1128          003564      001372          BNE  15
1129
1130          003566      012737      100000      001234      MOV  #515, IRC          ; IRC = 100000
1131          003574      012737      000110      001240      MOV  #110, LINECOUNT ; INITIALIZE LINECOUNT FOR 72 CHARACTERS
1132          003602      005037      001242          CLR  PASSLSB
1133          003606      005037      001230          CLR  ERRORS          ; PRIME FOR ANY ERROR COUNT
1134          003612      012737      177737      004144      MOV  #-41, INWC       ; "INPUT" WORD COUNT -41
1135          003620      012737      006092      004146      MOV  #INBUFFER, INBAR ; "INPUT" BUSS ADDRESS REGISTER BUFFER
1136
1137          003626      012737      177740      003530      XS202:  MOV  #-40, KWC
1138          003634      012737      005752      003532      MOV  #OUTBUFFER, KBAR
1139
1140          ; IF THE STATUS BIT = 0, THE OTHER KIT IS NOT READY, WAIT
1141
1142          003642      104400      OUT:    SCOPE
1143
1144          003644      012706      001200          MOV  #1200, SP
1145          003650      112777      000004      175326      MOVB #4, @ CSR
1146
1147          003656      004737      004300          JSR  PC, WAIT202
1148
1149          003662      013777      003530      175310      MOV  KWC, @ MC
1150          003670      001756          BEO  XS202          ; (KWC) = 0
1151          003672      013777      003532      175302      MOV  KBAR, @ BAR
1152          003700      012737      003736      001236      MOV  #DIRECTION, IR00K
1153          003706      052737      000001      001234      BIS  #1, IRC
1154          003714      052777      000102      175262      BIS  #102, @ CSR    ; "OUT"
1155          003722      112737      000200      177776      MOVB #200, PSH      ; PRIORITY 4
1156
1157          003730      004737      004300          25:         JSR  PC, WAIT202
1158          003734      000775          BR  25

```

```
1160 ; A PRIORITY LEVEL # 9 INTERRUPT OCCURED FROM THE BB11-D
1161
1162 ; IF " INPUT DEMAND " = 1, THEN STOP TRANSMISSIONS AND INPLY DATA
1163
1164 ; IF " NEX " THEN ERROR 202
1165
1166 003736 032777 020000 175240 ODIRECTION, BIT #SW13, @ CSR
1167 003744 001410 REG 15 ; INPUT DEMAND = 0
1168
1169 ; IF SW 11 = 1, IGNORE ALL INPUT DEMANDS UNTIL WORD COUNT CVENFLOW (READY = 1)
1170
1171 003746 032737 004000 177570 BIT #SW11, SWR
1172 003754 001412 REG IN ; SW11 = 0
1173 003756 105777 175222 TST @ CSR ; WAIT FOR BIT 07 (READY)
1174 003762 100375 BPL ,=4
1175 003764 000406 BR IN
1176
1177 003766 005777 175212 15; TST @ CSR
1178 003772 100315 BPL XS202 ; MUST BE WORD COUNT OVERFLOW
1179
1180 ; . . . ERROR . . .
1181
1182 ; NOT " INPUT DEMAND " OR WORD COUNT OVERFLOW
1183
1184 ; THE BB11-D MUST HAVE TRIED TO ADDRESS NON EXTENDED MEMORY
1185
1186 003774 104602 E202; ERROR+202 ; " SPECIAL CONDITION " = 1,
1187 003776 104777 SUBSCOPE ; MUST BE " NEX " BECAUSE DIRECTION WAS = OUT
1188 ; THEREFORE " XOF " COULD NOT CAUSE SC
1189 004000 000712 BR XS202
```

```

1191          ; INPUT DEMAND = 1
1192
1193          ; SAVE THE OUT WORD COUNT, AND THE OUT BUS ADDRESS REGISTER
1194
1195          ; ENABLE " IN "
1196
1197 004002 042777 000102 175174 INI      BIC #102, 0 CSR      ; DISABLE " OUT"
1198
1199 004010 017737 175164 003530      MOV 0 WC, XWC
1200 004016 017737 175160 003532      MOV 0 BAR, XBAR
1201 004024 013777 004144 175146      MOV INWC, 0 WC      ; WC = -41 (MAX), XOF SHOULD OCCUR AT -40
1202 004032 013777 004146 175142      MOV INBAR, 0 BAR
1203 004040 012737 004152 001236      MOV #DIRECTION, IRQOK
1204 004046 092737 000001 001234      RIS #1, IRC
1205 004054 112777 000101 175122      MOVB #101, 0 CSR
1206 004062 112737 000200 177776      MOVB #200, PSW      ; PRIORITY 4
1207
1208          ; IF AFTER 10 US THE CONTENTS OF THE WORD COUNT REGISTER IS NOT EQUIV-
1209          ; ALENT TO THE CONTENTS+2 OF THE WORD COUNT REGISTER PRIOR TO THE
1210          ; STALL, THE DATA HAS STOPPED COMING IN
1211
1212 004070 017737 175104 004150 151      MOV 0 WC, XWC
1213 004076 004737 005406                JSR PC, STALL      ; 10 US
1214 004102 004737 005406                JSR PC, STALL      ; 10 US
1215 004106 023777 004150 175064      CMP XWC, 0 WC
1216 004114 001365                BNE 15              ; DATA STILL COMING IN
1217
1218          ; DATA STOPPED COMING IN, SAVE "INWC", AND "INBAR"; THEN TRANSMIT
1219
1220 004116 042777 000101 175060      BIC #101, 0 CSR
1221 004124 017737 175050 004144      MOV 0 WC, INWC
1222 004132 017737 175044 004146      MOV 0 BAR, INBAR
1223
1224 004140 000137 003642                JMP OUT
1225
1226 004144 177737          INWC:  -41
1227 004146 006052          INBAR: INBUFFER
1228 004150 000000          XWC:   0
  
```

```

1230      ; A PRIORITY LEVEL 5 INTERRUPT OCCURED FROM THE BB11-D
1231
1232      ; IF = XOF = = 1, COMPARE DATA
1233
1234      ; IF NOT XOF, THEN MUST BE " NEX ", ERROR 206
1235
1236      ; ALL DATA MUST BE "IN", RESET "INHC", AND "INBAR"
1237
1238 004198 012737 177737 004144 IDIRECTI ONI MOV #41, INHC
1239 004168 012737 006052 004146      MOV #INBUFFER, INBAR
1240 004166 032777 010000 175010      BIT #SW12, 0 CSR
1241 004174 001004      BNE ICOMPARE      ; XOF = 1, COMPARE DATA
1242
1243 004176 104606      E2061  ERROR+206      ; INPUT ADDRESSED NON EXTENDED MEMORY
1244 004200 104777      SUBSCOPE      ; XOF=0, THEREFORE ILLEGAL WCO=1
1245
1246 004202 000137 003642      JMP OUT
1247
1248 004206 012701 009750      ICOMPAREI MOV #OUTBUFFER-2, R1
1249 004212 009721      ICLOOPI  TST (R1)      ; ADD #2, R1
1250 004214 010137 009460      MOV R1, @ADDRESS
1251 004220 011137 009462      MOV (R1), GOOD
1252 004224 016137 000100 005456      MOV 100(R1), BAD
1253 004232 026111 000100      CMP 100(R1), (R1)
1254 004236 001004      BNE E210
1255 004240 022701 006050      CMP #BUFFER+70, R1
1256 004244 001362      BNE ICLOOP
1257 004246 000401      BR OK210
1258
1259 004250 104610      E210:  ERROR+210      ; DATA ERROR (GOOD=BAD)
1260 004252 104777      OK210:  SUBSCOPE
1261
1262      ; DATA SWITCH 13 = 1 TO HALT AT END OF RECEIVE DATA COMPARE
1263
1264 004254 032737 020000 177970      BIT #SW13, SWR
1265 004262 001404      REQ 15
1266 004264 042777 000004 174712      BIC #4, 0 CSR
1267 004272 000000      HALT
1268 004274 000137 003642      15:  JMP OUT
1269
1270      ; WAIT HERE FOR THE KIT TO BECOME READY, OR FOR ANY INTERRUPT
1271
1272 004300 004737 009400      WAIT202: JSR PC, STALL      ; 10 US
1273 004304 032777 000010 174672      BIT #10, 0 CSR
1274 004312 001412      REQ 15
1275 004314 005201      INC R1
1276 004316 001370      BNE WAIT202
1277 004320 032737 010000 177970      BIT #SW12, SWR
1278 004326 001364      BNE WAIT202
1279 004330 004537 009000      JSR R9, PRINT
1280 004334 009577      MWAITING
1281 004336 000760      BR WAIT202
1282 004340 000207      15:  RTS PC

```

```

1284          ; TRAP (ERROR) CATCHES THE CODES 104400 TO 104777
1285
1286 004342 011637 001232  ERRORCATCHER: MOV @ Sp, ERRORX ; (ERRORX) = X ADDRESS + 2
1287 004346 102737 000002 001232  SUB #2, ERRORX ; (ERRORX) = X ADDRESS
1288 004394 017737 174652 001232  MOV @ ERRORX, ERRORX ; (ERRORX) = CONTENTS OF THE X ADDRESS
1289 004362 042737 104400 001232  BIC #ERROR, ERRORX ; SAVING BITS 7 THRU 0
1290
1291          ; IF (ERRORX) = 0 = SCOPE
1292          ; IF (ERRORX) = 377 = SUBSCOPE
1293          ; IF (ERRORX) = 1 TO 376 = SOME KIND OF KIT110 ERROR
1294
1295 004370 001927          BEQ XSCOPE ; " SCOPE "
1296 004372 022737 000377 001232  CMP #377, ERRORX
1297 004400 001933          BEQ XSSCOPE ; " SUBSCOPE "
1298
1299          ; (ERRORX) = 1 TO 376 = A KIT110 ERROR
1300
1301          ; SET " PCSSCOPE " JUST IN CASE THE PC (NOW RESIDING WITHIN THE Sp)
1302          ; IS EQUIVALENT TO THE ADDRESS OF X ERROR+2
1303
1304 004402 011637 001246          MOV @ Sp, PCSSCOPE
1305 004406 062737 000002 001246  ADD #2, PCSSCOPE ; NOW LOOKS LIKE X SUBSCOPE+2, OR X SCOPE+2
1306 004414 005237 001230  INCERRORS: INC ERRCRS ; +1 TO THE ERROR TABULATOR
1307 004420 001775          BEQ INCERRCRS ; SO ERRORS DO NOT = 0
1308
1309          ; IF ERRORS ARE ANY CODE 200, WRITE THE " F4CT " BIT = 0
1310
1311          ; (I.E. MAKING BOTHISEO KIT NOT READY FOR S202)
1312
1313 004422 032737 000100 001232  BIT #100, ERRORX
1314 004430 001007          ONE 13
1315 004432 032737 000200 001232  BIT #200, ERRORX
1316 004440 001403          BEQ 13
1317
1318 004442 042777 000004 174534  BIC #4, @ CSR

```

```
1320 ; DATA SWITCH 12 = 0, PRINT APPROPRIATE ERROR MESSAGE
1321
1322 004490 032737 010000 177570 15; BIT #SW12, SWR
1323 004496 001066 BNE 35
1324 004400 004537 005006 JSR R9, PRINT
1325 004414 005527 MNUMBER
1326 004416 004537 005100 JSR R9, TYPEOCTAL
1327 004472 001066 ERRORX ; (ERRORX) IS THE OPERAND TO BE OCTALIZED
1328 004474 005006 JSR R9, PRINT ; STARTING ADDRESS OF " SCOPE " LOOP
1329 004500 005527 MSCOPE
1330 004502 004537 005100 JSR R9, TYPEOCTAL
1331 004504 001066 PCSCOPE
1332 004510 004537 005006 JSR R9, PRINT ; PASS # IN OCTAL (177777 MAX)
1333 004514 005527 MPASS
1334 004516 004537 005100 JSR R9, TYPEOCTAL
1335 004522 001242 PASSLSB
1336 004524 004537 005006 JSR R9, PRINT ; ITERATION THROUGH ERROR SCOPE LOOP
1337 004530 005541 MITERATION
1338 004532 004537 005100 JSR R9, TYPEOCTAL
1339 004536 001230 ERRORS
1340
1341 ; IF THE ERROR IS #112, OR E120, OR E210, ALSO PRINT DATA ERROR (GOOD=BAD)
1342
1343 004540 022737 000112 001232 CMP #112, ERRORX
1344 004546 001410 BEO 25 ; E112
1345 004550 022737 000120 001232 CMP #120, ERRORX ; E120
1346 004556 001404 BEO 25
1347 004560 022737 000210 001232 CMP #210, ERRORX
1348 004566 001022 BNE 35
1349
1350 004570 004537 005066 25; JSR R9, PRINT ; E210
1351 004574 005591 MBUFFER
1352 004576 004537 005100 JSR R9, TYPEOCTAL
1353 004602 005460 BADDRESS
1354 004604 004537 005006 JSR R9, PRINT
1355 004610 005562 MGOOD
1356 004612 004537 005100 JSR R9, TYPEOCTAL
1357 004616 005462 GOOD
1358 004620 004537 005006 JSR R9, PRINT
1359 004624 005571 MBAD
1360 004626 004537 005100 JSR R9, TYPEOCTAL
1361 004632 005496 BAD
1362
1363 ; DATA SWITCH 15 = 0 HALT AT DETECTION OF ERROR
1364
1365 004634 032737 100000 177570 35; BIT #SW15, SWR
1366 004642 001001 BNE ,+4
1367 004644 000000 HALT
1368 004646 000002 RTI
```

```

1370                                     )           " SCOPE "
1371
1372 004698 005737 001238 XSCOPE: TST ERRORS
1373 004694 001015          ONE SCOPING          ) TO A SCOPE LOOP
1374
1375                                     ) NO ERRORS HAVE BEEN DETECTED HERE,
1376                                     ) JUST SET PCSCOPE = THE FIRST ADDRESS OF THE SCOPE LOOP
1377                                     ) (IN CASE ANY ERRORS ARE EVER DETECTED LATER)
1378
1379 004696 005837 001238 NOSCOPE: CLR ERRORS
1380 004662 011637 001244      MOV @ SP, PCSCOPE
1381 004666 000002          RTI
1382
1383                                     )           " SUBSCOPE "
1384
1385 004678 005737 001238 XSSCOPE: TST ERRORS
1386 004674 001001          ONE ,+4
1387 004676 000002          RTI          ) NO ERRORS EXIST - EXIT
1388
1389                                     ) ERRORS DO EXIST
1390
1391                                     ) IF THIS ERROR ADDRESS IS THE SAME ADDRESS WITHIN THE PROGRAM
1392                                     ) LOCATION " PCSSCOPE ", THEN THIS IS A SCOPE LOOP, IF NOT - EXIT
1393
1394 004700 021637 001246          CMP @ SP, PCSSCOPE
1395 004704 001401          BEQ ,+4
1396 004706 000002          RTI          ) IT ISN'T - EXIT
1397
1398                                     ) THIS IS A SCOPING LOOP
1399
1400 004710 032737 040000 177970 SCOPING: BIT @SW14, SWR
1401 004716 001397          ONE NOSCOPE
1402 004720 013716 001244      MOV PCSCOPE, @ SP
1403 004724 000002          RTI          ) EXIT

```

```
1405 004726 000000          VAI      0          ; KIT11D VECTOR ADDRESS
1406
1407                          ; IF THE CONTENTS OF PROGRAM LOCATION " IRQ " = 1, ANY IOT TRAP IS VALAD,
1408                          ; AND IT IS ASSUMED TO BE A KIT VECTOR I/O TRAP.
1409
1410 004730 011637 004726      KITCATCHER: MOV 0 SP, VA          ; KIT 11D VECTOR ADDRESS
1411 004734 102737 000004 004726      SUB #4, VA          ; BECAUSE PC=VA+2*2
1412 004742 012746 000340      MOV #340, -(SP)      ; RESET PRIORITY = 7 (AT EXIT)
1413 004746 013746 001236      MOV IRQOK, -(SP)    ; RETURN PC
1414 004752 005737 001234      TST IRQ
1415 004756 100010      RPL 15
1416
1417                          ; PRINT THE KIT'S VECTOR ADDRESS ON THE TTY
1418
1419 004760 004537 005006      JSR R5, PRINT
1420 004764 005464      MVECTOR          ; ADDRESS OF TEXT MESSAGE
1421 004766 004537 005100      JSR R5, TYPEOCTAL
1422 004772 004726      VA          ; VA (VECTOR ADDRESS) TO BE TYPED AS OCTAL
1423 004774 004737 005312      JSR PC, CRLF
1424 005000 005037 001234      ISI      CLR IRQ          ; RESET IRQ
1425 005004 000002      RTI
```