

SYSTEMS ENGINEERING LABORATORIES PROGRAM LIBRARY

SOFTWARE DESCRIPTION

CATALOG NO. 300010B

DOCUMENTATION REV N/A

DATE 11 NOVEMBER 1970

PROGRAM TITLE: SYSTEMS 810A/B DEBUG PACKAGE

PURPOSE: To provide a stand-alone debug capability.

CONFIGURATION: Basic SYSTEMS 810A/B with an ASR-33
and/or Variable Base Register

SOFTWARE ENVIRONMENT: Stand-Alone

PROGRAM LANGUAGE: SYSTEMS 810A/B Assembly Language

SIZE: 776₈ words

TIMING: N/A

SYSTEMS 300010B

REASON FOR CHANGE: To provide software capability on a SYSTEMS 810B with VBR.

- USE:
- 1.0 Loading Procedure: Load using the SYSTEMS 810A/B Standard load/Dump Package (Catalog No. 300001)
 - 2.0 Control Switch Settings:

CSW 0 - SET - High Speed Paper Tape System
CSW 0 - RESET - ASR-33 Paper Tape System
 - 3.0 Starting Location: The starting location for the DEBUG program is the relocation base used at load time.
 - 4.0 Procedure:
 - a. Load the starting location into the program counter and press START.
 - b. Set CSW 0 if a Load or Dump is to be made using high speed paper tape I/O.
 - c. Type in the keyword and address information required for the particular function to be used.
 - 5.0 Available Functions:
 - Type the contents of specified memory in octal or command format;
 - Modify the specified memory, (Input being in octal or command format);
 - Dump specified memory areas onto paper tape in absolute format;
 - Enter breakpoints in order to "Leap-frog" trace a program;
 - Clear specified areas of memory to zero;
 - Search memory for references to specified areas;
 - Initiate branches (or HALT AND BRANCH) to any part of memory;
 - Load a binary tape dumped by Debug or the Absolute Dump Program;

Each of these functions is initiated by typing a keyword through the console typewriter keyboard. This keyword consists of a letter an address (or addresses) and a terminator.

When a keyword requires two addresses (a lower and an upper bound) separate the address with a space, or comma.

SYSTEMS 300010B

USE: (Cont'd)

If an error is generated during input of the keyword but before the keyword was terminated, type a slash character (/). This will cause the keyword in error to be completely ignored, cause the typewriter to generate a carriage return, and request new keyword input. If the computer detects an error, it will initiate the same action automaticall

SYSTEMS 300010B

METHOD: 1.0 KEYWORDS AND THEIR MEANING:

● Type Memory Area (Octal):

Keyword: Taaaaa bbbbb cr

Keyword: Taaaaa cr

Definition: Type, in octal format, the memory words from Location aaaaa to and including bbbbb. If bbbbb is less than aaaaa or is not given, only the word at location aaaaa will be typed. Words are output four per line.

Example:

```
T100 110 cr
012111 000005 110400 177777 )
170056 000135 100300 077777 ) Output
177767 )
```

● Type All Registers:

Keyword: R cr

Definition: Type out the contents of the A, B, X and Variable Base Registers in octal format.

Example:

```
R cr
001745 177776 17776 0000XX
  A      B      X      VBR
```

● Type Memory Area (Command Format):

Keyword: Caaaaa bbbbb cr

Keyword: Caaaaa cr

Definition: Type in command format, the memory word from location aaaaa to and including bbbbb. If bbbbb is less than aaaaa or is not given, only the word at location aaaaa will be typed. Words are output four per line.

Example:

```
C100 110 cr
01.010.111 00.000.005 11.000.400 17.111.777
17/000.056 00.000.135 10.000.300 07.111.777
17.111.767
```

METHOD: (Cont'd)

- Input To Memory:
Keyword: laaaaa cr
Definition: Set the address where the next octal or command format input word is to be stored. If a sequence of word is to be stored, aaaaa would represent the starting address of that sequence.

- Octal Input Data:
Keyword: ddddd cr
Keyword: ddddd cr
Definition: The keyword is stored in memory at the location last specified by an laaaaa keyword. The address aaaaa is then incremented by 1. (When entering a sequential data block, it is not necessary to proceed each input word with an laaaaa keyword, only the first.)

- Command Format Input Data:
Keyword: 00.XIM.AAA cr
00 = OP Code (00-17)
X = Index Bit (0 or 1)
I = Indirect Bit (0 or 1)
M = Map Bit (0 or 1)
AAA = Address (000-777)
Definition: The keyword is condensed into binary form and stored into memory at the location last specified by an laaaaa keyword. The address aaaaa is then incremented by 1.

- Dump Memory On Paper Tape:
Keyword: Daaaaa bbbbb cr
Definition: The memory area from location aaaaa to and including location bbbbb will be punched on paper tape in absolute format compatible with the absolute load/dump programs. See "USE" for control switch settings.

METHOD: (Cont'd)

- Set Breakpoint:

Keyword: Baaaaa cr

Definition: Any outstanding breakpoints are reset. The contents of aaaaa and aaaaa + 1 are saved and a long brance (LOB) instruction to Debug is put in their place. When this instruction is executed it causes the original contents of these two locations to be restored, a line of output to be typed on the console typewriter as follows:

```
aaaaa 00.XIM.AAA XXXXXX YYYYYY
```

aaaaa = Location of instruction about to be executed.

00.XIM.AAA = Instruction in command format

XXXXXX = contents of A register before execution

YYYYYY = contents of B register before execution.

After typing this line, Debug pauses, waiting for a new keyword.

- Set Next Breakpoint:

Keyword: N cr

Definition: The address aaaaa from the last Baaaaa keyword is incremented by one and used as the new breakpoint location. After setting a breakpoint in this location a transfer is made to this location -1. In this way, the operator can trace each instruction in a sequential list of instructions with a minimum of effort.

- Clear Memory:

Keyword: Zaaaaa bbbbb cr

Keyword: Zaaaaa

Definition: The memory locations from location aaaaa to and including bbbbb are set to zero. If bbbbb is less than aaaaa or zero, location aaaaa only is set to zero.

SYSTEMS 300010B

METHOD: (Cont'd)

- Memory Search For Address:

Keyword: Maaaaa bbbbb cr

Xxxxx yyyy cr

Keyword: Maaaaa bbbbb cr

Xxxxx cr

(Note: xxxx, and yyyy are limited to values between 0_8 and 1777_8)

Definition: Search memory from location aaaaa to and including location bbbbb for any word whose last ten bits are in the range from xxxx to yyyy. For each such word type the location of the word and the word itself in octal format. If yyyy is not specified, search for xxxx only.

- Search Previous Memory Area:

Keyword: Xxxxx yyyy

Keyword: Xxxxx

Definition: Same as above except that the area searched is the one specified by the last Maaaaa bbbbb keyword.

Memory search examples:

MO 1000	Keyword
X100 110	
00005 000100)	
00770 030107)	
00772 020110)	Output
00773 040105)	
00777 020101)	
01000 102100)	
X200	Keyword
00500 022200)	Output
00776 156200)	

- Start Computer:

Keyword: Saaaaa cr

Definition: When entering the DEBUG program either initially or by breakpoint entry, the contents of the registers are saved. The Saaaaa keyword will cause the contents of the registers to be restored and then executes an unconditional branch (BRU) to location aaaaa.

- Halt And Branch:

Keyword: Haaaaa cr

Definition: This keyword is identical to the Saaaaa keyword except that after the registers are restored a halt takes place before the LAA and LBA instructions are executed. This allows the operator to switch to the single cycle mode.

- Load Binary Tape:

Keyword: L cr

Definition: This keyword will load a paper tape dumped in absolute format by either DEBUG or the absolute dump program. If a checksum error is detected a "K" will be typed out on the console typewriter.

2.0 KEYWORD SUMMARY

<u>Keyword</u>	<u>Definition</u>
Taaaaa bbbbb cr	Type Memory Area In Octal Format
Taaaaa cr	Type Memory Word In Octal Format
R cr	Type All Registers In Octal Format
Caaaaa bbbbb cr	Type Memory Area In Command Format
Caaaaa cr	Type Memory Word In Command Format
Iaaaaa cr	Set Next Input Address
dddd cr	Octal Input Data
-dddd cr	Negative Octal Input Data
00.XIM.aaa cr	Command Format Input Data
Daaaaa bbbbb cr	Dump Memory Area Onto Paper Tape In Absolute Format
Baaaaa cr	Set Breakpoint

METHOD: (Cont'd)

<u>Keyword</u>	<u>Definition</u>
N cr	Set Breakpoint Into Next Location
Zaaaaa aaaaa	Set Memory Area To Zero
Zaaaaa	Set Memory Word to Zero
Maaaaa bbbbb cr Xxxxx yyyy cr	Search Memory Area For Specified Address Range
Maaaaa bbbbb cr Xxxxx cr	Search Memory Area For Specified Address
Xxxxx yyyy cr	Search Previous Memory Area For Specified Address Range
Xxxxx cr	Search Previous Memory Area For Specified Address
Saaaaa cr	Start Computer
Haaaaa cr	Halt; Then Start
L cr	Load Absolute Tape


```

0001      *      300010B      810A/B SYSTEMS DEBUG PACKAGE
0002      *
0003      *
0004      *
0005      *          S. SW 0 SET LOAD AND/OR DUMP FROM H.S. READER/PUNCH
0006      *
0007      *          RESET LOAD AND/OR DUMP FROM ASR-33
0008      *
0009      *
0010      *
0011      *
0012      00000 00000000      REL
0013      00000 70000000      BRG  '0
0014      00000 11100020      DEBUG BRU  DXX
0015      00001 01300112      TABL  LAA*  D64          (00) .....(BLANK TAPE)
0016      00002 35400260      DAC   D230          (01)
0017      00003 02300403      LBA*  D320          (02) B....SET BREAKPOINT
0018      00004 02300273      LBA*  D264          (03) C....COMMAND TYPE-OUT
0019      00005 02300604      LBA*  DUMP          (04) D....DUMP BINARY TAPE
0020      00006 00000000      SAVE ZZ4  **
0021      00007 03100562      STA   A
0022      00010 04100563      STB   B
0023      00011 02300264      LBA*  D240          (10) H....HALT AND START
0024      00012 02300235      LBA*  D200          (11) I....SET INPUT ADDR.
0025      00013 01100773      LAA   SX47
0026      00014 11100770      BRU   XPND
0027      00015 02300702      LBA*  LOAD          (14) L....LOAD BINARY TAPE
0028      00016 02300472      LBA*  D360          (15) M....MEMORY SEARCH
0029      00017 02300445      LBA*  D340          (16) N....NEXT BREAKPOINT
0030      00020 12100006      DXX   SPB  SAVE
0031      00021 11100052      BRU   D32
0032      00022 00000000      D15   ***  **
0033      00023 02300363      LBA*  D300          (22) R....REGISTER TYPE
0034      00024 02300265      LBA*  D250          (23) S....START COMPUTE
0035      00025 02300271      LBA*  D260          (24) T....TYPE MEMORY AREA
0036      00026 00000000      D20   ***  **          (25)
0037      00027 01100042      LAA   D28          (26)          OUTPUT A C.R.
0038      00030 11100035      BRU   D26          (27)
0039      00031 02300501      LBA*  D370          (30) X....SEARCH ADDRESS
0040      00032 00000057      D22   DATA '57          (31) (CODE FOR *)

```

0041	00033	02300073		LBA*	D40	(32)	7,....ZFRØ MEMORY AREA
0042	00034	00000033		NØP		(33)	
0043	00035	12100542	D26	SPB	TYP1	(34)	
0044	00036	01100043		LAA	D29	(35)	OUTPUT A L.F.
0045	00037	12100542		SPB	TYP1	(36)	
0046	00040	11300026		BRU*	D20	(37)	
0047	00041	05200000		AMA*	0	(40)	(SP)
0048	00042	00000215	D28	DATA	'215	(41)	(CODE FØR C.R.)
0049	00043	00000012	D29	DATA	'12	(42)	(CODE FØR LINE FD)
0050	00044	00000000	D30	***	**	(43)	INCREMENT ADDR.
0051	00045	14100565		IMS	Y	(44)	
0052	00046	01100565		LAA	Y	(45)	CHECK FØR UPPER END
0053	00047	15100564		CMA	X	(46)	
0054	00050	11100077		BRU	D42	(47)	RETURN
0055	00051	11100077		BRU	D42	(50)	RETURN
0056	00052	11100104	D32	BRU	D60	(51)	OUTPUT A C.R./L.F.
0057	00053	00000000	D16	***	**		
0058	00054	00000040	D34	DATA	'40	(53)	(CODE FØR SPACE)
0059	00055	05200000		AMA*	0	(54)	,
0060	00056	02300240		LBA*	D210	(55)	-
0061	00057	14300243		IMS*	D220	(56)	.
0062	00060	01600000		LAA*	0,1	(57)	/ (ERRØR)
0063	00061	11200000		BRU*	0	(60)	0
0064	00062	11200000		BRU*	0	(61)	1
0065	00063	11200000		BRU*	0	(62)	2
0066	00064	11200000		BRU*	0	(63)	3
0067	00065	11200000		BRU*	0	(64)	4
0068	00066	11200000		BRU*	0	(65)	5
0069	00067	11200000		BRU*	0	(66)	6
0070	00070	11200000		BRU*	0	(67)	7
0071	00071	01600000		LAA*	0,1	(70)	8 (ERRØR)
0072	00072	01600000		LAA*	0,1	(71)	9 (ERRØR)
0073	00073	00000003	D40	CLA		(72)	ZERØ MEMORY
0074	00074	03300565		STA*	Y	(73)	
0075	00075	12100044		SPB	D30	(74)	INCREMENT ADDR.
0076	00076	11100073		BRU	D40	(75)	
0077	00077	11300044	D42	BRU*	D30	(76)	TØ EXIT
0078	00100	01200000		LAA*	0	(77)(DELETE CODE)
0079				*			
0080				*			
0081	00101	01100032	D58	LAA	D22		= / CHAR.

0082	00102	12100542		SPB	TYP1		OUTPUT A SLASH CHAR.	
0083	00103	12100026	D59	SPB	D20		OUTPUT A CARRIAGE RETURN	
0084	00104	01100034	D60	LAA	D22+2			
0085	00105	03100153		STA	D117			
0086	00106	00000003		CLA				
0087	00107	03100574		STA	ERFL			
0088	00110	03100565		STA	Y		FLAG = 0	
0089	00111	03100566		STA	KC		KEY CHARACTER = 0	
0090	00112	00000003	D64	CLA				
0091	00113	03100564		STA	X		SFT X = 0	
0092	00114	00130101	D70	CEU	1,N			OKA
0093	00115	00002000		DATA	'002000		ENABLE KEYBOARD	
0094	00116	00170301		AIF	1,N			OKA
0095	00117	03100603		STA	CHAR			
0096	00120	12100542		SPB	TYP1			
0097	00121	01100603		LAA	CHAR			
0098	00122	15100042		CMA	D28		CHECK FOR C.R.	
0099	00123	11100125		BRU	**2			
0100	00124	11100207		BRU	D130		KEYWORD TERMINATED (C.R.)	
0101	00125	02100576		LBA	K2		CONSTANT = '77	
0102	00126	00000027		ABA			EXTRACT TO SIX BITS	
0103	00127	00000005		TAB			SFT CHAR, INTO INDEX	
0104	00130	01500001		LAA	TABL,1			
0105	00131	00001215		RSL	10		SAVE ONLY INDIRECT BIT	
0106	00132	00001716		LSL	15			
0107	00133	00000024		SAP			SKIP IF NOT A KEY CHAR.	
0108	00134	01500001		LAA	TABL,1			
0109	00135	00001315		RSL	11		PROCESS CODE IN BITS 0-4	
0110	00136	00000006		TAB			PROCESS CODE TO INDEX	
0111	00137	11500140		BRU	D110,1		PROCESS KEY CHARACTER	
0112			*					
0113	00140	01100566	D110	LAA	KC		(0)....NOT A KEY CHARACTER	
0114	00141	00000022		SAZ			SKIP IF NO KEY CHAR.	
0115	00142	11100101		BRU	D58			
0116	00143	11100101		BRU	D58		(3) OUTPUT / (CR)	
0117			*					
0118	00144	00000005	D115	TAB			(4)....KEY CHARACTER	
0119	00145	01100566	D114	LAA	KC			
0120	00146	00000022		SAZ			SKIP IF NO KEY CHAR.	
0121	00147	11100101		BRU	D58			
0122	00150	04100566	D115	STB	KC		C=KEY CHAR TABLE POS.	

```

0123 00151 11100114      BRU    D70
0124
0125 00152 11100177 D110 BRU    D120      (12)....(SP) (,)
0126 00153 00000033 D117 NOP
0127 00154 01100564      LAA    X
0128 00155 03100565      STA    Y
0129 00156 01100151      LAA    D114+4
0130 00157 11100201      BRU    D122
0131 00160 11100702      BRU    LOAD
0132 00161 11100360      BRU    D297      (21)....REGISTER PRINT
0133
0134 00162 00001516      *      LSL    13      (22)....0 TO 7
0135 00163 02100564      LBA    X
0136 00164 00000314      FRL    3      LONG LEFT ROTATE
0137 00165 04100564      STB    X      SHIFT CHAR. INTO X
0138 00166 02100575      LBA    K1      CONSTANT =1
0139 00167 11100203      BRU    D124
0140 00170 02100564      LBA    X      (30)....(DEC. POINT)
0141 00171 04100565      STB    Y      SFT Y TO X
0142 00172 11100174      BRU    **2
0143
0144 00173 11100445      *      BRU    D340      (33)....SET NEXT BREAKPT.
0145 00174 01100601      LAA    K5      (,)
0146 00175 03100566      STA    KC
0147 00176 11100114      BRU    D70
0148 00177 14100574 D120 IMS ERFL
0149 00200 11100153      BRU    D117
0150 00201 03100153 D122 STA D117
0151 00202 11100112      BRU    D64
0152 00203 01100566 D124 LAA <C
0153 00204 00000022      SAZ
0154 00205 11100114      BRU    D70
0155 00206 11100150      BRU    D115
0156
0157      *      -----PROCESS A C.R. CHARACTER
0158 00207 01100043 D130 LAA    D29      OUTPUT A L/F
0159 00210 12100542      SPB    TYP1
0160 00211 01100566      LAA    KC      IF KC = (,)--
0161 00212 15100601      CMA    K5      X AND Y CONTAIN
0162 00213 11100215      BRU    **2      COMMAND INPUT DATA
0163 00214 11100227      BRU    D140

```



```

0205 00260 01100564 D230 LAA X
0206 00261 03300567 D231 STA* ADDR STORE INTO MEMORY
0207 00262 14100567 IMS ADDR INCREMENT MEMORY ADDR.
0208 00263 11100104 BRU D60
0209 *
0210 *-----HALT, THEN START
0211 00264 00000000 D240 HLT
0212 00265 01100562 D250 LAA A RESTORE REGISTERS
0213 00266 02100563 RSTR LBA B
0214 00267 00000000 XPNT ZZL **
0215 00270 11300564 BRU* X
0216 *
0217 *-----PRINT MEMORY AREA IN OCTAL FORMAT
0218 00271 01100365 D260 LAA D304
0219 00272 11100276 BRU D270
0220 *
0221 *-----PRINT MEMORY AREA IN COMMAND FORMAT
0222 00273 01100275 D264 LAA D265
0223 00274 11100276 BRU D270
0224 00275 12100324 D260 SPB D290 OUTPUT IN COMMAND FORMAT
0225 *
0226 *
0227 00276 03100303 D270 STA D275 SET OUTPUT MODE
0228 00277 01100600 D271 LAA K4 CONSTANT =-4
0229 00300 03100570 STA CNT SET COUNTER TO -4
0230 00301 12100547 D273 SPB TPSP TYPE SPACE
0231 00302 01300565 LAA* Y
0232 00303 00000000 D273 *** ** OUTPUT OCTAL OR COMMAND
0233 00304 14100565 IMS Y Y = Y+1
0234 00305 00000033 NOP
0235 00306 01100564 LAA X
0236 00307 15100565 CMA Y
0237 00310 11100103 BRU D59 DONE
0238 00311 00000033 NOP
0239 00312 14100570 IMS CNT COUNT = COUNT+1
0240 00313 11100301 BRU D273 IF NOT 4 IN THIS LINE
0241 00314 12100026 SPB D20
0242 00315 11100277 BRU D271
0243 *-----SUBROUTINE TO LIST WORD IN OCTAL FORMAT
0244 00316 00000000 D280 *** ** RETURN ADDR
0245 00317 00000005 TAB

```

```

0246 00320 00000003      CLA
0247 00321 12100344      SPB  D296      OUTPUT 1 BIT DIGIT
0248 00322 12100553 D284 SPB  AT15
0249 00323 11300316      BRU*  D280      EXIT SUBR.
0250
0251 00324 00000000 D290 ***  **      RETURN ADDR
0252 00325 00000005      TAB
0253 00326 00000003      CLA      SET ACC=0
0254 00327 12100344      SPB  D296      OUTPUT 1 BIT DIGIT
0255 00330 12100351      SPB  D298      OUTPUT 3 BIT DIGIT
0256 00331 01100356      LAA  D299      = , CHAR
0257 00332 12100542      SPB  TYP1      OUTPUT 1 CHAR.
0258 00333 12100344      SPB  D296      OUTPUT 1 BIT DIGIT
0259 00334 12100344      SPB  D296      OUTPUT 1 BIT DIGIT
0260 00335 12100344      SPB  D296      OUTPUT 1 BIT DIGIT
0261 00336 01100356      LAA  D299      = , CHAR
0262 00337 12100542      SPB  TYP1      OUTPUT 1 CHAR.
0263 00340 12100351      SPB  D298      OUTPUT 3 BIT DIGIT
0264 00341 12100351      SPB  D298      OUTPUT 3 BIT DIGIT
0265 00342 12100351      SPB  D298      OUTPUT 3 BIT DIGIT
0266 00343 11300324      BRU*  D290      EXIT SUBROUTINE
0267
0268
0269 00344 00000000 D290 ***  **      RETURN ADDR
0270 00345 00000113      FLL  1
0271 00346 05100357      AMA  D292      =160
0272 00347 12100542      SPB  TYP1      OUTPUT 1 CHAR. AND CLEAR
0273 00350 11300344      BRU*  D296
0274
0275 00351 00000000 D290 ***  **      RETURN ADDR
0276 00352 00000313      FLL  3
0277 00353 05100357      AMA  D292      =160
0278 00354 12100542      SPB  TYP1      OUTPUT 1 CHAR. AND CLEAR
0279 00355 11300351      BRU*  D298
0280
0281 00356 00000056 D299 DATA 1056      CODE FOR .
0282 00357 00000060 D292 DATA 1060      CODE FOR 0
0283
0284
0285 00360 01100566 D297 LAA  KC      IGNORE IF NOT FIRST CHAR
0286 00361 00000022      SAZ

```

```

0287 00362 11100114 BRU D70
0288 00363 12100547 D300 SPB TPSP
0289 00364 01100562 LAA A
0290 00365 12100316 D304 SPB D280 OUTPUT OCTAL FORMAT
0291 00366 12100547 SPB TPSP TYPE SPACE
0292 00367 01100563 LAA B
0293 00370 12100316 SPB D280 OUTPUT OCTAL FORMAT
0294 00371 12100547 SPB TPSP
0295 00372 00000046 XPX
0296 00373 00000053 IXA
0297 00374 00000047 XPB
0298 00375 12100316 SPB D280
0299 00376 12100547 SPB TPSP
0300 00377 00000043 TVB
0301 00400 00000713 FLL 7
0302 00401 12100316 SPB D280
0303 00402 11100103 BRU D59
0304
0305 *
*-----SET BREAKPOINT ADDRESS
0306 00403 01300420 D320 LAA* D326 LAST BREAKPOINT LOC.
0307 00404 06100423 SMA D328 BRANCH INSTR. TO DEBUG
0308 00405 00000022 SAZ
0309 00406 11100414 BRU D324 LAST BKPT PROPERLY RESET
0310 00407 01100421 LAA D327 RESET LAST BREAKPOINT
0311 00410 03300420 STA* D326
0312 00411 14100420 IMS D326
0313 00412 01100422 LAA D327+1
0314 00413 03300420 STA* D326
0315 00414 01100564 D324 LAA X
0316 00415 03100420 STA D326 SET NEW BREAKPOINT LOC.
0317 00416 12100454 SPB SBPT SET BREAKPOINT
0318 00417 11100104 BRU D60 RETURN
0319 *
0320 00420 00000000 D320 *** ** LOCATION OF BREAKPOINT
0321 00421 00000000 D327 *** ** DATA FROM BREAKPOINT LOC.
0322 00422 00000000 *** **
0323 00423 00000036 D320 L0B BRU TO DEBUG PR0G.
0324 00424 35400425 DAC D330
0325 *-----RETURN FROM BREAKPOINT LOCATION
0326 *
0327 00425 12100006 D330 SPB SAVE

```

```

0328 00426 01100420      LAA    D326                    LOCATION OF BREAKPOINT
0329 00427 12100316      SPB    D280                    OUTPUT IN OCTAL FORMAT
0330 00430 12100547      SPB    TPSP
0331 00431 01100421      LAA    D327                    BREAKPOINT DATA
0332 00432 12100324      SPB    D290                    OUTPUT IN COMMAND FORMAT
0333 00433 12100547      SPB    TPSP                    TYPE SPACE
0334 00434 01100421      LAA    D327                    RESET BREAKPOINT DATA
0335 00435 03300420      STA*   D326
0336 00436 14100420      IMS    D326
0337 00437 01100422      LAA    D327+1
0338 00440 03300420      STA*   D326
0339 00441 01100420      LAA    D326                    RESET BREAKPOINT LOC
0340 00442 05100602      AMA    K8                        FOR -N- NEXT BREAKPOINT
0341 00443 03100420      STA    D326
0342 00444 11100363      BRU    D300                    LIST REGISTERS
0343                    *
0344                    *-----SET NEXT BREAKPOINT
0345 00445 01100042      D340 LAA    D28
0346 00446 12100542                    SPB    TYP1
0347 00447 02100420                    LBA    D326
0348 00450 14100420                    IMS    D326                    ADVANCE BREAKPOINT ADDR
0349 00451 12100454                    SPB    SBPT                    SET BREAKPOINT
0350 00452 04100564                    STB    X
0351 00453 11100265                    BRU    D250
0352 00454 00000000      SBPT ZZZ                    **
0353 00455 01300420                    LAA*   D326                    SAVE DATA FROM BKPT. LOC.
0354 00456 03100421                    STA    D327
0355 00457 14100420                    IMS    D326
0356 00460 01300420                    LAA*   D326
0357 00461 03100422                    STA    D327+1
0358 00462 01100424                    LAA    D328+1                    SET BRANCH INTO BKPT. LOC.
0359 00463 03300420                    STA*   D326
0360 00464 01100420                    LAA    D326
0361 00465 05100602                    AMA    K8
0362 00466 03100420                    STA    D326
0363 00467 01100423                    LAA    D328
0364 00470 03300420                    STA*   D326
0365 00471 11300454                    BRU*   SBPT
0366                    *
0367                    *-----SET MEMORY SEARCH LIMITS
0368 00472 01100577      D360 LAA    K3                        PRFSET SEARCH MASK ('1777)

```

```

0369 00473 03100567 STA ADDR
0370 00474 01100565 LAA Y SET UPPER SEARCH ADDR.
0371 00475 03100022 STA D15
0372 00476 01100564 LAA X SET LOWER SEARCH ADDR.
0373 00477 03100053 STA D16
0374 00500 11100052 BRU D32
0375
0376 *
*-----SEARCH MEMORY AREA
0377 00501 02100567 D370 LBA ADDR SEARCH MASK (USUALLY='1777
0378 00502 01100564 LAA X RJT CAN BE SET BY I)
0379 00503 00000027 ABA SET TRUNCATED LOWER ADDR.
0380 00504 03100564 STA X
0381 00505 01100565 LAA Y
0382 00506 00000027 ABA SET TRUNCATED UPPER ADDR.
0383 00507 03100565 STA Y
0384 00510 02100022 LBA D15 SET CURRENT SEARCH ADDR.
0385 00511 04100775 STB D17
0386 00512 01300775 D374 LAA* D17 CONTENTS OF CURRENT ADDR.
0387 00513 02100567 LBA ADDR
0388 00514 00000027 ABA MASK ADDR. BITS
0389 00515 15100564 CMA X UPPER SEARCHED-FOR ADDR.
0390 00516 11100521 BRU D376 MAYBE
0391 00517 11100524 BRU D380 YES
0392 00520 11100534 BRU D382 NO
0393 00521 15100565 D370 CMA Y LOWER SEARCHED-FOR ADDR.
0394 00522 11100534 BRU D382 NO
0395 00523 11100524 BRU D380 YES
0396 00524 02100775 D380 LBA D17 CURRENT SEARCH ADDR.
0397 00525 00000113 FLL 1
0398 00526 00000003 CLA
0399 00527 12100553 SPB AT15 OUTPUT 15 OCTAL DIGITS
0400 00530 12100547 SPB TPSP TYPE SPACE
0401 00531 01300775 LAA* D17 OUTPUT WORD IN OCTAL
0402 00532 12100316 SPB D280
0403 00533 12100026 SPB D20 OUTPUT A C.R./L.F.
0404 00534 14100775 D382 IMS D17 INCREMENT REARCH ADDR.
0405 00535 01100775 LAA D17
0406 00536 15100053 CMA D16 UPPER ADDR
0407 00537 11100512 BRU D374 CONTINUE SEARCH
0408 00540 11100512 BRU D374
0409 00541 11100103 BRU D59

```

```

0410
0411 *-----SUBR, TO TYPE CHAR, IN A REG.
0412 00542 25400000 TYP1 DAC **
0413 00543 00001016 LSL R
0414 00544 00170101 AOP 1,W
0415 00545 00000003 CLA
0416 00546 11300542 BRU* TYP1
0417 *
0418 ***** TYPE A SPACE
0419 00547 25400000 TPSP DAC **
0420 00550 01100054 LAA D34
0421 00551 12100542 SPB TYP1
0422 00552 11300547 BRU* TPSP
0423 00553 25400000 WT15 DAC **
0424 00554 12100351 SPB D298
0425 00555 12100351 SPB D298
0426 00556 12100351 SPB D298
0427 00557 12100351 SPB D298
0428 00560 12100351 SPB D298
0429 00561 11300553 BRU* WT15
0430 00562 25400000 A DAC **
0431 00563 25400000 B DAC **
0432 00564 25400000 X DAC **
0433 00565 25400000 Y DAC **
0434 00566 25400000 KC DAC **
0435 00567 25400000 ADDR DAC **
0436 00570 25400000 CNT1 DAC **
0437 00571 00000000 CKSM DATA 0
0438 00572 00000000 WCNT DATA 0
0439 00573 00000000 STAD DATA 0
0440 00574 00000000 ERFL DATA 0
0441 00575 00000001 K1 DATA 1
0442 00576 00000077 K2 DATA '77
0443 00577 00001777 K3 DATA '1777
0444 00600 00177774 K4 DATA -4
0445 00601 00000056 K5 DATA '56
0446 00602 00177777 K8 DATA '177777
0447 00603 00000000 CHAR ZZZ **
0448 *-----DUMP MEMORY ONTO TAPE
0449 00604 01100701 DUMP LAA AOP1 AOP 1,W
0450 00605 00130400 SNS 0

```

CKA

OUTPUT 5 3-BIT DIGITS

```

A REGISTER SAVE
B REGISTER SAVE
LOWER LIMIT ADDR.
UPPER LIMIT ADDR.
KEY CHARACTER
INPUT ADDR.
COUNTER
CHECKSUM
WORD COUNT
START ADDR
ERROR FLAG

```

CHAR FROM KEYBOARD

DUMP MEMORY ONTO TAPE

AOP 1,W

TEST FOR WHICH PUNCH

0451	00606	11100610	BRU	**2	HIGH SPEED
0452	00607	11100613	BRU	**4	ASR-33
0453	00610	05100575	AMA	K1	H.S. IS UNIT 2
0454	00611	00130102	CEU	2,W	H.S. PUNCH POWER ON
0455	00612	00004000	DATA	*4000	
0456	00613	03100664	STA	A1	
0457	00614	03100666	STA	A2	
0458	00615	12100672	SPB	LDR	PUNCH LEADER
0459	00616	01100576	LAA	K2	START CODE
0460	00617	12100662	SPB	WDOT	
0461	00620	01100565	LAA	Y	OUTPUT STARTING ADDRESS
0462	00621	12100662	SPB	WDOT	
0463	00622	00000005	IAB		USE AS INDEX
0464	00623	06100564	SMA	X	ENDING ADDRESS
0465	00624	05100602	AMA	K8	-1
0466	00625	03100640	STA	NWCT	NEG WORD COUNT
0467	00626	12100662	SPB	WDOT	
0468	00627	01100737	UNIT LAA	M100	
0469	00630	03100572	STA	WCNT	NEGATIVE BLOCK COUNT
0470	00631	00000003	CLA		INITIALIZE CHECK SUM
0471	00632	03100571	STA	CKSM	
0472	00633	01400000	INPUT LAA	0,1	WORD FROM MEMORY
0473	00634	12100662	SPB	WDOT	
0474	00635	05100571	AMA	CKSM	UPDATE CHECK SUM
0475	00636	03100571	STA	CKSM	
0476	00637	00000026	IBS		INCREMENT OUTPUT INDEX
0477	00640	00000000	NWCT HLT		
0478	00641	14100640	IMS	NWCT	TEST FOR DUMP FIN.
0479	00642	11100644	BRU	**2	NOT FIN.
0480	00643	11100646	BRU	CSUM	
0481	00644	14100572	IMS	WCNT	TEST FOR BLOCK FIN.
0482	00645	11100633	BRU	INPUT	NOT FIN.
0483	00646	01100571	CSUM LAA	CKSM	OUTPUT CHECK SUM
0484	00647	12100662	SPB	WDOT	
0485	00650	01100640	LAA	NWCT	TEST FOR DUMP FIN.
0486	00651	00000024	SAP		
0487	00652	11100627	BRU	UNIT	
0488	00653	12100672	SPB	LDR	PUNCH TRAILER
0489	00654	00130400	SNS	0	
0490	00655	11100657	BRU	**2	
0491	00656	11100103	BRU	059	

```

0492 00657 00130102      CEU  2,W      TURN OFF PUNCH
0493 00660 00002000      DATA '2000
0494 00661 11100103      BRU  059
0495 00662 00000000      WDR#T ZZZ **
0496 00663 03100671      STA  7ZZ1
0497 00664 00000033      A1  NØP      AØP UNIT , W
0498 00665 00001016      LSL  8
0499 00666 00000033      A2  NØP      AØP UNIT , W
0500 00667 01100671      LAA  7ZZ1
0501 00670 11300662      BRU* WDR#T
0502 00671 00000000      ZZZ1 HLI
0503 00672 00000000      LDR  ZZZ **
0504 00673 02100737      LBA  M100
0505 00674 00000003      CLA
0506 00675 12100662      SPB  WDR#T
0507 00676 00000026      IBS
0508 00677 11100675      BRU  *-2
0509 00700 11300672      BRU* LDR
0510 00701 00170101      AØP1 AØP 1,W
0511
0512      *-----LOAD BINARY PAPER TAPE
0513      *
0514 00702 01100767      LØAU LAA  AIP1      AIP 1,W
0515 00703 00130400      SNS  0      TEST FOR WHICH READER
0516 00704 11100710      BRU  **4      HIGH SPEED
0517 00705 00130101      CEU  1,W      ASR-33
0518 00706 00004000      ZZZ2 DATA '4000      READER ENABLE
0519 00707 11100713      BRU  **4
0520 00710 05100575      AMA  K1      UNIT 2
0521 00711 00130102      CEU  2,W      H,S. READER ENABLE
0522 00712 00001000      DATA '1000
0523 00713 03100717      STA  A3
0524 00714 03100763      STA  A4
0525 00715 05100706      AMA  7ZZ2      ADD IN MERGE BIT
0526 00716 03100765      STA  A5
0527 00717 00000033      A3  NØP
0528 00720 00000022      SAZ      TEST FOR LEADER
0529 00721 11100723      BRU  **2
0530 00722 11100717      BRU  *-3      IGNORE LEADER
0531 00723 12100762      SPB  INWD      INPUT WORD
0532 00724 03100565      STA  Y      STARTING ADDRESS

```

0533	00725	00000005	TAB		USE AS INDEX
0534	00726	12100762	SPB	INWD	
0535	00727	03100640	STA	NWCT	NEGATIVE WORD COUNT
0536	00730	01100737	INIT	LAA	M100
0537	00731	03100572	STA	WCNT	NEGATIVE BLOCK COUNT
0538	00732	00000003	CLA		INITIALIZE CHECKSUM
0539	00733	03100571	STA	CKSM	
0540	00734	12100762	INPT	SPB	INWD
0541	00735	03400000	STA	0,1	INPUT TO MEMORY
0542	00736	00000026	IBS		
0543	00737	00177700	M100	DATA	-64
0544	00740	05100571	AMA	CKSM	UPDATE CHECKSUM
0545	00741	03100571	STA	CKSM	
0546	00742	14100640	IMS	NWCT	
0547	00743	11100745	BRU	**2	
0548	00744	11100747	BRU	CSML	LOAD COMPLETE
0549	00745	14100572	IMS	WCNT	TEST FOR BLOCK COMPLETE
0550	00746	11100734	BRU	INPT	NOT
0551	00747	12100762	CSML	SPB	INWD
0552	00750	15100571	LMA	CKSM	
0553	00751	11100753	BRU	CK	CHECKSUM BAD
0554	00752	11100756	BRU	OK	
0555	00753	00170501	CK	NOP	1, W
0556	00754	00142640		DATA	'K1'
0557	00755	00000000	HLI		
0558	00756	01100640	OK	LAA	NWCT
0559	00757	00000024		SAP	
0560	00760	11100730		BRU	INIT
0561	00761	11100103		BRU	D59
0562	00762	00000000	INWD	LZZ	**
0563	00763	00000033	A4	NOP	AIP UNIT, W
0564	00764	00001016		LSL	8
0565	00765	00000033	A5	NOP	AIP UNIT, W, R
0566	00766	11300762		BRU*	INWD
0567	00767	00170501	AIP1	AIP	1, W
0568			*		
0569	00770	00000050	XPND	SXB	
0570	00771	06100575		SMA	K1
0571	00772	03100267		STA	XPNT
0572	00773	00000047	SX4/	XPB	
0573	00774	11300006		BRU*	SAVE

0574 00775 00000000 D17 *** **
0575 00776 70400000 END DEBUG
ERRORS 0000 00000

...EXTERNALS...

...SYMBOLICS...

A	21	*	430							
A1	450	*	497							
A2	457	*	499							
A3	523	*	527							
A4	524	*	563							
A5	526	*	565							
ADDR	* 435									
AIP1	514	*	567							
AOP1	449	*	510							
B	22		213	*	431					
CHAR	95		97	*	447					
CK	553	*	555							
CKSM	* 437		471		474	475	483	539	544	545
C0NT	* 436									552
CSML	548	*	551							
CSUM	480	*	483							
D110	* 113									
D113	* 118									
D114	* 119		129							
D115	* 122		155							
D116	* 125									
D117	85	*	126		149	150	168			
D120	125	*	148							
D122	130	*	150							
D124	139	*	152							
D130	* 158									
D140	171	*	174							
D15	* 32									
D16	* 57									
D17	* 574									
D20	* 36		241							
D200	* 182									
D210	165	*	186							
D22	* 40		84		169					
D220	* 190									
D230	* 205									
D231	* 206									
D240	* 211									
D250	* 212		351							

D26	*	43				
D260	*	218				
D264	*	222				
D266	*	224				
D270	*	227				
D271	*	228				
D273	*	230				
D275	*	232				
D28	*	48				
D280	*	244	298	302		
D284	*	248				
D29	*	49				
D290	*	251				
D292	*	282				
D296	*	247	* 269			
D297	*	285				
D298	*	275				
D299	*	281				
D30	*	50				
D300	*	33	* 288			
D304	*	290				
D32	*	31	* 56			
D320	*	306				
D324	*	315				
D326	*	320				
D327	*	321				
D328	*	323				
D330	*	327				
D34	*	58	420			
D340	*	29	* 345			
D360	*	368				
D370	*	377				
D374	*	386				
D376	*	393				
D380	*	396				
D382	*	404				
D40	*	73				
D42	*	77				
D58	*	81	115	121	167	
D59	*	83	409	491	494	561
D60	*	56	* 84	318		

D64	*	90	151							
D70	*	92	154							
DEBUG	*	14	272							
DUMP		19	* 449							
DXX		14	* 30							
ERFL		87	148	164	* 440					
INIT	*	236	260							
INPT	*	540	250							
INWD		531	234	540	251	* 562	566			
K1	*	441	453	520	270					
K2	*	442	459							
K3	*	443								
K4	*	444								
K5	*	445								
K8	*	446	462							
KC		152	* 434							
LDR		458	488	* 503	209					
L0AD		27	131	* 514						
M100		468	204	236	* 243					
NWCT		466	* 477	478	485	532	246	558		
OK		554	* 258							
ONIT	*	468	487							
OPUT	*	472	482							
OT15		248	299	* 423	429					
RSTR	*	213								
SAVE	*	20	30	327	273					
SBPT		317	249	* 352	362					
STAD	*	439								
SX47		22	* 272							
TABL	*	12								
TPSP		230	288	291	294	299	330	333	400	* 419
		422								
TYP1		96	* 412	421						
W		92	94	414	454	492	210	217	221	222
		267								
WCNT	*	438	469	481	237	249				
WDOT		460	462	467	473	484	* 495	201	206	
X		127	172	212	250	* 432	464			
XPND		26	* 269							
XPNT	*	214	271							
Y		128	173	* 433	461	232				

ZZZ1
ZZZ2

* 490
* 510

200
225

* 502