

/PROGRAM TITLE:  
/IDENTIFICATION:  
/DATE CREATED:  
/AUTHOR:  
/MAINTAINED BY:

PDP-12 BASIC MEMORY CONTROL TEST  
MAINDEC 12-D1FA-D  
28 FEBRUARY, 1971  
HAROLD LONG  
DIAGNOSTIC GROUP

MEMCT  
RSW 65008  
RACDE  
SYNAP-20

inhibit per console RSW 6:1

sel 7777 = 5301

2

3

4

5

/PDP-12 MAINDEC 12-D1FA-L BASIC MEMORY CONTROL TEST  
/COPYRIGHT, 1970, 1971, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

/AUTHOR: HAROLD LONG

/THIS TEST IS DESIGNED TO EXERCISE ALL MEMORY  
/CONTROL INSTRUCTIONS AVAILABLE ON A PDP-12  
/COMPUTER. IT OPERATES IN BOTH 8 AND L MODE.  
/4K OF MEMORY IS REQUIRED,

/RIGHT SWITCH REGISTER OPTIONS:  
/SR00=1, INHIBIT ERROR HALT  
/SR01=1, INHIBIT ERROR PRINTOUT  
/SR02=1, SCOPE LOOP ON FAILING ROUTINE  
/SR03=1, SCOPE LOOP ON NON-FAILING ROUTINE  
/SR05=1, INHIBIT BELL  
/SR06=1, INHIBIT PASS COUNTER PRINTOUT

/NORMAL SWITCH SETTING IS RSW=0000.

/PROGRAM CONTROL IS HANDLED BY A MONITOR RESIDENT IN BANK 0,  
/LOCATIONS 5000 TO 5177. MOST ROUTINES VISIT THE MONITOR 4096 TIMES  
/AT THE COMPLETION OF A TEST, AN ERROR WILL CAUSE THE  
/PROGRAM TO TYPE OUT THE ERROR MESSAGE AND HALT. THE  
/HALT IS AT LOCATION 5033. THE HALTS IN THE PROGRAM  
/BLOCKS ARE NOT, REPEAT NOT, EXECUTED. THEY ARE  
/THERE FOR MANUAL PROGRAM CONTROL ONLY.

/I/O PRESET TO PMODE, START 20

*These switches do more than these notes indicate; I don't  
know the notes. Can I find the original program?*

```
/PDP-12 INSTRUCTION DEFINITIONS
/L MODE MEMORY REFERENCE
LOF=0640 /LOAD DATA FIELD 0-37
LIF=0600 /LOAD INSTRUCTION FIELD 0-37
DJR=0006 /DISABLE JUMP RETURN
/MODE CHANGE
PDP=0002 /SWITCH TO P MODE
LINC=6141 /SWITCH TO L MODE
/L MODE PROGRAMMING INSTRUCTIONS
LJMP=6000 /JMP
CLR=0011
AZE=0450
ADD=2000
IOB=0500
LNOP=0016 /NOP
ROR=0300
LSKP=0456
ROL=0240
BSE=1600
BCL=1540
SET=0060 /((REALLY SET I))
STC=4000 /USED AS A SWITCH CHECK
SRO=1500
LDA=1000
STA=1040
XSK=0220 /((REALLY XSK I))
/DATA MATRIX SWITCHES
EXITA=7777
EXITB=4444 /SPECIAL RESTART SWITCH
EXIT=0000
```

```

0000 0000 /P MODE INTERRUPT HANDLER
0001 7320 *0000
0002 6234 PINTR, 0000 /INTERRUPT RETURN STORAGE (ALSO LINC JUMP SAVE)
0003 3125 CLA CLL CML /SET LINK, CLEAR AC
0004 1124 RIB /READ SF
0005 7650 DCA PREG /SAVE IT
0006 5535 TAD PPOINT /GET SWITCH
0007 3124 SNA CLA /SET?
0010 6244 JMP I RETURN /NO, RETURN THROUGH PRESET LINKUP
0011 2000 DCA PPOINT /CLEAR SWITCH
0012 5400 ISZ 0 /RESTORE MEMORY
          JMP I PINTR /INCREMENT RETURN OVER JMP
          /BACK TO MAINLINE VIA INTERRUPT RETURN LINKUP
0013 0000 /AUTO-INDEX REGISTERS
0014 0000 LREG1, 0000 /DATA POINTER
0015 0000 PINT, 0000 /MESSAGE POINTER
0016 0000 AUTO11, 0000
0017 0000 AUTO12, 0000
          COUNT, 0000
          /
          /CROSS-PAGE REFERENCE TAGS AND CONSTANTS
          /
0020 5176 *0020 JMP 176 /MINOR START
          /
          /LOC 0021-0037 RESERVED FOR SUBROUTINES
          /

```



/MORE TAGS AND CONSTANTS

0066 5337 BELL, BELLS  
0067 5020 ERROR, ERRORS  
0070 0020 K0020, 0020  
0071 0040 K0040, 0040  
0072 0077 K0077, 0077  
0073 0100 K0100, 0100  
0074 0177 K0177, 0177  
0075 0207 K0207, 0207  
0076 0400 K0400, 0400  
0077 1026 K1026, 1026  
0100 1777 K1777, 1777  
0101 2000 K2000, 2000  
0102 2021 K2021, 2021  
0103 5252 K5252, 5252  
0104 6020 K6020, 6020  
0105 7774 K7774, 7774  
0106 0640 KLDF, LDF  
0107 0600 KLIF, LIF  
0110 6000 KLJMP, LJMP  
0111 0003 LMASK, 0003  
0112 0000 LSTERR, 0000  
0113 5000 NERROR, NERRORS  
0114 0354 PNIA, LOCA  
0115 0434 PNTC, LOCC  
0116 0400 PNTCA, LOCCA  
0117 0477 PNTE, LOCE  
0120 0757 PNTF, LOCF  
0121 1115 PNTJ, LOCJ  
0122 1161 PNT0, LOCO  
0123 5331 PNTP, LOCP  
0124 0000 PPOINT, 0000  
0125 0000 PREG, 0000  
0126 5200 RANDOM, RANDY  
0127 0000 REGA, 0000  
0130 0000 REGB, 0000  
0131 0000 REGC, 0000  
0132 0000 REGD, 0000  
0133 0000 REGE, 0000  
0134 5261 RELOC, RELOC  
0135 0000 RETURN, 0000  
0136 5253 SETFLG, FLAG  
0137 0000 SPACE, 0000  
0140 5306 TSTINT, INTTST  
0141 5244 TYPE, TYP0UT  
0142 1124 TST29N, TST29  
0143 1001 TRY22N, TRY22A  
0144 1163 TEST, TESTN

/CROSS PAGE TO BELL RINGER  
/CROSS PAGE TO ERROR MONITOR

/USED IN RELOCATION OF TESTS

/LMODE LDF  
/LMODE LIF  
/LMODE JMP  
/LIF/LDF MASK  
/LAST ERROR POINTER  
/CROSS PAGE TO NON-ERROR MONITOR  
/INTERRUPT RETURN TEST 07  
/INTERRUPT RETURN TEST 09  
/INTERRUPT RETURN  
/INTERRUPT RETURN TEST 11  
/INTERRUPT RETURN TEST 21  
/MESSAGE POINTER  
/PMODE SWITCH,  
/HOLDS SF  
/CROSS PAGE TO RANDOM GENERATOR  
/DA  
/DA  
/DA  
/DA  
/DA  
/CROSS PAGE TO RELOCATOR SUBR  
/PMODE INTERRUPT RETURN IF SWITCH=0  
/CROSS PAGE TO FLAG SET ROUTINE  
/DA I/O BUFFER  
/CROSS PAGE TO TYPEOUT SUBR

```

/
/TO HERE FROM MINOR START
/
*176
0176 7410 SKP /DON'T RING ON STARTUP, INITIALIZE TEST
0177 4466 JMS I BELL /GO RING BELL, RETURN TO TST03
/
/MAJOR START P MODE; INITIALIZATION ROUTINE
/
0200 7300 START, CLA CLL /CLEAR AC
0201 3127 DCA REGA /CLEAR LOOP COUNTER
0202 3112 DCA LSTERR /CLEAR OLD ERROR
0203 3017 DCA COUNT /CLEAR PASS COUNTER
0204 4540 JMS I TSTINT /TEST FOR NO INTERRUPT

```





```

0235 4526 TST24, JMS I RANDOM
0236 0111 AND LMASK
0237 3130 DCA REGB
0240 1130 TAD REGB
0241 1106 TAD KLDF
0242 3244 DCA .+2
0243 6141 LINC
0244 0000
0245 0500 IOB
0246 6214 RDF
0247 0002 PDP
0250 7110 RAR CLL
0251 3131 DCA REGC
0252 1131 TAD REGC
0253 7041 CIA
0254 1130 TAD REGB
0255 6201 CDF
0256 7650 SNA CLA
0257 4513 JMS I NERROR
0260 4467 JMS I ERROR
0261 5402 TST04M
0262 7402 HLT
0263 7610 SKP CLA
0264 0235 TST04

```

```

/ LMODE
/ CAN THE DATA FIELD REGISTER BE LOADED WITH RANDOM NUMBERS
/
/ GET RANDOM NUMBER
/ SAVE BITS 10-11
/ SAVE FOR OBSERVATION
/ FETCH IT
/ ADD LF
/ PLACE IN ROUTINE
/ GO TO LINC MODE
/ EXECUTE LDF
/ PREPARE TO GET DATA FIELD
/ GET DATA FIELD
/ BACK TO PMODE
/ JUSTIFY RIGHT TO AGREE WITH REGB
/ SAVE FOR TYPING
/ FETCH IT
/ 2'S COMPLEMENT
/ COMPARE
/ RESTORE DATA FIELD
/ INCORRECT IF NOT ZERO
/ CHECK WITH MONITOR
/ LDF FAILED
/ MESSAGE POINTER
/ ERROR HALT
/ GO TO NEXT TEST
/ SCOPE LOOP; ISZ LOOP

```

```

0265 4526 /L MODE
0266 0111 /GATE SHAKER TEST
0267 3130 /
0270 1130 TST06, JMS I RANDOM
0271 1106 AND LMASK
0272 3324 DCA REGB
0273 6141 TAD TAD REGB
0274 0640 DCA KLDF
0275 0677 NOW2
0276 0660 00
0277 0650 37
0300 0644 LDF 20
0301 0642 LDF 10
0302 0641 LDF 04
0303 0665 LDF 02
0304 0652 LDF 01
0305 0647 LDF 25
0306 0670 LDF 12
0307 0640 LDF 07
0310 0641 LDF 30
0311 0642 LDF 00
0312 0643 LDF 01
0313 0644 LDF 02
0314 0645 LDF 03
0315 0646 LDF 04
0316 0646 LDF 05
0317 0647 LDF 06
0320 0650 LDF 06
0321 0657 LDF 07
0322 0667 LDF 10
0323 0677 LDF 17
0324 0000 LDF 27
0325 0500 LDF 37
0326 6214 IOB
0327 0002 RDF
0330 7110 PDP
0331 3131 RAR CLL
0332 1131 DCA REGC
0333 7041 TAD REGC
0334 1130 CIA
0335 6201 TAD REGB
0336 7650 CDF 00
0337 4513 SNA CLA
0340 4467 JMS I NERROR
0341 5434 JMS I ERROR
0342 7402 TST06M
0343 7610 HLT
0344 0265 SKP CLA
TST06

```

```

/GET A RANDOM NUMBER
/SAVE BITS 10-11
/SAVE FOR OBSERVATION
/FETCH IT
/ADD LDF
/STORE FOR EXECUTION
/GO TO LINC MODE
/TRY SOME DATA FIELD
/NOISEMAKERS

```

```

/EXECUTE ACTUAL LDF
/PREPARE TO GET DATA FIELD
/GET DATA FIELD
/GO TO PMODE
/JUSTIFY WITH REGB
/SAVE FOR TYPING
/FETCH IT
/2'S COMPLEMENT
/COMPARE
/RESTORE DATA FIELD
/INCORRECT IF NOT ZERO
/CHECK WITH MONITOR
/PROBLEMS WITH NOISY DATA FIELD
/MESSAGE POINTER
/ERROR HALT
/GO TO NEXT TEST
/SCOPE LOOP; ISE LOOP

```



/LMODE  
/CHECK INTERRUPT FACILITY

0364	6041	TST9A,	JMS I	TSF	/CHECK FOR FLAG
0365	4536		TAD	SETFLG	/NOT UP; GO SET IT
0366	1116		AND	PNTCA	/GET RETURN ADDRESS
0367	0100		TAD	K1777	/10 BIT ADDRESS
0370	1110		DCA	KLJMP	/ADD LINC JUMP
0371	3050		CLL	LSET	/STORE FOR EXECUTION
0372	3065		LINC	LPOINT	/ZERO THE LMODE SWITCH
0373	7120		ION		/SET LINK
0374	6141		LNOP		/GO TO LINC MODE
0375	0500		IOB		/PREPARE TO EXECUTE IOT
0376	6001		IOF		/ENABLE INTERRUPTS
0377	0016		PDP		/WAIT
0400	0500	LOCCA,	SNL		/PREPARE TO EXECUTE IOT
0401	6002		JMS I	NERROR	/DISABLE INTERRUPTS
0402	0002		JMS I	ERROR	/BACK TO PMODE
0403	7420		TST9AM		/CHECK LINK, INCORRECT IF SET
0404	4513		HLT		/CHECK WITH MONITOR
0405	4467		SKP CLA		/INTERRUPT FAILED
0406	5507		TST9A		/MESSAGE POINTER
0407	7402				/ERROR HALT
0410	7610				/GO TO NEXT TEST
0411	0364				/ISZ LOOP; SCOPE LOOP

```

/ LMODE
/ CHECK RIB
/
0412 6041  TSTC9, TSF          SETFLG
0413 4536  TAD          JMS I     PNIC
0414 1115  TAD          AND      K1777
0415 0100  TAD          AND      KLJMP
0416 1110  TAD          DCA     LSET
0417 3050  TAD          JMS I     RANDOM
0420 4526  TAD          AND      LMASK
0421 0111  TAD          DCA     REGB
0422 3130  TAD          TAD     REGB
0423 1130  TAD          TAD     KLDF
0424 1106  TAD          DCA     '+3
0425 3230  TAD          CLL CML
0426 7120  TAD          LINC
0427 6141  TAD          0000
0430 0000  TAD          IOB
0431 0500  TAD          ION
0432 6001  TAD          LNOP
0433 0016  TAD          IOB
0434 0500  TAD          IOF
0435 6002  TAD          IOB
0436 0500  TAD          RIB
0437 6234  TAD          ROL
0440 0242  TAD          PDP
0441 0002  TAD          AND
0442 0111  TAD          DCA
0443 3131  TAD          TAD
0444 1131  TAD          CIA
0445 7041  TAD          SNA CLA
0446 1130  TAD          JMS I     NERRR
0447 7650  TAD          JMS I     ERROR
0450 4513  TAD          TST09M
0451 4467  TAD          HLT
0452 5530  TAD          SKP CLA
0453 7402  TAD          TST09
0454 7610  TAD
0455 0412  TAD

```

```

/ CHECK FOR FLAG
/ NOT UP; GO SET IT
/ GET RETURN ADDRESS
/ 10 BIT ADDRESS
/ ADD LINC JUMP
/ STORE IN RETURN ADDRESS
/ GET RANDOM NUMBER
/ SAVE BITS 10-11
/ SAVE FOR COMPARISON
/ FETCH IT
/ ADD LDF
/ STORE FOR EXECUTION
/ SET LINK
/ GO TO LINC MODE
/ EXECUTE LDF
/ PREPARE FOR IOT
/ ENABLE INTERRUPT
/ WAIT
/ PREPARE FOR IOT
/ DISABLE INTERRUPT
/ PREPARE FOR IOT
/ READ INTERRUPT BUFFER
/ JUSTIFY WITH REGB
/ BACK TO PMODE
/ SAVE BITS 10-11
/ SAVE FOR TYPING
/ FETCH IT
/ 2'S COMPLEMENT
/ COMPARE
/ INCORRECT IF NOT ZERO
/ CHECK WITH MONITOR
/ LMODE RIB FAILED
/ MESSAGE POINTER
/ ERROR HALT
/ GO TO NEXT TEST
/ SCOPE LOOP: ISZ LOOP

```

```

/LOMODE
/DOES THE DATA FIELD SET TO ZERO FOR AN INTERRUPT
/
TST11, TSF JMS I SETFLG /CHECK FLAG
0456 6041 TAD PNTI /NOT UP; GO SET IT
0457 4536 AND K1777 /GET RETURN ADDRESS
0460 1117 TAD KLJMP /10 BIT ADDRESS
0461 0100 DCA I LSET /ADD LINC MODE JMP
0462 1110 JMS I RANDOM /STORE IT
0463 3050 AND LMASK /GET RANDOM NUMBER
0464 4526 DCA REGB /SAVE BITS 10-11
0465 0111 TAD REGB /STORE FOR TYPING
0466 3130 TAD KLDF /FETCH IT
0467 1130 TAD *+2 /ADD LDF
0470 1106 DCA LINC /STORE FOR EXECUTION
0471 3273 LINC /GO TO LINC MODE
0472 6141 0000 /EXECUTE LDF
0473 0000 IOB /PREPARE FOR IOT
0474 0500 ION /ENABLE INTERRUPT
0475 6001 LNOP /WAIT
0476 0016 IOB /PREPARE FOR IOT
0477 0500 IOB /DISABLE INTERRUPT
LOCE, IOF /PREPARE FOR IOT
0500 6002 IOB /READ INTERRUPT BUFFER
0501 0500 RIB /JUSTIFY WITH REGB
0502 6234 ROL 2 /SAVE FOR TYPING
0503 0242 STC /PREPARE FOR IOT
0504 4131 IOB /READ DATA FIELD
0505 0500 RDF /BACK TO PMODE
0506 6214 PDP /JUSTIFY WITH REGB
0507 0002 RAR CLL REGD /SAVE FOR TYPING
0510 7110 DCA REGD /FETCH IT
0511 3132 TAD SNA CLA /INCORRECT IF NOT ZERO
0512 1132 TAD JMS I NERROR /CHECK WITH MONITOR
0513 7650 JMS I ERROR /DATA FIELD FAILED TO ZERO ON INTERRUPT
0514 4513 TST11M /MESSAGE POINTER
0515 4467 HLT /ERROR HALT
0516 5562 SKP CLA /GO TO NEXT TEST
0517 7402 TST11 /SCOPE LOOP; ISZ LOOP
0520 7610
0521 0456

```

```

/LMODE
/DOES STA-LDA WORK FOR ALL DATA FIELDS
/
TST13, CLA CLL /CLEAR AC
0522 7300 TAD REGA /GET CURRENT BANK
0523 1127 AND LMASK /SAVE BITS 10-11
0524 0111 DCA REGB /SAVE FOR OBSERVATION
0525 3130 TAD REGB /GET IT
0526 1130 TAD KLDF /ADD LDF
0527 1106 DCA EXC13 /STORE FOR EXECUTION
0530 3335 TAD K2021 /GET ADDRESS
0531 1102 DCA LREG1 /STORE FOR INDIRECT ACCESS
0532 3013 TAD K5252 /GET CONSTANT
0533 1103 LINC /GO TO LMODE
0534 6141 0000 /EXECUTE LDF
0535 0000 LREG1 /STORE INDIRECT TO DF
0536 1053 LDA LREG1 /FETCH NUMBER
0537 1013 LDF 0 /RESTORE DATA FIELD
0540 0640 PDP /TO PMODE
0541 0002 DCA REGC /SAVE FOR TYPING
0542 3131 TAD REGC /FETCH IT
0543 1131 CIA /2'S COMPLEMENT
0544 7041 TAD K5252 /COMPARE
0545 1103 SNA CLA /INCORRECT IF NOT ZERO
0546 7650 JMS I NERROR /CHECK WITH MONITOR
0547 4513 JMS I ERROR /STA OR LDA FAILED
0550 4467 TST13M /MESSAGE POINTER
0551 5626 HLT /ERROR HALT
0552 7402 SKP CLA /NEXT TEST
0553 7610 TST13 /SCOPE LOOP; ISZ LOOP
0554 0522

```



```

/TEST THE DJR FUNCTION FOR ALL COMBINATIONS
/LMODE
/DOES DJR NOT FUNCTION WHEN NOT SET?
/
TST14,   CLA CLL      /CLEAR AC
          TAD          /GET CONSTANT
          DCA          /SET 0
          LINC         /GO TO LINC MODE
          LJMP        .+1 /DO A LINC JUMP
          POP         /BACK TO P MODE
          TAD         /GET 0
          CIA         /2'S COMPLEMENT
          TAD          /ADD CONSTANT
          SZA CLA     /WAS LOCATION 0 CHANGED?
          JMS I       /YES; CHECK WITH MONITOR
          JMS I       /LINC JUMP SAVE RETURN FAILED
          TST14M     /MESSAGE POINTER
          HLT        /ERROR HALT
          SKP CLA    /TO NEXT TEST
          TST14     /SCOPE LOOP; ISZ LOOP
          0555      7300
          0556      1103
          0557      3000
          0560      6141
          0561      6562
          0562      0002
          0563      1000
          0564      7041
          0565      1103
          0566      7640
          0567      4513
          0570      4467
          0571      6040
          0572      7402
          0573      7610
          0574      0555

```

```

/LMODE
/DOES DJR FUNCTION WHEN IT'S SET?
/
TST15,   CLA CLL      /CLEAR AC
          TAD          /GET CONSTANT
          DCA          /SET 0
          LINC         /TO L MODE
          DJR         /DISABLE JUMP SAVE RETURN
          LJMP        .+1 /DO A LINC JUMP
          POP         /BACK TO PMODE
          TAD         /GET 0
          CIA         /2'S COMPLEMENT
          TAD          /COMPARE WITH CONSTANT
          SNA CLA     /DID DJR WORK?
          JMS I       /CHECK WITH MONITOR
          JMS I       /DJR FAILED
          TST15M     /MESSAGE POINTER
          HLT        /ERROR HALT
          SKP CLA    /TO NEXT TEST
          TST15     /SCOPE LOOP; ISZ LOOP
          0575      7300
          0576      1103
          0577      3000
          0600      6141
          0601      0006
          0602      6603
          0603      0002
          0604      1000
          0605      7041
          0606      1103
          0607      7650
          0610      4513
          0611      4467
          0612      6075
          0613      7402
          0614      7610
          0615      0575

```

```

0616 7300 /LMODE
0617 1103 /DOES A LINC JUMP CLEAR DJR?
0620 3000 /
0621 6141 TST16, CLA CLL /CLEAR AC
0622 0006 TAD K5252 /GET CONSTANT
0623 6624 DCA 0 /SET 0
0624 6625 DJR ,+1 /TO LMODE
0625 0002 LJMPL ,+1 /DISABLE JUMP SAVE RETURN
0626 1000 PDP 0 /DO A LINC JUMP
0627 7041 CIA K5252 /DO ANOTHER LINC JUMP
0630 1103 TAD /BACK TO PMODE
0631 7640 SZA CLA /GET 0
0632 4513 JMS I NERROR /2'S COMPLEMENT
0633 4467 JMS I ERROR /COMPARE WITH CONSTANT
0634 6123 TST16M /DID DJR CLEAR?
0635 7402 HLT /CHECK MONITOR
0636 7610 SKP CLA /DJR FAILED TO CLEAR
0637 0616 TST16 /MESSAGE POINTER
/ERROR HALT
/TO NEXT TEST
/SCOPE LOOP; ISZ LOOP

```

```

0640 7300 /PMODE
0641 1103 /DOES JUMP SAVE RETURN WORK IN ERROR FOR 8 MODE JUMPS?
0642 3000 /
0643 5244 TST17, CLA CLL /CLEAR AC
0644 1000 TAD K5252 /GET CONSTANT
0645 7041 DJR ,+1 /SET 0
0646 1103 PDP 0 /DO AN 8 MODE JUMP
0647 7650 CIA K5252 /2'S COMPLEMENT
0650 4513 SNA CLA NERROR /COMPARE WITH CONSTANT
0651 4467 JMS I ERROR /DID WE SAVE IN ERROR?
0652 6150 TST17M /CHECK MONITOR
0653 7402 HLT /JUMP SAVE RETURN OPERATED IN ERROR
0654 7610 SKP CLA /MESSAGE POINTER
0655 0640 TST17 /ERROR HALT
/TO NEXT TEST
/ISZ LOOP; SCOPE LOOP

```

/PMODE JUMP SAVE RETURN WORK IN ERROR FOR NON-JUMP COMMANDS?  
/DOES

0656	7300	TST18,	CLA CLL	/CLEAR AC
0657	1103	TAD	K5252	/GET CONSTANT
0660	3000	DCA	Ø	/SET Ø
0661	6002	IOF	Ø	/IOF LOOKS LIKE LINC JUMP
0662	1000	TAD	Ø	/GET Ø
0663	7041	CIA	Ø	/2'S COMPLEMENT
0664	1103	TAD	K5252	/COMPARE WITH CONSTANT
0665	7650	SNA CLA	Ø	/DID CELL Ø CHANGE?
0666	4513	JMS I	NERROR	/CHECK MONITOR
0667	4467	JMS I	ERROR	/IOF CHANGED CELL Ø
0670	6174	TST18M	Ø	/MESSAGE POINTER
0671	7402	HLT	Ø	/ERROR HALT
0672	7610	SKP CLA	Ø	/TO NEXT TEST
0673	0656	TST18	Ø	/SCOPE LOOP; ISZ LOOP

/LMODE JUMP SAVE RETURN WORK IN ERROR FOR NON-JUMP COMMANDS?  
/DOES

0674	7300	TST19,	CLA CLL	/CLEAR AC
0675	1103	TAD	K5252	/GET CONSTANT
0676	3000	DCA	Ø	/SET Ø
0677	6141	LINC	Ø	/GO TO LMODE
0700	0500	IOB	Ø	/PREPARE FOR IOT
0701	6002	IOF	Ø	/DISABLE INTERRUPTS
0702	0002	PDP	Ø	/BACK TO PMODE
0703	1000	TAD	Ø	/FETCH Ø
0704	7041	CIA	Ø	/2'S COMPLEMENT
0705	1103	TAD	K5252	/ADD CONSTANT
0706	7650	SNA CLA	Ø	/EQUAL?
0707	4513	JMS I	NERROR	/CHECK MONITOR
0710	4467	JMS I	ERROR	/IOB/IOF CAUSED LOC 0000 TO ALTER
0711	6220	TST19M	Ø	/MESSAGE POINTER
0712	7402	HLT	Ø	/ERROR HALT
0713	7610	SKP CLA	Ø	/TO NEXT TEST
0714	0674	TST19	Ø	/ISZ LOOP; SCOPE LOOP

```

0715 7300 /LMODE
0716 1103 /DOES DJR CLEAR IN ERROR WITH 8 MODE JUMP?
0717 3000 /
0720 6141 TST20, CLA CLL
0721 0006 TAD K5252
0722 0002 DCA 0
0723 5324 DJR LINC
0724 6141 PDP
0725 6726 JMP ,+1
0726 0002 LINC ,+1
0727 1000 LJMP 0
0730 7041 TAD 0
0731 1103 CIA K5252
0732 7650 TAD SNA CLA
0733 4513 JMS I NERROR
0734 4467 JMS I ERROR
0735 6244 TST20M
0736 7402 HLT
0737 7610 SKP CLA
0740 0715 TST20

```

```

/CLEAR AC
/GET CONSTANT
/SET 0
/TO LMODE
/DISABLE JUMP RETURN SAVE
/TO PMODE
/JUMP
/TO LMODE
/JUMP
/TO PMODE
/FETCH 0
/2'S COMPLEMENT
/ADD CONSTANT
/EQUAL?
/CHECK MONITOR
/8 MODE JUMP CLEARED DJR
/MESSAGE POINTER
/ERROR HALT
/TO NEXT TEST
/ISZ LOOP; SCOPE LOOP

```

```

0741 7300 /PMODE
0742 1120 /DOES DJR INHIBIT 8 MODE INTERRUPT SAVE IN ERROR?
0743 3135 /
0744 1103 TST21, CLA CLL
0745 3000 PNTF
0746 6041 RETURN
0747 4536 DCA
0750 6141 TAD
0751 0006 DJR
0752 0002 PDP
0753 6001 ION
0754 7000 NOP
0755 6002 IOF
0756 7410 SKP
0757 1000 TAD
0760 7041 CIA
0761 1103 TAD
0762 7640 SZA CLA
0763 4513 JMS I
0764 4467 JMS I
0765 6265 TST21M
0766 7402 HLT
0767 7610 SKP CLA
0770 0741 TST21

/GET RETURN POINTER TO LOCF
/SET UP INTERRUPT HANDLER
/GET CONSTANT
/STORE IN 0
/FLAG SET?
/NO, GO SET IT
/TO LMODE
/SET DJR
/TO PMODE
/ENABLE INTERRUPTS
/WAIT
/DISABLE INTERRUPTS
/IF NO INTERRUPT, THIS CAUSES ERROR
/GET 0
/2'S COMPLEMENT
/ADD CONSTANT
/EQUAL?
/CHECK MONITOR
/DJR INHIBITED 8 MODE INTERRUPT
/MESSAGE POINTER
/ERROR HALT
/TO NEXT TEST
/ISZ LOOP; SCOPE LOOP

```

/LMODE  
/NOW CHECK THE LMODE SF

0771	7300	CLA CLL	/CLEAR AC
0772	1127	TAD	/GET DATA WORD
0773	0111	AND	/SAVE BITS 10-11
0774	3131	DCA	/SAVE FOR OBSERVATION
0775	1131	TAD	/GET OF
0776	1106	TAD	/ADD KLDF
0777	3543	DCA I	/STORE FOR EXECUTION
1000	6141	LINC	
1001	0000	TRY22A, 0000	/LDF
1002	0600	LIF	/BACK TO IF 0, LOAD SF
1003	0500	IOB	/
1004	6234	RIB	/READ SAVE FIELD
1005	0242	ROL	/JUSTIFY
1006	0002	PDP	/TO PMODE
1007	3133	DCA	/SAVE FOR TYPEOUT
1010	1133	TAD	/GET IT
1011	7041	CIA	/2'S COMPLEMENT
1012	1131	TAD	/COMPARE
1013	7650	SNA CLA	/INCORRECT IF NOT ZERO
1014	4513	JMS I	/CHECK MONITOR
1015	4467	JMS I	/LIF FAILED TO LOAD SF
1016	6314	TST22M	/MESSAGE POINTER
1017	7402	HLT	/ERROR HALT
1020	7610	SKP CLA	/TO NEXT TEST
1021	0771	TST22	/SCOPE LOOP; ISZ LOOP.

```

1022 7300 /LMODE
1023 1127 /EXECUTE LIF N AND SEE IF WE REALLY GET THERE.
1024 0111 /
1025 3130 TST23, CLA CLL
1026 1130 TAD REGA
1027 1107 AND LMASK
1030 3234 DCA REGB
1031 4534 TAD REGB
1032 0007 TAD KLIF
1033 6141 DCA TRY23
1034 0000 JMS I RELOCR
1035 6024 0007
1036 0500 LINC
1037 6224 LJMP
1040 0600 IOB
1041 7042 RIF
1042 0002 LIF
1043 7010 PDP
1044 3131 RAR
1045 1131 DCA
1046 7041 TAD
1047 1130 CIA
1050 7650 TAD SNA CLA
1051 4513 JMS I NERROR
1052 4467 JMS I ERROR
1053 6357 TST23M
1054 7402 HLT
1055 7610 SKP CLA
1056 1022 TST23
1022 7300 /CLEAR AC
1023 1127 /GET DATA WORD
1024 0111 /SAVE BITS 10-11
1025 3130 /STORE FOR OBSERVATION
1026 1130 /GET IT
1027 1107 /ADD LIF
1030 3234 /STORE FOR EXECUTION
1031 4534 /GO RELOCATE THE NEXT
1032 0007 /7 LINES OF CODE
1033 6141 /TO LMODE
1034 0000 /EXECUTE LIF N
1035 6024 /(.+1 IN RELOCATED PROGRAM)
1036 0500 /
1037 6224 /READ INSTRUCTION FIELD
1040 0600 /BACK TO FIELD 0
1041 7042 /BACK TO TEST PROGRAM
1042 0002 /TO PMODE
1043 7010 /JUSTIFY
1044 3131 /SAVE IT
1045 1131 /GET IT
1046 7041 /2'S COMPLEMENT
1047 1130 /ADD TARGET
1050 7650 /COMPARE; INCORRECT IF NOT ZERO
1051 4513 /CHECK MONITOR
1052 4467 /LIF-JMP N FAILED TO FIND IF
1053 6357 /MESSAGE POINTER
1054 7402 /ERROR HALT
1055 7610 /TO NEXT TEST
1056 1022 /IS LOOP; SCOPE LOOP

```

```

/ LMODE
/ INTERRUPT INHIBIT TEST BANK 0 -BANK N- BANK 0
/
TST28, CLA CLL
1057 7300 /CLEAR AC
1060 3065 /CLEAR HANDLER SWITCH
1061 1121 /GET ERROR RETURN
1062 1110 /MAKE IT A LINC JUMP
1063 3050 /PLACE IT IN HANDLER
1064 1127 /GET DATA WORD
1065 0111 /MASK OUT TO BITS 10-11
1066 3130 /SAVE BANK
1067 1130 /FETCH IT
1070 1107 /MAKE IT A LIF N
1071 3277 /STORE FOR EXECUTION
1072 6041 /FLAG SET?
1073 4536 /NO, GO SET IT
1074 4534 /GO RELOCATE THE NEXT
1075 0017 /17 LINES OF CODE.
1076 6141 /TO LINC MODE
1077 0000 /EXECUTE LIF N
1100 0500 /
1101 6001 /ENABLE INTERRUPTS (SHOULD INHIBIT)
1102 6026 /TO EXTENDED MEMORY (.+1 IN RELOCATED SUBROUTINE)
1103 0016 /WAIT FOR INTERRUPT
1104 0600 /LOAD IB
1105 0500 /
1106 6001 /ENABLE INTERRUPT AGAIN
1107 6033 /BACK TO BANK 0
1110 0016 /WAIT FOR INTERRUPT
1111 0500 /
1112 6002 /DISABLE INTERRUPT
1113 0002 /BACK TO PMODE
1114 5542 /JUMP TO NEXT PORTION OF TEST
1115 0002 /BACK HERE IF INTERRUPT OCCURS
1116 6002 /DISABLE INTERRUPT
1117 4467 /LIF FAILED TO INHIBIT INTERRUPT
1120 6417 /MESSAGE POINTER
1121 7402 /ERROR HALT
1122 7610 /TO NEXT TEST
1123 1057 /ISZ LOOP; SCOPE LOOP

DCA DCA
TAD LPOINT
TAD PNTJ
TAD KLJMP
DCA LSET
TAD REGA
AND LMASK
DCA REGB
TAD REGB
TAD KLIF
DCA TRY28
TSF SETFLG
JMS I RELOCR
JMS I
0017
LINC
0000
IOB
ION
LJMP 26
LNOP 0
IOB 33
ION
LJMP
LNOP
IOB
PDP
JMP I TST29N
PDP
IOF
TST28M
HLT
SKP CLA
TST28

```



```

1124 7300 TST29, CLA CLL
1125 6141 LINC
1126 0500 IOB
1127 6234 RIB
1130 0303 ROR
1131 0202 POP
1132 0111 AND
1133 3131 DCA
1134 1131 TAD
1135 7041 CIA
1136 1130 TAD
1137 7690 SNA CLA
1140 4513 JMS I
1141 4467 JMS I
1142 6454 TST29M
1143 7402 HLT
1144 7610 SKP CLA
1145 1057 TST28

/TO LINC MODE
/READ SAVE FIELD

/JUSTIFY
/TO PMODE
/MASK OUT NOISE
/SAVE IT
/FETCH IT
/COMPLEMENT
/COMPARE WITH IF
/EQUAL?
/YES, CONTINUE WITH TST28
/NO, LIF FAILED TO LOAD SF
/MESSAGE POINTER
/ERROR HALT
/TO NEXT TEST
/ISZ LOOP, SCOPE LOOP

3
LMASK
REGC
REGC
REGB
NERROR
NERROR
ERROR
TST29M
HLT
SKP CLA
TST28

```



```

/ALERT OPERATOR OF PASS COMPLETION (INHIBIT IF RSW 06=01)
/
1146 7300 CLA CLL /CLEAR REGA
1147 3127 DCA REGA /INCREMENT COUNT
1150 2017 ISZ COUNT /DON'T SKIP
1151 7000 NOP /GET SWITCHES
1152 7604 LAS /PICK OUT BIT 06
1153 0071 AND K0040 /SET ?
1154 7640 SZA CLA /YES, INHIBIT AND RESTART
1155 5177 JMP 177 /GET POINTER TO TEXT
1156 1122 TAD PNT0 /CHEAT MONITOR
1157 3467 OCA I ERROR /GO TYPE MESSAGE
1160 5762 JMP I PASPNT /MESSAGE POINTER
1161 6514 LOCO, TST37M /LINKUP POINTER
1162 5050 PASPNT, ASCII

/TEST FOR CTL-C AND RETURN TO PS12 MONITOR IF STRUCK
/
1163 0000 TESTN, 0 /GET KEYBOARD
1164 6036 KRB /ADD IN MINUS 203
1165 1372 TAD M203 /CTL-C?
1166 7650 SNA CLA /YES, CALL MONITOR
1167 5773 JMP I K7605 /NO, GET SWITCHES
1170 7604 LAS /RETURN TO CALLING ROUTINE
1171 5763 JMP I TESTN
1172 7575 M203, -203
1173 7605 K7605, 7605

```

/LOC 2020-2040 RESERVED

\*2020

2020	0016	LNOP
2021	0016	LNOP
2022	0016	LNOP
2023	0016	LNOP
2024	0016	LNOP
2025	0016	LNOP
2026	0016	LNOP
2027	0016	LNOP
2030	0016	LNOP
2031	0016	LNOP
2032	0016	LNOP
2033	0016	LNOP
2034	0016	LNOP
2035	0016	LNOP
2036	0016	LNOP
2037	0016	LNOP

\*4020

/LOC 4020 - 4040 RESERVED

4020	0016	LNOP
4021	0016	LNOP
4022	0016	LNOP
4023	0016	LNOP
4024	0016	LNOP
4025	0016	LNOP
4026	0016	LNOP
4027	0016	LNOP
4030	0016	LNOP
4031	0016	LNOP
4032	0016	LNOP
4033	0016	LNOP
4034	0016	LNOP
4035	0016	LNOP
4036	0016	LNOP
4037	0016	LNOP

```

5000 0000 /NON ERROR MONITOR DETERMINES IF OPERATOR WANTS TO LOOP ON NON FAILING TEST
5001 7307 /NON ERRORS, 0
5002 1200 CLA CLL IAC RTL
5003 3200 TAD ERRORS
5004 1600 DCA ERRORS
5005 3220 TAD I ERRORS
5006 2127 DCA ERRORS
5007 5620 ISZ REGA
5010 4544 JMP I ERRORS
5011 0076 JMS I TEST
5012 7640 AND K0400
5013 5620 SZA CLA ERRORS
5014 7040 JMP I ERRORS
5015 1200 CMA ERRORS
5016 3200 TAD ERRORS
5017 5600 DCA ERRORS
5017 JMP I ERRORS

/ERROR PROCESSOR, SCOPE LOOP, HALT, PRINT
5020 0000 /NON ERRORS, 0
5021 4544 JMS I TEST
5022 7004 RAL CLA
5023 7700 SMA CLA
5024 5250 JMP ASCII
5025 1220 TAD ERRORS
5026 7041 ASCRXT, TAD
5027 3112 DCA LSTERR
5030 2220 ISZ ERRORS
5031 7604 LAS
5032 7700 SMA CLA
5033 7402 HLT
5034 2220 ISZ ERRORS
5035 2220 ISZ ERRORS
5036 1620 TAD I ERRORS
5037 3200 DCA ERRORS
5040 7604 LAS
5041 7006 RTL
5042 7710 SPA CLA
5043 5600 JMP I ERRORS
5044 7040 CMA ERRORS
5045 1220 TAD ERRORS
5046 3220 DCA ERRORS
5047 5620 JMP I ERRORS

/RETURN ADDRESS
/SET AC = 4
/GET RETURN ADDRESS
/RETURN ADDRESS +4
/GET SCOPE LOOP ADDRESS
/STORE IT
/UPDATE DATA
/LOOP BACK TO TEST
/CHECK FOR CTL-C AND RETURN WITH SWITCHES
/SAVE SR3
/TEST AND CLEAR
/ LOOPING
/SET AC=-1
/ADD NERROS
/STORE IN NERROS
/JUMP INDIRECT LOOP

/RETURN ADDRESS STORAGE
/CHECK FOR CTL-C AND RETURN WITH SWITCHES
/MOVE SR1 INTO AC00
/IS IT SET
/NO TYPE A MESSAGE
/GET CURRENT ERROR ADDRESS
/INVERT IT
/STORE IN LAST ERROR
/YES INDEX ESCAPE
/READ SWITCHES
/IS SR0 SET?
/NO, ERROR HALT
/YES INDEX ESCAPE TO JUMP OUT
/INDEX ERRORS TO SCOPE MODE
/GET SCOPE ADDRESS
/STORE IN TYPE
/READ SWITCHES
/MOVE SR02 TO AC0
/IS SCOPE MODE SELECTED
/YES CONTINUE IN SCOPE LOOP
/NO SET AC=7777
/SUBTRACT ONE FROM ERRORS
/STORE SELECTED ADDRESS
/EXIT TO NEXT TEST

```

5050	7240	ASCII,	CLA CMA	ERRORS	/SET C(AC)=-1
5051	1620	TAD I		PINT	/GET MESSAGE ADDRESS STORAGE
5052	3014	DCA		ERRORS	/STORE IT IN AUTO INDEX REGISTER
5053	1220	TAD		LSTERR	/GET RETURN ADDRESS
5054	1112	TAD			/SUBTRACT LAST ERROR ADDRESS
5055	7650	SNA CLA			/TEST
5056	5362	JMP	DATYP		/SAME GO TYPE DATA
5057	1414	TAD I	PINT		/GET FIRST CHARACTER
5060	3200	DCA	NERROS		/SAVE IT
5061	1200	TAD	NERROS		/GET IT
5062	7450	SNA			/TEST IT
5063	5225	JMP	ASCRXT		/NUMBER=EXIT
5064	7040	CMA			/INVERT IT
5065	7450	SNA			/NUMBER=EXITA
5066	5314	JMP	DATUM		/TYPE OUT DATA ROUTINE
5067	7040	CMA			/CHANGE IT BACK
5070	7112	RTR CLL			/SWAP AC TO THE RIGHT
5071	7012	RTR			/MOVE
5072	7012	RTR			/MOVE
5073	4277	JMS	TYPECH		/TYPE IT
5074	1200	TAD	NERROS		/GET IT AGAIN
5075	4277	JMS	TYPECH		/TYPEIT
5076	5257	JMP	ASCII*7		/MUST BE MORE WORDS THAT NEED TYPING
5077	0000	TYPECH, 0			
5100	0072	AND	K0077		/SAVE SIGNIFICANT PART
5101	3137	DCA	SPACE		/STORE WORD
5102	1137	TAD	SPACE		/FETCH IT
5103	7650	SNA CLA			/TEST FOR 00 CRLF CODE
5104	4353	JMS	CRLF		/YES IT WAS
5105	1137	TAD	SPACE		/NO TYPE IT
5106	1377	TAD	M40		/SUBTRACT 40
5107	7510	SPA			/TEST POLARITY
5110	1073	TAD	K0100		/ADD 340
5111	1376	TAD	K240		/ADD 240
5112	4541	JMS I	TYPE		/TYPE
5113	5677	JMP I	TYPECH		/EXIT

5114	1414	DATUM,	TAD I	PINT	/GET ADDRESS OF REGISTER
5115	3200		DCA	NERROS	/STORE IN TEMP
5116	1200		TAD	NERROS	/GET TEMP
5117	7652		SNA CLA		/TEST FOR EXIT
5120	5225		JMP	ASCRXT	/EQUALS 0000 EXIT
5121	1200		TAD	NERROS	/GET TEMP
5122	1373		TAD	M4444	/ADD CONSTANT
5123	7650		SNA CLA		/TEST FOR RESTART
5124	4466		JMS I	BELL	/IT'S THERE; RESTART
5125	1600		TAD I	NERROS	/GET DATA
5126	4332		JMS	OCTYP	/TYPE IT
5127	1376		TAD	K240	/SPACE
5130	4541		JMS I	TYPE	/TYPE IT
5131	5314		JMP	DATUM	/TYPE NUMERIC DATA
5132	0000	OCTYP,	0		/RETURN ADDRESS STORAGE
5133	3277		DCA	TYPECH	/STORE DATA TO BE PRINTED
5134	1105		TAD	K7774	/SET UP TALLY
5135	3137		DCA	SPACE	/SET IT

5136	1077	HERE,	TAD	K1026	/GET FLAG NUMBER
5137	3353	REDO,	DCA	CRLF	/STORE
5140	1277		TAD	TYPECH	/GET DATA BACK
5141	7004		RAL		/JUSTIFY
5142	3277		DCA	TYPECH	/SAVE IT
5143	1353		TAD	CRLF	/GET CRLF CODE
5144	7004		RAL		/JUSTIFY
5145	7420		SNL		/FINISHED?
5146	5337		JMP	REDO	/NO, CONTINUE
5147	4541		JMS I	TYPE	/NOW TYPE IT
5150	2137		ISZ	SPACE	/FINISHED?
5151	5336		JMP	HERE	/NO, CONTINUE
5152	5732		JMP I	OCTYP	/EXIT
5153	0000	CRLF,	Ø		/RETURN ADDRESS STORAGE
5154	1374		TAD	K0215	/GET CR
5155	4541		JMS I	TYPE	/TYPE IT
5156	1375		TAD	K0212	/GET LF
5157	4541		JMS I	TYPE	/TYPE IT
5160	1074		TAD	K0177	/SET TO RUBOUT
5161	5753		JMP I	CRLF	/EXIT
5162	1414	DATYP,	TAD I	PINT	/GET A TERM OFF OF TYPE LIST
5163	7450		SNA		/END OF LIST?
5164	5225		JMP	ASCRXT	/YES EXIT
5165	7040		CMA		/INVERT
5166	7640		SZA CLA		/BEGINNING OF DATA
5167	5362		JMP	DATYP	/NO
5170	4353		JMS	CRLF	/YES OK RETURN THE TTY CARRIAGE AND LINE FEED
5171	7300		CLA CLL		/CLEAR AC AND LINK
5172	5314		JMP	DATUM	/GO TYPE THE DATA
5173	3334	M4444,	-4444		/SWITCH CHECK
5174	0215	K0215,	0215		
5175	0212	K0212,	0212		
5176	0240	K240,	0240		
5177	7740	M40,	-40		

\*5200 RANDY, 0 /NEW PAGE /RANDOM NUMBER GENERATOR

```

5200 0000 RNA
5201 1240 TAD
5202 1241 TAD
5203 1103 TAD
5204 3243 DCA
5205 1243 TAD
5206 1242 TAD
5207 3240 DCA
5210 7004 RAL
5211 1240 TAD
5212 1241 TAD
5213 1103 TAD
5214 3243 DCA
5215 1243 TAD
5216 1242 TAD
5217 3241 DCA
5220 7004 RAL
5221 1240 TAD
5222 1103 TAD
5223 3243 DCA
5224 1243 TAD
5225 1241 TAD
5226 1242 TAD
5227 3242 DCA
5230 7004 RAL
5231 1240 TAD
5232 3240 DCA
5233 1241 TAD
5234 1103 TAD
5235 3243 DCA
5236 1243 TAD
5237 5600 JMP I
5240 7601 RNA
5241 3542 RNB
5242 3755 RNC
5243 0016 RND
5244 0000 TYP0UT, 0
5245 6046 TLS
5246 6041 TSF
5247 5246 JMP
5250 6042 TCF
5251 7200 CLA
5252 5644 JMP I

/AC TO PRINTER
/FLAG SET?
/NOT UP; WAIT
/NOW CLEAR IT
/CLEAR AC
/INDIRECT RETURN

/TELEPRINTER FLAG SET ROUTINE
FLAG, 0000
5253 0000 CLA
5254 7200 TLF
5255 6046 TSF
5256 6041 JMP
5257 5256 JMP I
5260 5653 FLAG

/CLEAR AC
/BUMP PRINTER
/WAIT 100 MS
/INDIRECT RETURN.

```



```

/PROGRAM RELOCATOR
/CALL: RELOC; LENGTH. --"BANK" MUST BE IN REGB
/
5261 0000 RELOC, 0000 /CONTAINS CALLING LOCATION +1
5262 7302 CLA CLL /CLEAR AC
5263 1261 TAD RELOC /GET ADDRESS
5264 3131 DCA REGC /SAVE ADDRESS
5265 1531 TAD I REGC /LENGTH
5266 7040 CMA /COMPLEMENT
5267 3132 DCA REGD /SAVE IT
5270 1261 TAD RELOC /GET ADDRESS
5271 3015 DCA AUTO11 /SAVE AS PICK-UP POINTER
5272 1130 TAD REGB /GET BANK
5273 7012 RTR /JUSTIFY ADDRESS
5274 7010 RAR /JUSTIFY
5275 1070 TAD K0020 /ASSEMBLE ADDRESS
5276 3016 DCA AUTO12 /SAVE ADDRESS
5277 2261 ISZ RELOC /MOVE RETURN POINTER
5300 2132 ISZ REGD /CHECK IF DONE
INCREL, JMP PICKUP /NOT DONE; MOVE A WORD
5301 5303 JMP I RELOC /RETURN
5302 5661 JMP I AUTO11 /GET WORD
5303 1415 PICKUP, TAD I AUTO12 /DEPOSIT WORD
5304 3416 DCA I /CHECK BACK
5305 5300 JMP INCREL

```

```

5306 0002 /PMODE-LMODE
5307 7340 /INTERRUPT TEST: DO WE HAVE A SPURIOUS INTERRUPT ON-LINE?
5310 3127 /
5311 6041 INTTST, 0002
5312 4536 CLA CLL CMA
5313 6141 DCA REGA
5314 1020 TSF
5315 0020 JMS I SETFLG
5316 0004 LDA 20
5317 0002 PDP
5320 5321 ,+1
5321 5322 ,+1
5322 7300 CLA CLL
5323 1123 TAD
5324 3135 DCA
5325 6001 ION
5326 7000 VOP
5327 6002 IOF
5330 4513 JMS I NERROR
5331 4467 JMS I ERROR
5332 6532 INTSTM
5333 7402 HLT
5334 7410 SKP
5335 5307 INTTST+1
5336 5706 JMP I INTTST

```

```

/FLAG SET?
/NOT UP; GO SET IT
/TO LMODE
/GET BIT 07
/I/O PRESET
/ESF
/TO PMODE
/CLEAR INHIBIT
/CLEAR INHIBIT
/ZERO AC, LINK
/GET POINTER
/SET UP RETURN
/ENABLE INTERRUPTS
/WAIT
/DISABLE INTERRUPTS
/NO INTERRUPT ON-LINE
/SPURIOUS INTERRUPT!
/MESSAGE POINTER
/ERROR HALT
/RETURN
/ISZ LOOP; SCOPE LOOP
/RETURN

```

```

,+1
,+1
PNTP
RETURN
NERROR
ERROR
INTSTM
HLT
SKP
INTTST+1
JMP I INTTST

```

LOOP,

/RING THE BELL

```

5337 0000 BELLS, 0000 /READ SWITCHES
5340 7404 OSR /SAVE SR05
5341 0073 AND /IS IT SET?
5342 7640 SZA CLA K0100 /YES, INHIBIT BELL
5343 5737 JMP I BELLS /GET BELL
5344 1075 TAD K0207 /GO RING IT
5345 4541 JMS I TYPE /RETURN
5346 5747 JMP I ,+1 /AVOID CLOBBERING PASS COUNTER
5347 0205 TST03

```

```

5350 0024 TST03M, 0024 /TST03
5351 2324 2324
5352 6063 6263
5353 0014 2414 /LDF OR RDF FAILED (LMODE)
5354 0406 0406 /SENT RCVD
5355 4017 4017
5356 2240 2240
5357 2204 2204
5360 0640 0640
5361 0601 0601
5362 1114 1114
5363 0504 0504
5364 4050 4050
5365 1415 1415
5366 1704 1704
5367 0551 0551
5370 4000 4000
5371 2305 2305
5372 1624 1624
5373 4022 4022
5374 0326 0326
5375 0400 0400
5376 7777 EXITA
5377 0130 REGC
5400 0131 REGC
5401 0000 EXIT

```

```

5402 0024 TST04M, 0024 /TST04
5403 2324 2324 /LDF OR RDF FAILED (LMODE)
5404 6064 6064 /SENT RCVD
5405 0014 0014
5406 0406 0406
5407 4017 4017
5410 2240 2240
5411 2204 2204
5412 0640 0640
5413 0601 0601
5414 1114 1114
5415 0504 0504
5416 4050 4050
5417 1415 1415
5420 1704 1704
5421 0551 0551
5422 0023 0023
5423 0516 0516
5424 2440 2440
5425 2203 2203
5426 2604 2604
5427 4000 4000
5430 7777 EXITA
5431 0130 REGC
5432 0131 REGC
5433 0000 EXIT

```

5434 0024 TST06M, 0024  
 5435 2324  
 5436 6066  
 5437 0014  
 5440 0406  
 5441 4017  
 5442 2240  
 5443 2204  
 5444 0640  
 5445 0601  
 5446 1114  
 5447 0504  
 5450 4050  
 5451 1415  
 5452 1704  
 5453 0551  
 5454 0023  
 5455 0516  
 5456 2440  
 5457 2203  
 5460 2604  
 5461 4000  
 5462 7777  
 5463 0130  
 5464 0131  
 5465 0000

/TST06  
 /LDF OR RDF FAILED (LMODE)  
 /SENT RCVD

5466 0024 TST07M, 0024  
 5467 2324  
 5470 6067  
 5471 0020  
 5472 1517  
 5473 0405  
 5474 4011  
 5475 1624  
 5476 0522  
 5477 2225  
 5500 2024  
 5501 4006  
 5502 0111  
 5503 1405  
 5504 0400  
 5505 7777  
 5506 0000

/TST07  
 /PMODE INTERRUPT FAILED

5507 0024 TST9AM, 0024  
 5510 2324  
 5511 7101  
 5512 0014  
 5513 1517  
 5514 0405  
 5515 4011  
 5516 1624  
 5517 0522

/TST9A  
 /LMODE INTERRUPT FAILED

5520 2225  
5521 2024  
5522 4006  
5523 0111  
5524 1405  
5525 0400  
5526 7777  
5527 0000  
EXITA  
EXIT

/TST09

5530 0024  
5531 2324  
5532 6071  
5533 0014  
5534 1517  
5535 0405  
5536 4014  
5537 1701  
5540 0440  
5541 2306  
5542 4017  
5543 2240  
5544 2211  
5545 0240  
5546 0601  
5547 1114  
5550 0504  
5551 0040  
5552 0406  
5553 4040  
5554 4023  
5555 0600  
5556 7777  
5557 0130  
5560 0131  
5561 0000  
TST09M, 0024  
2324  
6071  
0014  
1517  
0405  
4014  
1701  
0440  
2306  
4017  
2240  
2211  
0240  
0601  
1114  
0504  
0040  
0406  
4040  
4023  
0600  
EXITA  
REGC  
REGC  
EXIT

/LMODE LOAD SF OR RIB FAILED  
/ DF SF

/TST11

5562 0024  
5563 2324  
5564 6161  
5565 0014  
5566 1517  
5567 0405  
5570 4004  
5571 2640  
5572 0601  
5573 1114  
5574 0504  
5575 4024  
5576 1740  
5577 3205  
5600 2217  
5601 4017  
5602 1640  
5603 0116  
5604 4011  
TST11M, 0024  
2324  
6161  
0014  
1517  
0405  
4004  
0640  
0601  
1114  
0504  
4024  
1740  
3205  
2217  
4017  
1640  
0116  
4011

/LMODE OF FAILED TO ZERO ON AN INTERRUPT  
/SENT SF RCVD

5605 1624  
 5606 0522  
 5607 2225  
 5610 2024  
 5611 0023  
 5612 0516  
 5613 2440  
 5614 4023  
 5615 0640  
 5616 4022  
 5617 0326  
 5620 0400  
 5621 7777  
 5622 0130  
 5623 0131  
 5624 0132  
 5625 0000

/TST13

TST13M, 0024  
 5626 0024  
 5627 2324  
 5630 6163  
 5631 0014  
 5632 1517  
 5633 0405  
 5634 4023  
 5635 2401  
 5636 5514  
 5637 0401  
 5640 4006  
 5641 0111  
 5642 1405  
 5643 0400  
 5644 0201  
 5645 1613  
 5646 4003  
 5647 0514  
 5650 1440  
 5651 1017  
 5652 1704  
 5653 4002  
 5654 0104  
 5655 4000  
 5656 7777  
 5657 0130  
 5660 0102  
 5661 0103  
 5662 0131  
 5663 0000

/LMODE STA-LDA FAILED  
 /BANK CELL GOOD BAD

EXITA  
 REGB  
 REGC  
 REGD  
 EXIT

/LOCATIONS 6022-6040 RESERVED FOR SUBROUTINES

\*6020

6020	0016	LNOP
6021	0016	LNOP
6022	0016	LNOP
6023	0016	LNOP
6024	0016	LNOP
6025	0016	LNOP
6026	0016	LNOP
6027	0016	LNOP
6030	0016	LNOP
6031	0016	LNOP
6032	0016	LNOP
6033	0016	LNOP
6034	0016	LNOP
6035	0016	LNOP
6036	0016	LNOP
6037	0016	LNOP

/TST14

6040	0024	TST14M, 0024
6041	2324	2324
6042	6164	6164
6043	0014	0014
6044	1517	1517
6045	0405	0405
6046	4012	4012
6047	2515	2515
6050	2040	2040
6051	2301	2301
6052	2605	2605
6053	4022	4022
6054	0524	0524
6055	2522	2522
6056	1640	1640
6057	0601	0601
6060	1114	1114
6061	0504	0504
6062	4006	4006
6063	1722	1722
6064	4016	4016
6065	1722	1722
6066	1501	1501
6067	1440	1440
6072	1225	1225
6071	1520	1520
6072	4000	4000
6073	7777	EXITA
6074	0000	EXIT

/TST15

6075	0024	TST15M, 0024
6076	2324	2324
6077	6165	6165
6080	0004	0004
6081	1222	1222
6082	4006	4006

/DJR FAILED TO INHIBIT JUMP SAVE



6103 0111  
 6104 1405  
 6105 0440  
 6126 2417  
 6107 4011  
 6110 1610  
 6111 1102  
 6112 1124  
 6113 4012  
 6114 2515  
 6115 2040  
 6116 2301  
 6117 2605  
 6120 4000  
 6121 7777  
 6122 0000

TST16M, 0024  
 6123 0024  
 6124 2324  
 6125 6166  
 6126 0014  
 6127 1517  
 6130 0405  
 6131 4012  
 6132 1520  
 6133 4006  
 6134 0111  
 6135 1405  
 6136 0440  
 6137 2417  
 6140 4003  
 6141 1405  
 6142 0122  
 6143 4004  
 6144 1222  
 6145 4000  
 6146 7777  
 6147 0000

/TST16

/LMODE JUMP FAILED TO CLEAR DJR

TST17M, 0024  
 6150 0024  
 6151 2324  
 6152 6167  
 6153 0020  
 6154 1517  
 6155 0405  
 6156 4012  
 6157 2515  
 6160 2040  
 6161 0114  
 6162 2405  
 6163 2205  
 6164 2440  
 6165 2305  
 6166 1414  
 6167 4060

/TST17

/PMODE JUMP ALTERED CELL 0000

6170 6060  
6171 6000  
6172 7777  
6173 0000

TST18M, 0024 /TST18

6174 0024  
6175 2324  
6176 6170  
6177 0020  
6200 1517  
6201 0405  
6202 4011  
6203 1706  
6204 4001  
6205 1424  
6206 0522  
6207 0504  
6210 4003  
6211 0514  
6212 1440  
6213 6060  
6214 6060  
6215 4000  
6216 7777  
6217 0000

/PMODE IOF ALTERED CELL 0000

TST19M, 0024 /TST19

6220 0024  
6221 2324  
6222 6171  
6223 0014  
6224 1517  
6225 0405  
6226 4011  
6227 1706  
6230 4001  
6231 1424  
6232 0522  
6233 0504  
6234 4003  
6235 0514  
6236 1440  
6237 6060  
6240 6060  
6241 4000  
6242 7777  
6243 0000

/LMODE IOF ALTERED CELL 0000

TST20M, 0024 /TST20

6244 0024  
6245 2324  
6246 6260  
6247 0020  
6250 1517  
6251 0405  
6252 4012  
6253 2515

/PMODE JUMP CLEARED DJR

6254 2040  
 6255 0314  
 6256 0501  
 6257 2205  
 6260 0440  
 6261 0412  
 6262 2200  
 6263 7777  
 6264 0000  
 EXITA  
 EXIT

TST21M, 0024  
 6265 0024  
 6266 2324  
 6267 6261  
 6270 0004  
 6271 1222  
 6272 4011  
 6273 1610  
 6274 1102  
 6275 1124  
 6276 0504  
 6277 4020  
 6300 1517  
 6301 0405  
 6302 4011  
 6303 1624  
 6304 0522  
 6305 2225  
 6306 2024  
 6307 4023  
 6310 0126  
 6311 0520  
 6312 7777  
 6313 0000  
 TST21  
 /DJR INHIBITED PMODE INTERRUPT SAVE  
 /TST21  
 EXITA  
 EXIT

TST22M, 0024  
 6314 0024  
 6315 2324  
 6316 6262  
 6317 0014  
 6320 1517  
 6321 0405  
 6322 4014  
 6323 1106  
 6324 4006  
 6325 0111  
 6326 1405  
 6327 0440  
 6330 2417  
 6331 4014  
 6332 1701  
 6333 0440  
 6334 2306  
 6335 0040  
 6336 0406  
 6337 4040  
 6340 4023  
 TST22  
 /LMODE LIF FAILED TO LOAD SF  
 / DF SF SF IN ERROR  
 /  
 EXITA  
 EXIT

6341 2640 0640  
 6342 4240 4040  
 6343 2306 2306  
 6344 4211 4011  
 6345 1640 1640  
 6346 0522 0522  
 6347 2217 2217  
 6350 2200 2200  
 6351 7777 EXITA  
 6352 0130 REGB  
 6353 0131 REGC  
 6354 0132 REGD  
 6355 0133 REGE  
 6356 0000 EXIT

TST23M, 0024  
 6357 0024 0024  
 6360 2324 2324  
 6361 6263 6263  
 6362 0014 0014  
 6363 1517 1517  
 6364 0405 0405  
 6365 4014 4014  
 6366 1106 1106  
 6367 4030 4030  
 6370 4006 4006  
 6371 0111 0111  
 6372 1405 1405  
 6373 0440 0440  
 6374 2417 2417  
 6375 4006 4006  
 6376 1116 1116  
 6377 0440 0440  
 6400 2022 2022  
 6401 1720 1720  
 6402 0522 0522  
 6403 4011 4011  
 6404 0600 0600  
 6405 2422 2422  
 6406 3140 3140  
 6407 4006 4006  
 6410 1725 1725  
 6411 1604 1604  
 6412 4000 4000  
 6413 7777 EXITA  
 6414 2130 REGB  
 6415 2131 REGC  
 6416 0000 EXIT

/TST23  
 /LMODE LIF X FAILED TO FIND CORRECT IF  
 /TRY FOUND

TST28M, 0024  
 6417 0024 0024  
 6420 2324 2324  
 6421 6270 6270  
 6422 0014 0014  
 6423 1517 1517  
 6424 0405 0405  
 6425 4014 4014

/TST28  
 /LMODE LIF FAILED TO INHIBIT INTERRUPTS  
 /BANK

6426 1106  
6427 4026  
6430 0111  
6431 1405  
6432 0442  
6433 2417  
6434 4011  
6435 1610  
6436 1102  
6437 1124  
6440 4011  
6441 1624  
6442 0522  
6443 2225  
6444 2024  
6445 2300  
6446 0201  
6447 1613  
6450 4000  
6451 7777  
6452 0130  
6453 0000

TST29M, 0024  
6454 0024  
6455 2324  
6456 6271  
6457 0014  
6460 1517  
6461 0405  
6462 4014  
6463 1106  
6464 4006  
6465 0111  
6466 1405  
6467 0440  
6470 2417  
6471 4014  
6472 1701  
6473 0440  
6474 2306  
6475 4000  
6476 4011  
6477 0640  
6500 4040  
6501 1106  
6502 0023  
6503 0516  
6504 2440  
6505 2203  
6506 2604  
6507 4000  
6510 7777  
6511 0130  
6512 0131  
6513 0000

/TST29  
/LMODE LIF FAILED TO LOAD SF

/ IF IF  
/SENT RCVD

EXITA  
RECB  
EXIT

/BASIC MEM TST PASS--(PASS)

6514	0002	TSTSTM, 0002
6515	0123	0123
6516	1123	1123
6517	4315	4315
6522	0515	0515
6521	4224	4224
6522	2324	2324
6523	4020	4020
6524	0123	0123
6525	2355	2355
6526	5555	5555
6527	7777	EXITA
6532	0017	COUNT
6531	4444	EXITB

/SPECIAL RESTART: EVENTUALLY GETS TO TST03

/SPURIOUS INTERRUPT!  
/(CHECK IOC I/O PRESET)

6532	0023	INTSTM, 0023
6533	2025	2025
6534	2211	2211
6535	1725	1725
6536	2340	2340
6537	1116	1116
6542	2405	2405
6541	2222	2222
6542	2520	2520
6543	2441	2441
6544	0050	0050
6545	0310	0310
6546	0503	0503
6547	1340	1340
6550	1117	1117
6551	0340	0340
6552	1157	1157
6553	1740	1740
6554	2022	2022
6555	0523	0523
6556	0524	0524
6557	5100	5100
6560	0000	EXIT

\$

ADD	2220	ASCII	5320	ASCRXT	5225	AUTO11	0215	AUTO12	0016
AZE	2450	BCL	1540	BELL	2266	BELLS	5337	BSE	1600
CLR	2411	COUNT	2217	CRLF	5153	CATUM	5114	DATYP	5162
DJR	2226	ERROR	2267	ERRORS	2220	EXC13	2535	EXIT	2000
EXITA	7777	EXITS	4444	FLAG	5253	HERE	5136	INCREL	5302
INTSTM	6532	INTST	5376	IOB	2522	KLDF	0106	KLIF	0107
KLJMP	2110	K0200	0270	K0240	0501	K0277	0272	K0100	0073
K0177	2274	K0207	0275	K2212	5175	K0215	5174	K0400	0076
K1026	2277	K1777	0100	K2000	0101	K2021	0102	K240	5176
K5252	2123	K6020	0104	K7605	1173	K7774	0105	LDA	1000
LDF	2640	LHAN	0042	LIF	0600	LINC	6141	LINTR	0040
LJMP	6220	LMSAK	0111	LNOP	0016	LOCA	0354	LOCC	0434
LOCCA	2422	LOCE	0477	LOCF	2757	LOCJ	1115	LOCO	1161
LOCP	5331	LPOINT	0065	LREG	2264	LREG1	0013	LSET	0050
LSKP	2456	LSTERR	0112	M203	1172	M40	5177	M4444	5173
NERROS	2113	NERROS	5002	NOW2	2324	OCTYP	5132	PASPNT	1162
PASS	1140	PDP	0202	PICKUP	5303	PINT	0014	PINTR	0000
PNTA	2114	PNTC	0115	PNTCA	0116	PNTF	0117	PNTF	0120
PNTJ	2121	PNTO	0122	PNTP	0123	PPOINT	0124	PREG	0125
RANDOM	2126	RANDY	5200	REDO	5137	REGA	0127	REGB	0130
REGC	2131	REGD	0132	REGE	0133	RNC	5261	RELOC	0134
RETURN	2135	RNA	5240	RNB	5241	RNC	5242	RND	5243
ROL	2240	ROR	2300	SET	0060	SETFLG	0136	SPACE	0137
SRO	1520	START	2202	STC	4000	TEST	0144	TESTN	1163
TRY22A	1201	TRY22N	0143	TRY23	1034	TRY28	1077	TSTINT	0140
TST03	2225	TST03M	5350	TST04	2235	TST04M	5402	TST06	0265
TST06N	5434	TST07	0345	TST07M	5466	TST09	0412	TST09M	5530
TST11	2456	TST11M	5562	TST13	2522	TST13A	0531	TST13M	5626
TST14	2555	TST14M	6040	TST15	2575	TST15M	6075	TST16	0616
TST16M	6123	TST17	0640	TST17M	6150	TST18	0656	TST18M	6174
TST19	2674	TST19M	6220	TST20	0715	TST20M	6244	TST21	0741
TST21M	6265	TST22	0771	TST22M	6314	TST23	1022	TST23M	6357
TST28	1257	TST28M	6417	TST29	1124	TST29M	6454	TST29N	0142
TST37M	6514	TST9A	0364	TST9AM	5507	TYPE	0141	TYPECH	5077
TYPOUT	5244	XSK	0220						

