

MASTER

J U U P P P P I I I I I T T T T E E E E E R R R R R R R R R O O O M M M	V V 2 2 2 6 6 6
J U U P P I T E R R R R R O D M M M M	V V 2 2 6
J U U P P I T E R R R R R O D M M M M	V V 2 6
J U U P P P P I T E E E E E R R R R R R R R R O D M M M M	V V 2 2 6 6 6 6
J U U P I T E R R R R R O D M M M	V V 2 6 6
J J U U P I T E R R R R R O D M M M	V V 2 .. 6 6
J J J U U U P I I I I I T E E E E E R R R R R O O O M M M	V 2 2 2 2 2 .. 6 6 6

1 2 2 2	1 0 0 0	4 4 7 7 7 7	1 2 2 2	1 9 9 9	8 8 8 2 2 2
11 2 2 :::	11 0 0 :::	4 4 7	11 2 2 /	11 9 9 /	8 8 2 2
1 2 :::	1 0 0 0 :::	4 4 7	1 2 /	1 9 9 /	8 8 2
1 2 2	1 0 0 0	4 4 4 4 7	1 2 2 /	1 9 9 9 /	8 8 8 2 2
1 2 :::	1 0 0 0 :::	4 7	1 2 /	1 9 /	8 8 2
1 2 :::	1 0 0 0 :::	4 7	1 2 /	1 9 /	8 8 2
11111 22222	11111 000	4 7	11111 22222	11111 999	888 22222

possible
improvements
noted

```

1: *      V2.6 BOOTROM FOR 6800 CPU CARD WITH IRQ MODIFICATION
2: *
3: *      BOOT ROM FOR IDB/M6800 RUNNING ON
4: *      THE WAVE MATE JUPITER II COMPUTER
5: *
6: *      WRITTEN BY DENNIS BROWN 3/12/78
7: *      COPYRIGHT 1979 WAVE MATE
8: *      EDITED 8/25/78 IRA BAXTER FOR IDB VER1.1
9: *      EDITED 1/23/79 DENNIS BROWN
10: *     EDITED 4/4/79 DENNIS PAINTER
11: *     EDITED 8/30/79 DENNIS PAINTER
12: *     EDITED 9/4/80 DWP FOR FASTER WM VIDE
13: *     EDITED 2/4/81 DWP FOR IDB IN RAM
14: *     V2.5 11/6/81 DWP FOR DUAL BOOT
15: *     V2.6 12/18/82 IDB TO MAKE DUAL BOOT WORK
16: *
17: *

```

*Add "Wave" mode
 to ROM.*

```

0000 0000 19:      ABS

```

*** Undefined Symbol.

```

20: **** EQUATES ****

```

```

FC00      21: TOPAGE EQU   #FC00  TOP PAGE ON SERIAL SYSTEMS
0040      22: LINSIZ EQU   64
C000      23: SCREEN EQU  #C000  SCREEN ADDRESS
0800      24: SCRISZ EQU  2048
0002      25: NULS1  EQU   2
0002      26: NULS2  EQU   2
008C      27: SKP2   EQU   #8C   CMPX # (SKIP 2 BYTES)
28: *
29: ***SYSTEM PAGE***
30: *
0080      31:      ORG   #80   RELATIVE TO PAGE
32: *
0080 0001 33: CURH   RMB   1   CURSOR HIGH HALF
0081 0001 34: CURL   RMB   1   CURSOR LOW HALF
0082 0001 35: CURCHR RMB   1   CHARACTER CURSOR IS ON TOP OF
0083 0015 36: TABTBL RMB  21   SYSTEM TAB TABLE FOR MTS
0098 0001 37: CURSTAT RMB   1   CURSOR ON/OFF FLAG
0099 0001 38: TRANSP RMB   1   TRANSPARENT VIDE MODE FLAG
39: *
00D3      40:      ORG   #D3   RELATIVE FROM SOME PAGE BOUNDRY
41: *
00D3 0002 42: SYSVARS RMB   2   POINTER TO USER VARIABLES IN MTS
00D5 0003 43: SYSCCI RMB   3   SYSTEM COMMAND LEVEL VECTOR
00D8 0003 44: SYSRST RMB   3   SYSTEM SOFT RESET VECTOR
00DB 0003 45: SYSSWI RMB   3   SYSTEM SOFTWARE INTERRUPT VECTOR
00DE 0002 46: SYSPWR RMB   2   POWER UP TEST CODE
00E0 0001 47: SYSMASK RMB   1   SYSTEM INTERRUPT MASK VALUE
00E1 0007 48: SYSCLK RMB   7   SYSTEM CLOCK
00E8 0003 49: SYSINT7 RMB   3   LEVEL 7 INT FLAG
00E8 0003 50: SYSINT6 RMB   3   LEVEL 6 INT VECTOR
00EE 0003 51: SYSINT5 RMB   3   LEVEL 5 INT VECTOR
00F1 0003 52: SYSINT4 RMB   3   LEVEL 4 INT VECTOR
00F4 0003 53: SYSINT3 RMB   3   LEVEL 3 INT VECTOR
00F7 0003 54: SYSINT2 RMB   3   LEVEL 2 INT VECTOR
00FA 0003 55: SYSINT1 RMB   3   LEVEL 1 INT VECTOR

```

*Mod ROM so IDB can
 use top \$80 bytes,
 ROM uses \$70-\$7F.*

These are deletable.

*many of these
 are deletable.
 Simply code that
 refers to this
 stuff.*

MAL/6800 1.3F: 00FD

12/18/82 12:03:57; Page 2; Form 1

JUPITERROM26.ASM

00FD 0003	56: SYSINT0 RMB	3	LEVEL 0 INT VECTOR
00FE	57: DR6	\$FE	SYSTEM PAGE POINTER
00FE 0001	58: SYSPG RMB	1	POINTER TO SYSTEM PAGE
00FF 0001	59: RUNNINGENCRYPTED	RMB 1	FLAG, IN ENCRYPTED MODE

← delete this ugly thing.
← move to system page

```

1: *
0100 0000 2: ABS
*** Undefined Symbol.
3: * START OF THE ROM
4:
5: *
FC00 6: ORG $FC00 BOOT ROM STARTING LOCATION
7: ***VECTOR JUMP TABLE***
8: *
FC00 7EFD25 9: BDDTV JMP BOOT BOOT LOAD VECTOR
FC03 7EFE11 10: INICV JMP INIS1 INIT BOTH SERIAL & VIDEO
11: *
FC06 7EFDE6 12: PUTCV JMP PUTS1 CONSOLE OUTPUT
FC09 7EFDC7 13: GETCV JMP GETS1 CONSOLE INPUT
FC0C 7EFDDB 14: TSTCV JMP TSTS1 CONSOLE INPUT TEST
FC0F 7EFC42 15: ESCCV JMP ESCC CONSOLE INPUT ESCAPE TEST
16: *
FC12 7EFDB1 17: INIFV JMP INIS2 INIT FILE DEVICE
FC15 7EFE0A 18: PUTFV JMP PUTS2 FILE OUTPUT
FC18 7EFDDB 19: GETFV JMP GETS2 FILE INPUT
FC1B 7EFE02 20: ODNFV JMP OUTFON FILE OUTPUT START
FC1E 7EFE05 21: DOFFFV JMP OUTFOFF FILE OUTPUT STOP
FC21 7EFD9C 22: IONFV JMP INFON FILE INPUT START
FC24 7EFDFF 23: IOFFFV JMP INFOFF FILE INPUT STOP
24: *
FC27 7EFD81 25: INILV JMP INIS2 INIT LIST DEVICE
FC2A 7EFE0A 26: PUTLV JMP PUTS2 LIST OUTPUT
FC2D 7EFD94 27: DONLV JMP RETURN LIST OUTPUT START
FC30 7EFD94 28: DOFFLV JMP RETURN LIST OUTPUT STOP
29: *
FC33 01 30: INTDISV NOP DISABLE INTERRUPTS VECTOR
FC34 0F 31: SEI
FC35 39 32: RTS
FC36 0E 33: INTENV CLI ENABLE INTERRUPTS VECTOR
FC37 39 34: RTS
FC38 01 35: NOP MAINTAIN OLD 3 BYTE JMP SPACE
FC39 3B 36: NOINT RTI RETURN FROM INTERRUPT VECTOR
FC3A 7EFE16 37: TSTMEM JMP TESTRAM
38: *
FC3D 8DF4 39: IDBNMI BSR INTDISV IDB NMI ENTRY POINT, STOP INTS
FC3F 7EF00B 40: JMP $F00B ENTER THE DEBUGGER
41: *
FC42 8DC8 42: ESCC BSR TSTCV CONSOLE ESCAPE INPUT TEST RETURNS
FC44 2606 43: BNE ESCCE ZERO FLAG SET IF ESCAPE HIT
FC46 8DC1 44: BSR GETCV GET DATA
FC4B 847F 45: ANDA #$7F MASK TO 7 BITS
FC4A 811B 46: CMPA #$1B CK IF ESCAPE CHAR
FC4C 39 47: ESCCE RTS
48: *

```

these 6 are wasted.

1: *
 2: * SYSTEM PAGE INITIAL VALUES
 3: *

```

FC4D      4: SYSINIT EQU      *
FC4D 7EFC3D 5:      JMP      IDBNMI  SYSCCI
FC50 7EFC05 6:      JMP      $F005  SYSRST
FC53 7EFC3D 7:      JMP      IDBNMI  SYSSWI
FC56 A55A   8:      FDB      $A55A  SYSPWR
FC58 00     9:      FCB      0      SYSMSK
FC59 00000000 10:     FCB      0,0,0,0,0,0,0  SYSCLK
FC60 020000 11:     FCB      2,0,0  SYSINT7 FLAG
FC63 7EFC39 12:     JMP      NOINT   SYSINT6
FC66 7EFC39 13:     JMP      NOINT   SYSINT5
FC69 7EFC39 14:     JMP      NOINT   SYSINT4
FC6C 7EFC39 15:     JMP      NOINT   SYSINT3
FC6F 7EFC39 16:     JMP      NOINT   SYSINT2
FC72 7EFC39 17:     JMP      NOINT   SYSINT1
FC75 7EFC39 18:     JMP      NOINT   SYSINT0
    
```

} simplify

19: *
 20: * I/O INTERRUPT ENTRY POINTS
 21: *
 22: * ALL INTS VECTOR THROUGH SYSTEM PAGE
 23: *

```

FC78      24: LEVEL0 EQU      *      INTERRUPT LEVEL 0
FC78 86FD  25:      LDAA     $SYSINT0&#xFF
FC7A 36    26:      PSHA
FC7B 96FE  27:      LDAA     SYSPG
FC7D 36    28:      PSHA
FC7E 39    29:      RTS
          30: *      LDX      SYSPG
          31: *      JMP      SYSINT0,X
          32: *
    
```

} simplify - also speeds up computer!!

```

FC7F      33: LEVEL7 EQU      *      INTERRUPT LEVEL 7 (FPI)
          34: *      LDAA     RUNNINGENCRYPTED
FC7F B6    35:      FCB      $B6      LDAA
FC80 00FF  36:      FDB      RUNNINGENCRYPTED
FC82 26B5  37:      BNE      NOINT
FC84 DEFE  38:      LDX      SYSPG
FC86 6FE8  39:      CLR      SYSINT7,X      SET FPI FLAG
FC88 CEC350 40:     LDX      $50000  SET TIMER
FC8B 09    41: FPITIM DEX
FC8C 26FD  42:      BNE      FPITIM
    
```

```

          43: *
FC8E      44: FINT2 EQU      *      FPI END
FC8E DEFE  45:      LDX      SYSPG
FC90 6CE8  46:      INC      SYSINT7,X      CHANGE FPI FLAG BACK TO NORMAL
FC92 6ED5  47:      JMP      SYSCCI,X      JUMP TO COMMAND LEVEL
          48: *
    
```

```

1: *
2: *      SYSTEM RESET ENTRY POINT
3: *
FC94      4: HARDRESET      EQU      *
FC94 7EFC9F 5:          JMP      RESET9
FC97 02000000 6:          FCB      2,0,0,0,0,0,0,0 SERIAL NUMBERS
FC9F 8E00FF 7: RESET9 LDS      #FF      PUSH RET ADDR INTO SYSPG PTR
8: *      LDAA      RUNNINGENCRYPTED
FCA2 B6 9:          FCB      $B6      LDAA
FCA3 00FF 10:         FDB      RUNNINGENCRYPTED
FCA5 2710 11:         BEQ      NOTENCRYPT
FCA7 8600 12:         LDAA      #0
FCA9 BDFC4C 13: CLR      JSR      ESCCE
FCAC CEBFFF 14:         LDX      #$BFFF
FCAF      15: LOOP      SET      *
FCAF 6F00 16:         CLR      0,X
FCB1 09 17:         DEX
FCB2 26FB 18:         BNE      LOOP
FCB4 4A 19:         DECA
FCB5 2AF2 20:         BPL      CLR
FCB7      21: NOTENCRYPT EQU      *
FCB7 CEFC00 22:         LDX      #TOPAGE
FCBA DFFE 23:         STX      SYSPG
24: *
FCBC 7A00FE 25: TOPLOOP DEC      SYSPG      TRY NEXT PAGE
FCBF DEFE 26:         LDX      SYSPG
FCC1 A600 27:         LDAA      0,X      WILL THE LOCATION HOLD
FCC3 43 28:         COMA      IT'S COMPLEMENT?
FCC4 A700 29:         STAA      0,X
FCC6 A000 30:         SUBA      0,X
FCC8 26F2 31:         BNE      TOPLOOP B/ NO MEMORY HERE
FCCA 6300 32:         COM      0,X      RESTORE THE BYTE
FCCC 6DE8 33:         TST      SYSINT7,X      FPI SWITCH DOWN?
FCCE 2707 34:         BEQ      RESET B/ YES HIT FULL RESET
35: *
FCD0 EEDE 36:         LDX      SYSPWR,X      GET POWER UP CODE
FCD2 BCA55A 37:         CPX      #$A55A
FCD5 272D 38:         BEQ      SOFTRESET      POWER UP RESET IF CODE TRASHED.
39: *
FCD7      40: RESET EQU      *      POWER UP RESET ENTRY CODE
FCD7 DEFE 41:         LDX      SYSPG      RESTORE SYSTEM PAGE
FCD9 BEFC4C 42:         LDS      #SYSINIT-1      INITIALIZE SYSTEM PAGE
43: *
44: *
FCDC C6D5 45:         LDAB     #$D5
46: *
FCDE      47: SYSLOOP EQU      *      COPY LOOP
FCDE 32 48:         PULA
FCDF A7D5 49:         STAA     $D5,X
FCE1 08 50:         INX
FCE2 5C 51:         INCB
FCE3 26F9 52:         BNE      SYSLOOP
53: *
FCE5 FEF001 54:         LDX      #F001      CHECK TO SEE IF IDB IS THERE
FCE8 8C01BD 55:         CPX      #01BD

```

} simplify

JUPITERROM26.ASM

```

FCEB 2607      56:      BNE      SETBOOT B/ NOT HERE
FCED B6F043    57:      LDAA     #F043  GET IDB SCRATCH PAGE
FCF0 91FE      58:      CMPA     #FE     SEE IF SYSTEM HAS RAM THERE
FCF2 2310      59:      BLS      SOFTRESET      B/ YES, DON'T BOOT
FCF4 DEFE      60: SETBOOT LDX     SYSPB
FCF6 B6FC      61:      LDAA     #BOOTV/256     IF NOT MAKE RESET LEVEL
FCF8 C600      62:      LDAB     #BOOTV&255
FCFA A7D9      63:      STAA     SYSRST+1,X     POINT TO BOOT ROUTINE
FCFC E7DA      64:      STAB     SYSRST+2,X
FCFE C639      65:      LDAB     #NOINT&255     NO IDB HERE SO POINT FPI
FD00 A7D6      66:      STAA     SYSCCI+1,X     VECTOR TO NOINT SUBR
FD02 E7D7      67:      STAB     SYSCCI+2,X
68: *
69: *      SOFT RESET ENTRY POINT
70: *
FD04           71: SOFTRESET      EQU      *
FD04 9EFE      72:      LDS      SYSPB  INIT STACK POINTER
FD06 31        73:      INS
FD07 31        74:      INS
FD08 31        75:      INS      STAY OFF RAM PAGE BELOW SYSPAGE
FD09 BDFC03    76:      JSR      INICV  INIT CONSOLE DEVICE
FD0C BDFC12    77:      JSR      INIFV  INIT DEFAULT DEVICE
FD0F BDFC27    78:      JSR      INILV  INIT LIST DEVICE
FD12 9EFE      79:      LDS      SYSPB  NOW SET UP CONTEXT BLOCK
FD14 30        80:      TSX      POINT TO SYSPB (PLUS 1)
FD15 B625      81:      LDAA     #BOOT&255     SET UP USER ENVIRONMENT
FD17 A706      82:      STAA     6,X      SO THAT A GO COMMAND
FD19 B6FD      83:      LDAA     #BOOT/256     WILL BOOTSTRAP IN A BIGGER
FD1B A705      84:      STAA     5,X      SYSTEM PROGRAM.
FD1D A700      85:      STAA     0,X      USE THE BOOT LOADER PAGE #FD TO SET CCR
86: *      AS INTERRUPTS OFF
FD1F 6ED7      87:      JMP      SYSRST-1,X     AND GO TO COMMAND LEVEL
88: *
89: *      SWI ENTRY POINT, VECTORS THROUGH THE SYSTEM PAGE SYSSWI
90: *
FD21           91: SWIV      EQU      *
FD21 DEFE      92:      LDX      SYSPB
FD23 6EDB      93:      JMP      SYSSWI,X     JUMP THROUGH VECTOR

```

```

1: *   DUAL BOOT LOADER FOR EITHER FDI-125 & PERSCI 277 OR FDI-127 & T&E
2: *   BOOT TRIES TO LOAD FIRST SECTOR FROM EITHER DEVICE WHICH INDICATES
3: *   READY STATUS. IF A SECTOR ZERO IS NOT FOUND IT TRIES TO FIND A SECTOR
4: *   NUMBER ONE. IF SECTOR NUMBER ONE IS FOUND ON AN 8 INCH DRIVE IT
5: *   ASSUMES IBM FORMAT DISK AND COMPLEMENTS THE DATA. IF THE FIRST
6: *   INSTRUCTION IS NOT A LOAD STACK IMMEDIATE ($8E) IT DOES NOT EXECUTE
7: *   THE LOADED BOOTSTRAP.
FD25 8D3A 8: BOOT  BSR  INITIALIZE  INITIALIZE WHATEVER HARDWARE IS ON SYSTEM
FD27 8199 9:      CMPA  #$99  ANY ERRORS?
FD29 26FA 10:     BNE   BOOT    B/ WELL TRY AGAIN
11:
FD2B 4F   12: RETRY CLRA      SET SECTOR 0
FD2C 8D22 13:     BSR  READSECT  TRY TO READ IT
FD2E 2716 14:     BEQ  TSTBOOT  B/ NO ERRORS CK FOR GOOD BOOTSTRAP DATA
FD30 818D 15:     CMPA  #$8D  RECORD NOT FOUND?
FD32 26F7 16:     BNE  RETRY   B/ NO, TRY AGAIN
FD34 8601 17:     LDAA  #1     TRY SECTOR ONE
FD36 8D18 18:     BSR  READSECT
FD38 26F1 19:     BNE  RETRY   B/ ERROR, TRY SECTOR 0 AGAIN
FD3A 8D71 20:     BSR  ISMINI  READ MINI SECTOR 1?
FD3C 2708 21:     BEQ  TSTBOOT  B/ YES, DON'T COMPLEMENT DATA
FD3E CE080 22:    LDX  #$17F-$FF  MUST BE IBM FORMAT, COMPLEMENT IT
FD41 63FF 23:  COMDATA COM  $FF,X
FD43 09   24:     DEX
FD44 26FB 25:     BNE  COMDATA
26:
FD46 B60100 27: TSTBOOT LDAA  $100  FIRST OPCODE MUST BE
FD49 818E 28:     CMPA  #$8E  LOAD STACK IMMEDIATE
FD4B 26D8 29:     BNE  BOOT    OR WE ABORT THE WHOLE SHOW
FD4D 7E0100 30:    JMP  $100  GO EXECUTE THE BOOTSTRAP
31:
32: * READ SELECTED SECTOR SUBROUTINE
33:
FD50 8D5B 34: READSECT  BSR  ISMINI  CK WHICH DRIVE IS SELECTED
FD52 2701 35:     BEQ  SETSECT  B/ MINI, DON'T COMPLEMENT SECTOR NUMBER
FD54 43   36:     COMA
FD55 A706 37: SETSECT STAA  6,X  SELECT 17XX SECTOR
FD57 8601 38:     LDAA  #1     SET READ PAGE
FD59 A702 39:     STAA  2,X
FD5B 17   40:     TBA      GET READ COMMAND INTO ACCA
FD5C 8D27 41:     BSR  CMD    ISSUE READ COMMAND
FD5E 819D 42:     CMPA  #$9D  CK FOR ERRORS
FD60 39   43:     RTS      AND RETURN THEM
44:
45: * INITIALIZE HARDWARE
46:
FD61 47: INITIALIZE  EQU  *
FD61 CEFFA0 48: INIBIN LDX  #$FFA0  SET BASE ADDRESS OF PERSCI
FD64 8D2F 49:     BSR  INIHWD
FD66 A103 50:     CMPA  3,X  DOES THE HARDWARE EXIST?
FD68 260A 51:     BNE  INISIN  B/ NO, TRY THE MINI-FLOPPY
FD6A A604 52:     LDAA  4,X  IS DRIVE READY?
FD6C 2A06 53:     BPL  INISIN  B/ NO, TRY 5" DRIVE
FD6E 86F0 54:     LDAA  #$F0  ISSUE RESTORE COMMAND
FD70 C673 55:     LDAB  #$73  SET UP READ COMMAND FOR MAIN

```



```

FD72 2011      56:      BRA      CMD
              57:
FD74 CEFF80   58: INISIN LDX    #FF80 SET BASE ADDRESS OF T&E
FD77 8D1C     59:      BSR      INIHWD
FD79 A103     60:      CMPA     3,X    DOES THE HARDWARE EXIST?
FD7B 26E4     61:      BNE      INIBIN B/ NO, TRY THE 8 INCH DRIVE
FD7D A604     62:      LDAA     4,X    IS DRIVE READY?
FD7F 2BE0     63:      BMI      INIBIN B/ NO, TRY 8" DRIVE
FD81 860D     64:      LDAA     #0D    SET RESTORE FLAG
FD83 C680     65:      LDAB     #80    SET UP READ COMMAND FOR MAIN
              66:
FD85 A704     67: CMD    STAA     4,X    ISSUE COMMAND TO HARDWARE
FD87 A601     68: FDWAIT LDAA     1,X    WAIT FOR DONE
FD89 2AFC     69:      BPL      FDWAIT
FD8B A604     70:      LDAA     4,X    GET CONTROLLER STATUS
FD8D 8D1C     71:      BSR      DISKACKDONE ACK DONE FLAGS, WHICH DISK DRIVE ?
FD8F 2601     72:      BNE      FDCMDX B/ GOT 8 INCH STATUS, EXIT
FD91 43       73:      COMA                      CONVERT 5 INCH STATUS TO LOOK LIKE 8 INCH STATUS
FD92 849D     74: FDCMDX ANDA    #9D    STRIP GARBAGE BITS
FD94 39       75: RETURN RTS
              76:
              77:
              78: * INITIALIZE HARDWARE POINTED TO BY X-REG
              79:
FD95 4F       80: INIHWD CLRA                      GET A ZERO
FD96 A700     81:      STAA     0,X    INITIALIZE PIA
FD98 A701     82:      STAA     1,X
FD9A 4A       83:      DECA                      GET A #FF
FD9B A702     84:      STAA     2,X
FD9D A703     85:      STAA     3,X
FD9F 862C     86:      LDAA     #2C
FDA1 A700     87:      STAA     0,X
FDA3 8616     88:      LDAA     #16
FDA5 A701     89:      STAA     1,X
FDA7 8640     90:      LDAA     #40    (60 FOR MINI, 48 FOR BIN, HOPE THIS WORKS)
FDA9 A703     91:      STAA     3,X    SELECT UNIT 0
  FDAB       92: DISKACKDONE ; ACKNOWLEDGE DISK DONE FLAGS
FDAB A503     93:      BITA     3,X    CLR DONE FLAGS
  FDAD       94: ISMINI ; CHECK FOR MINI FLOPPY: RETURN Z SET IF MINI
FDAD 8CFF80   95:      CPX      #FF80 USING MINIFLOPPY HARDWARE BASE ?
FDB0 39       96:      RTS
  
```

```
1: *  
2: * SERIAL 2 INITIALIZATION  
3: *  
4: * M6850 ACIA BASED AT $FFC4  
5: * INIT FOR 1 START/ 8 DATA/ 2 STOP BITS  
6: * /16 CLOCK/ NO INTERRUPTS ENABLED  
7: *  
FDB1 CEFFC4 8: INIS2 LDX $FFC4  
FDB4 8603 9: INISA LDAA #3 GET RESET CODE  
FDB6 A700 10: STAA 0,X AND STUFF IT INTO ACIA  
FDB8 8611 11: LDAA #$11 GET INIT CODE  
FDBA A700 12: STAA 0,X AND STUFF IT  
FDBC 39 13: RTS EXIT  
14: *  
15: * SERIAL 1 INPUT DONE TEST  
16: *  
17: * M6850 ACIA BASED AT $FFC0  
18: * SETS STATUS ZERO IF DONE  
19: *  
FDBD CEFFC0 20: TSTS1 LDX $FFC0  
FDC0 E600 21: TSTSA LDAB 0,X GET STATUS BITS  
FDC2 C401 22: ANDB #1  
FDC4 C801 23: EDRB #1 TEST BOTTOM BIT  
FDC6 39 24: RTS EXIT  
25: *  
26: * SERIAL 1 INPUT  
27: *  
28: * M6850 ACIA BASED AT $FFC0  
29: * RETURNS CHARACTER IN A  
30: *  
FDC7 CEFFC0 31: GETS1 LDX $FFC0  
FDCA 8DF4 32: GETSA BSR TSTSA  
FDCC 26FC 33: BNE GETSA WAIT FOR INPUT DONE  
FDCE A601 34: LDAA 1,X GET CHARACTER  
FDD0 847F 35: ANDA #$7F MASK IT  
FDD2 27F6 36: BEQ GETSA IGNORE NULLS  
FDD4 39 37: RTS EXIT  
38: *  
39: * SERIAL 2 INPUT  
40: *  
FDD5 CEFFC4 41: GETS2 LDX $FFC4  
FDD8 20F0 42: BRA GETSA  
43:  
44: *  
45: * SERIAL 1 OUTPUT TEST  
46: *  
47: * M6850 ACIA BASED AT $FFC0  
48: * SETS STATUS ZERO IF OUTPUT DONE  
49: *  
FDDA 50: TSTS10 EQU *  
FDDA CEFFC0 51: LDX $FFC0  
FDD8 37 52: TSTSA0 PSHB SAVE USR ACCB  
FDDE E600 53: LDAB 0,X  
FDE0 C402 54: ANDB #$2  
FDE2 C802 55: EDRB #$2 TEST BIT 1 OF ACIA
```

```
FDE4 33      56:      PULB      RESTORE USR ACCB
FDE5 39      57:      RTS        EXIT
          58: *
          59: *      SERIAL 1 OUTPUT
          60: *
          61: *      CHARACTER TO OUTPUT IS IN A
          62: *      A LINE FEED GETS SOME NULLS AFTER IT
          63: *
FDE6        64: PUTS1  EQU      *
FDE6 C602    65:      LDAB     #NULS1  SET NUMBER OF NULLS AFTER A LF
FDEB CEFFC0  66:      LDX     #FFC0
FDEB        67: PUTSA  EQU      *
FDEB 8DF0    68: PUTSB  BSR     TSTSAD
FDED 26FC    69:      BNE     PUTSB  WAIT FOR OUTPUT DONE
FDEF A701    70:      STAA   1,X    STUFF CHR INTO ACIA
FDF1 810A    71:      CMPA   ##A
FDF3 2606    72:      BNE     PUTS1E  EXIT IF NOT A LF
FDF5 4F      73: PUTSN  CLRA     PUT A NULL
FDF6 8DF3    74:      BSR     PUTSA  PUT A NULL
FDF8 5A      75:      DECB
FDF9 26FA    76:      BNE     PUTSN
FDFB 39      77: PUTS1E  RTS        EXIT
          78:
          79: *      FILE READER START --OUTPUT DC1
          80: *      FILE READER STOP ---OUTPUT DC3
          81: *      FILE PUNCH START --OUTPUT DC2
          82: *      FILE PUNCH STOP ---OUTPUT DC4
          83: *
FDFC 8611    84: INFON  LDAA   ##11  INPUT ON
FDFE 8C      85:      SKP2
FDFE 8613    86: INFOFF LDAA   ##13  INPUT OFF
FE01 8C      87:      SKP2
FE02 8612    88: OUTFON  LDAA   ##12  OUTPUT ON
FE04 8C      89:      SKP2
FE05 8614    90: OUTFOFF LDAA   ##14  OUTPUT OFF
FE07 7EFC15  91:      JMP     PUTFV  OUTPUT CODE TO FILE DEVICE
          92:
          93: *
          94: *      SERIAL 2 OUTPUT
          95: *
          96: *      ACIA BASED AT #FFC4
          97: *
FE0A C602    98: PUTS2  LDAB     #NULS2
FE0C CEFFC4  99:      LDX     #FFC4
FE0F 20DA    100:     BRA     PUTSA
          101: *
          102: *      SERIAL 1 INITIALIZATION
          103: *
FE11 CEFFC0  104: INIS1  LDX     #FFC0
FE14 209E    105:     BRA     INISA
          106:
```

JUPITERROM26.ASM

```

0000      1: UPPER EQU 0      TWO BYTES FOR UPPER LIMIT
0002      2: LOWER EQU 2      TWO BYTES FOR LOWER LIMIT
0004      3: COUNT EQU 4      ONE BYTE
0005      4: PASS EQU 5       ONE BYTE
0006      5: PATT EQU 6       ONE BYTE
        6: * MEMORY TEST
        7: *      DOES PATTERN TEST ON ALL OF MEMORY
        8: *      DISPLAYS SLOWLY INCREMENTING PATTERN NUMBER 0 TO $FF
        9: *      PRINTS "VERIFIED" OR "ERR @XXXX GOOD=GG BAD=BB"
       10:
FE16 CE0000 11: TESTRAM LDX #0
FE19 DF01   12:      STX  UPPER+1 SET LOW BYTE OF UPPER & HI BYTE OF LOWER
FE1B DF04   13:      STX  COUNT  SET COUNT & PASS
FE1D DF06   14:      STX  PATT   SET PATT
FE1F 8607   15:      LDAA #PATT+1 SET LOWER LIMIT
FE21 9703   16:      STAA LOWER+1
FE23 96FE   17:      LDAA SYSPB  SET UPPER LIMIT
FE25 4C     18:      INCA
FE26 9700   19:      STAA  UPPER
FE28 0F     20:      SEI
       21:
       22: * DO PATTERN SENSITIVITY TEST
       23:
FE29 730005 24: PTEST1 COM  PASS
FE2C 261D   25:      BNE  INIT
FE2E 9606   26:      LDAA PATT
FE30 4C     27:      INCA
FE31 840F   28:      ANDA #$F
FE33 9706   29:      STAA PATT
FE35 2614   30:      BNE  INIT
FE37 DE02   31:      LDX  LOWER  LONG TST COMES HERE FIRST
FE39 08     32:      INX      SO LOCATION LOWER+1 IS NOT TESTED
FE3A DF02   33:      STX  LOWER
FE3C D604   34:      LDAB COUNT
FE3E 5C     35:      INCB
FE3F D704   36:      STAB COUNT  SAVE NEW COUNT
FE41 2759   37:      BEQ  DONE  MEMORY DIAGNOSTIC DONE
FE43 BDFECA 38:      JSR  PRINTSTR
FE46 0888   39:      FCB  $B,$B+$B0
FE48 17     40:      TBA      GET COUNT IN ACCA
FE49 8D5E   41:      BSR  PUTHX  DISPLAY
       42:
       43: * FILL MEMORY FROM LOWER TO UPPER WITH PATTERN OF ALTERNATING
       44: * BITS WHICH SHIFT FROM $55 TO $AA ON BOUNDARIES DEFINED BY
       45: * THE VARIABLE PATT. IF PATT = 0 THEN FILL ALL MEMORY WITH
       46: * AN ALTERNATING PATTERN.
       47:
FE4B DE02   48: INIT  LDX  LOWER  FILL FROM LOWER TO UPPER
FE4D 8655   49:      LDAA #$55
FE4F 9805   50:      EORA PASS
FE51 D606   51: INIT1 LDAB PATT
FE53 43     52:      COMA
FE54 A700   53: INIT2 STAA X
FE56 08     54:      INX
FE57 9C00   55:      CPX  UPPER

```

JUPITERROM26.ASM

```

FE59 2708    56:      BEQ     CHKUP
FE5B 5D      57:      TSTB     IF PATT = 0
FE5C 27F6    58:      BEQ     INIT2  THEN FILL ALL MEMORY W/ PASS=[0 OR FF]
FE5E 5A      59:      DECB     ELSE ALTERNATE BIT PATTERN
FE5F 26F3    60:      BNE     INIT2  BASED ON RANGE 1-15
FE61 20EE    61:      BRA     INIT1  WHEN RANGE = 0, RESET RANGE
62:
63: *
64:
FE63 DE02    65: CHKUP  LDX     LOWER  CHECK FROM LOWER TO UPPER
FE65 8655    66:      LDAA    #55
FE67 9805    67:      EORA    PASS
FE69 D606    68: CHKUP1  LDAB    PATT
FE6B 43      69:      COMA
FE6C A100    70: CHKUP2  CMPA    X
FE6E 2661    71:      BNE     ERROR
FE70 43      72:      COMA
FE71 A700    73:      STAA   X
FE73 43      74:      COMA
FE74 08      75:      INX
FE75 9C00    76:      CPX     UPPER
FE77 2708    77:      BEQ     CHKDN
FE79 5D      78:      TSTB
FE7A 27F0    79:      BEQ     CHKUP2
FE7C 5A      80:      DECB
FE7D 26ED    81:      BNE     CHKUP2
FE7F 20E8    82:      BRA     CHKUP1
83:
84: *
85:
FE81 43      86: CHKDN  COMA          CHECK FROM UPPER TO LOWER
FE82 09      87: CHKDN1  DEX
FE83 A100    88:      CMPA    X
FE85 264A    89:      BNE     ERROR
FE87 43      90:      COMA
FE88 A700    91:      STAA   X
FE8A 43      92:      COMA
FE8B 9C02    93:      CPX     LOWER
FE8D 279A    94:      BEQ     PTEST1
FE8F 5D      95:      TSTB
FE90 27F0    96:      BEQ     CHKDN1
FE92 5C      97:      INCB
FE93 D106    98:      CMPB    PATT
FE95 23EB    99:      BLS     CHKDN1
FE97 43      100:     COMA
FE98 C601    101:     LDAB    #1
FE9A 20E6    102:     BRA     CHKDN1
103:
FE9C 8D26    104: DONE  BSR     PRINTSTR
FE9E 20564552 105:     FCC    / VERIFIE/
FEA6 C4      106:     FCB    'D+#80
FEA7 20FE    107:     BRA     *
108:
109: * PUTHX OUTPUTS ASCII OF HEX IN ACCA
110:

```

```

FEA9 36      111: PUTHX PSHA          SAVE A COPY
FEAA 44      112:      LSRA
FEAB 44      113:      LSRA
FEAC 44      114:      LSRA
FEAD 44      115:      LSRA
FEAE 8D01    116:      BSR    PUTHX1
FEB0 32      117:      PULA
FEB1 840F    118: PUTHX1 ANDA    #$F
FEB3 8B90    119:      ADDA    #$90  CONVERT TO ASCII
FEB5 19      120:      DAA
FEB6 8940    121:      ADCA    #$40
FEB8 19      122:      DAA
FEB9 37      123: PUTHX2 PSHB          SAVE ACCUMULATOR
FEBA 8DFC06  124:      JSR    PUTCV
FEBD 33      125:      PULB          RESTORE ACCUMULATOR
FEBE 39      126:      RTS
127:
128: * PRINT STRING SUBROUTINE. PRST IS TOP OF STRING LOOP, NOT ENTRY POINT.
129: * PRINTS STRING POINTED TO BY RETURN ADDRESS ON TOP OF STACK. STRING
130: * TERMINATED BY BIT 7 SET ON ON LAST CHAR.
131:
FEBF 8DF8    132: PRST   BSR    PUTHX2 OUTPUT DATA IN ACCA
FEC1 30      133:      TSX          INDEX INTO STACK
FEC2 6C01    134:      INC    1,X    ADVANCE RETURN ADDRESS POINTER
135: ;      BNE    PRINTSTR    ** ** NOTE. THIS BRANCH WILL ALWAYS BE
136: ;      INC    0,X    IN BOOT ROM BECAUSE OF PLACMENT OF CALLS.
137:
138: * ENTRY POINT OF SUBROUTINE.
139:
FEC4 30      140: PRINTSTR    TSX          ;NEEDED AT ENTRY TO SUBR
FEC5 EE00    141:      LDX    0,X    POINT TO DATA TO OUTPUT
FEC7 A600    142:      LDAA   0,X    GET DATA BYTE IN ACCA FOR OUTPUT
FEC9 2AF4    143:      BPL    PRST   B/ NOT AT END OF DATA STRING
FECB 8DEC    144:      BSR    PUTHX2 OUTPUT LAST BYTE
FECD 30      145:      TSX          ADVANCE RETURN ADDRESS
FECE 6C01    146:      INC    1,X
FED0 39      147:      RTS
148:
149:
150: * ERROR, REPORT GOOD, BAD, AND ADDRESS
151:
FED1 E600    152: ERROR  LDAB   0,X    GET BAD BYTE
FED3 36      153:      PSHA          SAVE GOOD BYTE
FED4 DF02    154:      STX    LOWER  SAVE ADDRESS
FED6 8DEC    155:      BSR    PRINTSTR  PRINT HEADER
FED8 20455252 156:      FCC    / ERR @/
FEDE A0      157:      FCB    $20+$80
FEDF 9602    158:      LDAA   LOWER
FEE1 8DC6    159:      BSR    PUTHX
FEE3 9603    160:      LDAA   LOWER+1
FEE5 8DC2    161:      BSR    PUTHX PRINT ADDRESS
FEE7 8DD8    162:      BSR    PRINTSTR
FEE9 20474F4F 163:      FCC    / GOOD/
FEEE 8D      164:      FCB    '=$80
FEFF 32      165:      PULA          GET GOOD BYTE
  
```

MAL/6800 1.3F: FEF0
12/18/82 12:03:57; Page 14; Form 7
JUPITERROM26.ASM

FEF0 8DB7	166:	BSR	PUTHEX
FEF2 8DD0	167:	BSR	PRINTSTR
FEF4 20424144	168:	FCC	/ BAD/
FEF8 BD	169:	FCB	'=+\$80
FEF9 17	170:	TBA	GET BAD BYTE
FEFA 8DAD	171:	BSR	PUTHEX
FEFC 20FE	172:	BRA	*
	173:		

```
1: *      ROM I/O AREA
2:
FF00      3:      ORG      $FF00  START OF I/O AREA
4:
5: *
6: * MINI-FLOPPY DISK CONTROLLER
7: *
FF80      8:      ORG      $FF80  DISK CONTROLLER ADDRESS
FF80 0002  9:      RMB      2      PIA STATUS REGISTERS
FF82 0002 10:      RMB      2      PIA DATA REGISTERS
FF84 0001 11:      RMB      1      FD 1793 COMMAND/STATUS REGISTER
FF85 0001 12:      RMB      1      FD 1793 CURRENT TRACK REGISTER
FF86 0001 13:      RMB      1      FD 1793 NEXT SECTOR REGISTER
FF87 0001 14:      RMB      1      FD 1793 NEXT TRACK REGISTER
15: *
16: * HYTYPE PARALLEL PRINTER CONTROLLER
17: *
FF90      18:     ORG      $FF90  HYTYPE PIA ADDRESS
FF90 0002 19:     RMB      2      STATUS REGISTERS
FF92 0002 20:     RMB      2      DATA REGISTERS
21: *
22: * PERSCI FLOPPY DISK CONTROLLER
23: *
FFA0      24:     ORG      $FFA0  DISK CONTROLLER ADDRESS
FFA0 0002 25:     RMB      2      PIA STATUS REGISTERS
FFA2 0002 26:     RMB      2      PIA DATA REGISTERS
FFA4 0001 27:     RMB      1      FD 1771 COMMAND/STATUS REGISTER
FFA5 0001 28:     RMB      1      FD 1771 CURRENT TRACK REGISTER
FFA6 0001 29:     RMB      1      FD 1771 NEXT SECTOR REGISTER
FFA7 0001 30:     RMB      1      FD 1771 NEXT TRACK REGISTER
31: *
32: * WAVE MATE VIDED KEYBOARD (SCREEN AT $C000)
33: *
FFB0      34:     ORG      $FFB0  VIDED PIA ADDRESS
FFB0 0002 35:     RMB      2      STATUS REGISTERS
FFB2 0002 36:     RMB      2      DATA REGISTERS
```


1: *

2: * SERIAL PORTS

3: *

FFC0	4:	ORG	\$FFC0	SERIAL ACIA ADDRESS AREA
FFC0 0002	5:	RMB	2	SERIAL PORT 0
FFC2 0002	6:	RMB	2	UNASSIGNED
FFC4 0002	7:	RMB	2	SERIAL PORT 1
FFC6 0002	8:	RMB	2	UNASSIGNED
FFC8 0002	9:	RMB	2	SERIAL PORT 2
FFCA 0002	10:	RMB	2	UNASSIGNED
FFCC 0002	11:	RMB	2	SERIAL PORT 3
FFCE 0002	12:	RMB	2	UNASSIGNED
FFD0 0002	13:	RMB	2	SERIAL PORT 4 (OVERLAY CASSETTE I/O)
FFD2 0002	14:	RMB	2	UNASSIGNED
FFD4 0002	15:	RMB	2	SERIAL PORT 5
FFD6 0002	16:	RMB	2	UNASSIGNED
FFD8 0002	17:	RMB	2	SERIAL PORT 6
FFDA 0002	18:	RMB	2	UNASSIGNED
FFDC 0002	19:	RMB	2	SERIAL PORT 7
FFDE 0002	20:	RMB	2	UNASSIGNED

21: *

22: * CASSETTE PORT

23: *

FFD0	24:	ORG	\$FFD0	CASSETTE ADDRESS
FFD0 0002	25:	RMB	2	CASSETTE ACIA

26: *

27: * EPROM PROGRAMMER PORT (2704-2708 EPROMS)

28: *

FFD4	29:	ORG	\$FFD4	
FFD4 0001	30:	RMB	1	DATA A
FFD5 0001	31:	RMB	1	STATUS A
FFD6 0001	32:	RMB	1	DATA B
FFD7 0001	33:	RMB	1	STATUS B

34: *

35: * HARDWARE BREAKPOINT REGISTERS

36: *

FFD8	37:	ORG	\$FFD8	BREAKPOINT PIA ADDRESS
FFD8 0002	38:	RMB	2	STATUS REGISTERS
FFDA 0002	39:	RMB	2	DATA REGISTERS

40:

41:

42: * ROM INTERRUPT VECTORS

43: *

FFF8	44:	ORG	\$FFF8	
FFF8 FC78	45:	FDB	LEVEL0	
FFFA FD21	46:	FDB	SWIV	
FFFC FC7F	47:	FDB	LEVEL7	
FFFE FC94	48:	FDB	HARDRESET	
0000	49:	END		

JUPITERROM26.ASM

Symbols Sorted by Name:

ABS/****	BOOT/FD25	BOOTV/FC00	CHKDN/FEB1	CHKDN1/FE82	CHKUP/FE63	CHKUP1/FE69	CHKUP2/FE6C
CLR/FCA9	CMD/FD85	COMDATA/FD41	COUNT/0004	*CURCHR/0082	*CURH/0080	*CURL/0081	*CURSTAT/0098
DISKACKDONE/FDAB		DONE/FE9C	ERROR/FED1	ESCC/FC42	ESCCE/FC4C	*ESCCV/FC0F	FDCMDX/FD92
FDWAIT/FD87	*FINT2/FC8E	FPITIM/FC8B	GETCV/FC09	*GETFV/FC18	GETS1/FDC7	GETS2/FDD5	GETSA/FDCA
HARDRESET/FC94	IDBNMI/FC3D	INFOFF/FDFF	INFON/FDFC	INISIN/FD74	INISIN/FD61	INICV/FC03	INIFV/FC12
INIHWI/FD95	INILV/FC27	INIS1/FE11	INIS2/FDB1	INISA/FDB4	INIT/FE48	INIT1/FE51	INIT2/FE54
INITIALIZE/FD61		INTDISV/FC33	*INTENV/FC36	*IOFFV/FC24	*IONFV/FC21	ISKINI/FDAD	LEVEL0/FC78
LEVEL7/FC7F	*LINSIZ/0040	LOOP/FCAF	LOWER/0002	NOINT/FC39	NOTENCRYPT/FCB7		NULS1/0002
NULS2/0002	*ODFFV/FC1E	*ODFFLV/FC30	*OONFV/FC1B	*OONLV/FC2D	OUTFOFF/FE05	OUTFON/FE02	PASS/0005
PATT/0006	PRINTSTR/FEC4	PRST/FE8F	PTEST1/FE29	PUTCV/FC06	PUTFV/FC15	PUTHEX/FEA9	PUTHEX1/FEB1
PUTHEX2/FEB9	*PUTLV/FC2A	PUTS1/FDE6	PUTS1E/FDFB	PUTS2/FE0A	PUTSA/FDEB	PUTSB/FDEB	PUTSN/FDF5
READSECT/FD50	RESET/FC07	RESET9/FC9F	RETRY/FD2B	RETURN/FD94	RUNNINGENCRYPTED/00FF		*SCREEN/C000
*SERSIZ/0800	SETBOOT/FCF4	SETSECT/FD55	SKP2/008C	SOFTRESET/FD04	SWIV/FD21	SYSCCI/00D5	*SYSCLK/00E1
SYSINIT/FC4D	SYSINT0/00FD	*SYSINT1/00FA	*SYSINT2/00F7	*SYSINT3/00F4	*SYSINT4/00F1	*SYSINT5/00EE	*SYSINT6/00EB
SYSINT7/00E8	SYSLOOP/FCDE	*SYSMSK/00E0	SYSPG/00FE	SYSPWR/00DE	SYSRST/00D8	SYSSWI/00DB	*SYSVARS/00D3
*TABTBL/00B3	TESTRAM/FE16	TOPAGE/FC00	TOPLOOP/FCBC	*TRANSP/0099	TSTBOOT/FD46	TSTCV/FC0C	*TSTMEM/FC3A
TSTS1/FD8D	*TSTS10/FDDA	TTSA/FDC0	TSTSA/D0DD	UPPER/0000			

JUPITERROM26.ASM

Symbols Sorted by Value:

ABS/****	UPPER/0000	LOWER/0002	NULS1/0002	NULS2/0002	COUNT/0004	PASS/0005	PATT/0006
*LINSIZ/0040	*CURH/0080	*CURL/0081	*CURCHR/0082	*TABTBL/0083	SKP2/008C	*CURSTAT/0098	*TRANSP/0099
*SYSVARS/00D3	SYSCCI/00D5	SYSRST/00D8	SYSSWI/00DB	SYSPWR/00DE	*SYSMSK/00E0	*SYSCLK/00E1	SYSINT7/00E8
*SYSINT6/00EB	*SYSINT5/00EE	*SYSINT4/00F1	*SYSINT3/00F4	*SYSINT2/00F7	*SYSINT1/00FA	SYSINT0/00FD	SYSP6/00FE
RUNNINGENCRYPTED/00FF	*SCRSIZ/0800	*SCREEN/C000	BODTV/FC00	PUTFV/FC15	*GETFV/FC18	*DONFV/FC1B	*ODFFV/FC1E
GETCV/FC09	TSTCV/FC0C	*ESCCV/FC0F	INIFV/FC12	*DONLV/FC2D	*DOFFLV/FC30	INTDISV/FC33	*INTENV/FC36
*IDNFV/FC21	*IDFFFV/FC24	INILV/FC27	*PUTLV/FC2A	ESCC/FC42	ESCC/FC42	SYSINIT/FC4D	LEVEL0/FC7B
NDINT/FC39	*TSTMEM/FC3A	IDBNMI/FC3D	ESCC/FC42	ESCC/FC42	ESCC/FC42	SYSINIT/FC4D	LEVEL0/FC7B
FPITIM/FC8B	*FINT2/FC8E	HARDRESET/FC94	RESET9/FC9F	CLR/FCA9	LOOP/FCAF	NOTENCRYPT/FCB7	LEVEL7/FC7F
TOPLOOP/FCBC	RESET/FCD7	SYSLOOP/FCDE	SETBOOT/FCF4	SOFTRESET/FD04	SWIV/FD21	BOOT/FD25	RETRY/FD2B
COMDATA/FD41	TSTBOOT/FD46	READSECT/FD50	SETSECT/FD55	INI8IN/FD61	INITIALIZE/FD61		INISIN/FD74
CMD/FD85	FDWAIT/FD87	FDCMDX/FD92	RETURN/FD94	INIHWDFD95	DISKACKDONE/FDAB		ISMINI/FDAD
INIS2/FDB1	INISA/FDB4	TSTS1/FDBD	TSTSA/FDC0	GETS1/FDC7	GETSA/FDCA	GETS2/FDD5	*TSTS10/FDDA
TSTSA0/FDDD	PUTS1/FDE6	PUTSA/FDEB	PUTSB/FDEB	PUTSN/FDF5	PUTS1E/FDFB	INFON/FDFC	INFOFF/FDFF
OUTFON/FE02	OUTFOFF/FE05	PUTS2/FE0A	INIS1/FE11	TESTRAM/FE16	PTEST1/FE29	INIT/FE4B	INIT1/FE51
INIT2/FE54	CHKUP/FE63	CHKUP1/FE69	CHKUP2/FE6C	CHKDN/FE81	CHKDN1/FE82	DONE/FE9C	PUTHEX/FEA9
PUTHEX1/FEB1	PUTHEX2/FEB9	PRST/FEBF	PRINTSTR/FEC4	ERROR/FED1			

129 Symbols.

MAL/6800 1.3F: FFFE

12/18/82 12:03:57; Page 19; Form 9

Symbols Sorted by Value

JUPITERROM26.ASM

Error Lines:

JUPITERROM26.ASM

1-19 2-2

*** 2 Errors.