

[Iris]<wd0>MPCodes.bravo  
Last modified on August 30, 1979.  
Last editor was Maxion.

This file contains a list of the D0 Maintenance-Panel display codes. Since completion is an ongoing task, this file is incomplete by definition. Note that there are many names for the Maintenance-Panel. They are used interchangeably here though the correct name is indeed Maintenance-Panel, as defined by the *D0 Functional Specification*. In future documents and files the correct name should be used to reduce confusion.

From: [Iris]<Kennedy>Memo>DNPCodes.memo

The general D0 Maintenance-Panel display format is the following:

```
  X X X X
  | | | |
  | | | Subtest (error number)
  | | Test
  Program (FDP)
```

The most significant two digits indicate the functional program running at the time. For example:

<u>FDP</u>	<u>DNP (Most significant 2 digits)</u>
Navigator	00
Mechanic	01-05
MicroCode	07
Pilot	09
Firemen	10-19
Loadmaster	20-59

The next digit (tens) indicates the test running in the program. This number should relate to a system or subsystem within Star such as Processor, Memory, User Terminal, Disk, etc.

The least significant digit (units) indicates errors. If it is 0, the test is running and has found no errors. If it is non-zero (1 through 9), the test has failed and the number corresponds to the subtest that failed.

The full number should give an indication of progress or status in the system and errors, when found. The errors should be decoded through documentation and procedures to a Field Replaceable Unit and follow-on procedure.

A normal progression of codes would be, once the boot button has been pushed, 0000 through 0099, 0100 through 0590, 0700 through 0790, and then some code to indicate everything is successful and control is being transferred to Pilot. This number would be 08xx where the x's will be replaced with the microcode version, such as 0840 for microcode 4.0.

#### Reference Documentation:

- {A} [Iris]<wd0>Fault.mc
- {B} [Iris]<wd0>Initialize.mc
- {C} [Iris]<Kennedy>EPROM1ProcessChart.sil
- {D} [Iris]<wDiag>MechanicDNPCodes.memo
- {E} [Iris]<wPilot>HowToUse.memo
- {F} D0 Hardware Manual
- {G} [Iris]<Kennedy>Memo>DNPCodes.memo

{H} [Iris]<wDiag>Mechanic.Codes

0000 {E} PilotClient.run called  
 {C} Test Goto and clear maintenance panel

0001 {C} Test that ALU can be loaded with all zeroes; if Goto fails, output 0001  
 {C} Test that ALU can be loaded with all ones

0002 {C} Midas boot

0003 {???????} **Drive 0 not ready during push-button boot**  
 {C} Disk boot  
 {C} WakeUp test failed

0004 {C} Logical function (XOR) failure with zero in F-field & increment MP

0005 {C} Same as 0004 but source is T-memory

0006 {C} ALU logic function failure, alternating 1's (125) in low byte F-source

0007 {C} Same as 0006 but source is T-memory

0008 {C} ALU logic function failure, alternating 1's (252) in low byte F-source

0009 {C} Same as 0008 but source is T-memory

0010 {C} ALU logic function failure, alternating 1's (125) in high byte F-source

0011 {C} Same as 0010 but source is T-memory

0012 {C} ALU logic function failure, alternating 1's (252) in high byte F-source

0013 {C} Same as 0012 but source is T-memory

0014 {C} ALU logic function failure, all 1's (125) in high byte F-source

0015 {C} Same as 0014 but source is T-memory

0016 {C} Failure to write all register addresses into all registers using STACK

0017 {C} Failure to read all registers & compare addresses w/data using STACK

0018 {C} Testing completed

0091 {C} No header/label/data found on disk after 256 tries

0092 {C} Failure to find disk controller; clear D0 control store

0093 {C} Sector wakeup failure (after ten tries)

0094 {C} Compare failure with checksum read from last address vs CS checksum

0095 {C} Sector-2 wakeup failure (after ten tries), trying 256 sectors

0100 {D} Start of 'MECHANIC'  
 {B} **StartMapInit**

0101 {D} Bad 'cTask' number from NAVIGATOR  
 {B} **Not Enough Memory**

0102 {D} R Memory Access Error  
 {B} **Bad Map**

0103 {D} Stack Error

0110 {D} Can't find disk  
 {B} **StartDeviceInit**

0111 {D} M31, NOT READY

0112 {D} M31, Incorrect SEEK address

0113 {D} M31, Can't find sector

0114 {D} M31, Sector wakeup timeout

0115 {D} M31, Data timeout

0116 {D} M31, Countdown runout (Disk is HUNG)

0118 {D} M31, Can't find disk  
 {A} Not stack over/under flow (8 bits)

0119 {D} M31, NOT READY

0120 {D} M31, Incorrect SEEK address  
 {B} **StartDiskBoot**

0121 {D} M31, Can't find sector  
 {B} **No Disk Status**

- 0122 {D} M31, Sector wakeup timeout  
{B} **Bad Boot**
- 0123 {D} M31, Data timeout
- 0124 {D} M31, Countdown runout (Disk is HUNG)
- 0125 {D} **M31, Can't find disk 'LDTtask'**
- 0126 {D} M31, Can't read disk ID in specified task 'LDTtask'
- 0128 {A} R or CS parity error
- 0129 {A} Real breakpoint
- 
- 0130 {D} Start of control program  
{B} SystemRunning
- 0131 {D} Invalid error; return to control program from test program  
{A} H4 parity error
- 0132 {D} Bad 'cTask' when control returned to control program  
{A} Memory out of bounds and H4 parity error
- 0133 {A} Got MC2 error when unable to handle it by RETURNing
- 0134 {A} Two MC2 errors
- 0135 {A} MC1 fault when emulator couldn't accept it
- 0136 {A} Fault from the instruction following a LoadPage
- 0137 {A} Stack over/under flow
- 0138 {A} Map out of bounds
- 0139 {D} Unexpected main memory error
- 
- 0140 {D} Incorrect starting address, record 01 (see note 1)
- 0141 {D} Incorrect starting address, record 02 (see note 1)
- 0142 {D} Incorrect starting address, record 03 (see note 1)
- 0143 {D} Incorrect starting address, record 04 (see note 1)
- 0144 {D} Incorrect starting address, record 05 (see note 1)
- 0145 {D} Incorrect starting address, record 06 (see note 1)
- 0146 {D} Incorrect starting address, record 07 (see note 1)
- 0147 {D} Incorrect starting address, record 08 (see note 1)
- 0148 {D} Incorrect starting address, record 09 (see note 1)
- 0149 {D} Incorrect starting address, record 10 (see note 1)
- 
- 0150 {D} Incorrect starting address, record 11 (see note 1)
- 0151 {D} Incorrect starting address, record 12 (see note 1)
- 0152 {D} Incorrect starting address, record 13 (see note 1)
- 0153 {D} Incorrect starting address, record 14 (see note 1)
- 0154 {D} Incorrect starting address, record 15 (see note 1)
- 0155 {D} Incorrect starting address, record 16 (see note 1)
- 0156 {D} Incorrect starting address, record 17 (see note 1)
- 0157 {D} Incorrect starting address, record 18 (see note 1)
- 0158 {D} Incorrect starting address, record 19 (see note 1)
- 0159 {D} Incorrect starting address, record 20 (see note 1)
- 
- 0160 {D} Illegal wakeup # 0 (see note 3) (see note 3)
- 0161 {D} Unexpected wakeup # 1 (see note 3)
- 0162 {D} Unexpected wakeup # 2 (see note 3)
- 0163 {D} Unexpected wakeup # 3 (see note 3)
- 0164 {D} Unexpected wakeup # 4 (see note 3)
- 0165 {D} Unexpected wakeup # 5 (see note 3)
- 0166 {D} Unexpected wakeup # 6 (see note 3)
- 0167 {D} Unexpected wakeup # 7 (see note 3)
- 0168 {D} Unexpected wakeup # 8 (see note 3)
- 0169 {D} Unexpected wakeup # 9 (see note 3)
- 
- 0170 {D} Unexpected wakeup # 10 (see note 3)

0171 {D} Unexpected wakeup # 11 (see note 3)  
 0172 {D} Unexpected wakeup # 12 (see note 3)  
 0173 {D} Unexpected wakeup # 13 (see note 3)  
 0174 {D} Unexpected timer wakeup (see note 3)  
 0175 {D} Illegal wakeup # 15 (see note 3)  
 note 3)  
 0176 {D} Selected Load Device not available  
 0177 {D} Incorrect starting address, record 20A (see note 1)  
 0178 {D} Incorrect starting address, record 20B (see note 1)

0180 {D} Page 0: control got there incorrectly (see note 2)  
 0181 {D} Page 1: control got there incorrectly (see note 2)  
 0182 {D} Page 2: control got there incorrectly (see note 2)  
 0183 {D} Page 3: control got there incorrectly (see note 2)  
 0184 {D} Page 4: control got there incorrectly (see note 2)  
 0185 {D} Page 5: control got there incorrectly (see note 2)  
 0186 {D} Page 6: control got there incorrectly (see note 2)  
 0187 {D} Page 7: control got there incorrectly (see note 2)  
 0188 {D} Page 10: control got there incorrectly (see note 2)  
 0189 {D} Page 11: control got there incorrectly (see note 2)

0190 {D} Page 12: control got there incorrectly (see note 2)  
 0191 {D} Page 13: control got there incorrectly (see note 2)  
 0192 {D} Page 14: control got there incorrectly (see note 2)  
 0193 {D} Page 15: control got there incorrectly (see note 2)  
 0194 {D} Page 16: control got there incorrectly (see note 2)  
 0195 {D} Page 17: control got there incorrectly (see note 2)  
 0196 {D} Fault would NOT reset  
 0197 {D} R memory parity error  
 0198 {D} Control store parity error

0200 {D} Start of memory testing  
 0201 {D} Not enough memory  
 0202 {D} No GOOD memory card

0210 {D} Setting up map for operational microcode

0215 {D} Identity map subroutine; task # NOT ZERO on entry  
 0216 {D} Identity map subroutine; virtual = real test  
 0217 {D} Identity map subroutine; no good pages  
 0218 {D} Identity map subroutine; errors  
 0219 {D} Map exerciser error

0220 {D} Start of main memory testing

0240 {D} Start of extended main memory testing

0299 {D} Error data page initialization

0300 {D} Start of storage device equipment testing

0400 {D} Begin display equipment testing

0490 {D} IUTFP being tested  
 0491 {D} Address subtest error  
 0492 {D} Control register subtest error  
 0493 {D} Horizontal events subtest error  
 0494 {D} Data buffer subtest error

0495 {D} Wakeup logic subtest error

0500 {D} Begin communication equipment testing  
 0501 {D} Output device ID wrong  
 0502 {D} Not zero state & status read  
 0503 {D} State register not set  
 0504 {D} State not reset  
 0505 {D} Output task didn't task  
 0506 {D} Output task didn't wakeup  
 0507 {D} Output is hung up  
 0508 {D} Input or output is hanging  
 0509 {D} UnderRun not set

0510 {D} Output task ran  
 0511 {D} Input task ran again  
 0512 {D} Maximum number of soft errors  
 0513 {D} Ethernet test - False Error  
 0514 {D} Ethernet test took too long

0590 {D} Pre-initialization cleanup  
 0591 {D} Memory Refresh / D0 Buss Error  
 0599 {D} End of MECHANIC

0808 {F} R parity error  
 0880 {F} Control store parity error  
 0888 {F} Lamp test (while START is depressed)

0900 {E} Boot file being loaded by Germ  
 0901 {E} Germ alloc trap (bug)  
 0902 {E} Germ control trap (bug)  
 0903 {E} Germ start fault (bug)  
 0904 {E} Germ page or write-protect fault (bug)  
 0905 {E} Germ device error (e.g. hard disk error, Swatee too short)  
 0906 {E} Germ found bad boot file (e.g. wrong version of StartPilot)  
 0909 {E} Germ unnamed error (bug)

0910 {E} Mesa runtime being initialized  
 0912 {E} Trap or KFCB before Mesa runtime initialized (bug)

0919 {E} Attempt to swap to CoPilot before Mesa runtime initialized (bug)

0920 {E} Storage drivers being initialized  
 0930 {E} Filer being initialized  
 0940 {E} Swapper being initialized  
 0950 {E} FileMgr being initialized  
 0960 {E} VMMgr being initialized  
 0970 {E} Client and other non-boot loaded code being mapped  
 0980 {E} Temporary files being deleted  
 0990 {E} Communication being initialized

Notes:

1. MECHANIC is composed of a series of records read from the Input Load Device (currently, only the Diablo Model 31 disk unit). These records are currently identified as:

x0a.mc, x0b.mc, x0c.mc, x.mc (one record; 4 source files)  
x1a.mc, x1b.mc, x1c.mc, x1.mc (one record; 4 source files)  
x2a.mc, x2.mc (one record; 2 source files)  
x3a.mc, x3.mc (one record; 2 source files)  
x4.mc  
x5.mc  
x6.mc  
x20.mc

Records x7 through x19 are reserved for future expansion.

Each record resident on the disk will have a predetermined control store starting address that will be checked by the microcode performing the load operation. Should there be a mismatch, the microcode performing the load operation will ABORT the boot operation and report an 'incorrect starting address' in the DNP.

The occurrence of a mismatch is a critical error. It is likely a disk-related error since MECHANIC was loaded successfully by NAVIGATOR.

2. Unused locations of control store have been loaded with a microinstruction sending control to a common point that will ABORT the boot operation and report in the MP that control got to an invalid point in the control store. This response to such an error has been further isolated to such errors detected on a single page (e.g., there are sixteen MP codes assigned to indicating when such an error occurs on each of the sixteen control store pages).

The occurrence of such an error is critical. Control has somehow landed outside MECHANIC. Suspicion should be directed at the processor and control store boards. Experience has shown this error can occur infrequently on different equipments with certain equipment exhibiting both more regularity and variety. A re-boot has to date always been accomplished successfully.

3. Early in MECHANIC microcode is executed that prepares MECHANIC to handle unexpected wakeups. Should an unexpected wakeup occur, MECHANIC will report its occurrence and ABORT the boot operation. Note also that the hardware reporting a wakeup of '0' (emulator task), '14' (timer), or '15' (fault handler) will be distinguished.