

```

1 *****
2 *
3 *           DBStatus - An AppleWorks Init
4 *
5 *           - Adds Real Time File Status Indicator to
6 *             Data Base REVIEW/ADD/CHANGE Window -
7 *
8 *           - 'New' or 'Changed' File Indicated by Mousetext
9 *             Diamond Character in place of colon after
10 *            'File' in top line of Display.
11 *
12 *           Version 2.3 (for AppleWorks Version 5.1)
13 *             (c) 2015 Hugh Hood
14 *
15 *           - Seg $02 is patch-in segment, but NEW code requires
16 *             space at end of Seg $01 (Main), because placing
17 *             code at either $BB00 (default) or at most other
18 *             locations results in it being overwritten by
19 *             newly-loaded Segments. By moving the new code
20 *             to the Main DB Segment patch, we ensure that it
21 *             is always present whenever Data Base is the
22 *             active application, since Seg $01 must be loaded.
23 *
24 *****
25
26             TR             ADR             ; truncate bank address
27
28             XC             ; enable 65C02 code
29
30 * Equates *
31
32 CurRec      EQU           $B4             ; current data base record # (word)
33 MTDiamond   EQU           $DB            ; mousetext 'diamond' character
34 DTFStatus   EQU           $C6C          ; current file status flag byte:
35             ; ($01/0001) or $03/0011) - new
36             ; ($02/0010) or $06/0110) - changed
37             ; ($04/0100) - saved
38             ; ($00/0000) - unchanged
39 AVersion    EQU           $1003          ; $33/51 = 5.1 / $28/40 = 4.0 /
40             ; $1E/30 = 3.0
41 MvLeftRtn   EQU           $1148         ; memory move / follow with TO/FROM/LENGTH
42 StrWrRtn    EQU           $116F         ; writes a string at a fixed location
43             ; on the screen
44             ; (follow JSR with col/row/string address)
45 imSavePatch EQU           $3006          ; Patch Manager save routine in SEG.IM
46 InitAdr     EQU           $4000         ; load address for Init files
47 PatchPoint  EQU           $251A        ; patch point in SEG $02
48             ; (routine to print current record
49             ; # on screen)
50 Patch2Run   EQU           $91D0         ; final destination for new code
51             ; NOTE: Seg $02 runs from $2000 - $5105.
52             ; Other Data Base Segs load at
53             ; $3900, $3D00, $4000, $4100,
54             ; $4B00, $5200 and $5A00
55             ; - Seg $01 runs from $7800-$91B5
56             ; - DB work area starts $9200

```

```

57 PatchAdr      EQU          $BB00      ; load address for patch code
58                                     ; (NOTE: uses ProDOS I/O buffer -
59                                     ;           1K max length -
60                                     ;           $BB00 - $BEFF)
61 Word2Str      EQU          $D044      ; convert Hex word to Decimal string
62
63
64                                     ORG          InitAdr      ; ($4000)
65                                     TYP          $06          ; create binary file
66
67 *****
68 *           Init Header           *
69 *****
70 START
71                                     JMP          IStart      ; skip over header
72
73 **-----
74
75                                     ASC          'mb'          ; Init ID Bytes (AW 5.1)
76                                     DB           $17          ; Init Version - programmer assigned
77                                     ; e.g. - $0A/1.0 $0B/1.1 $17/2.3
78                                     STR          'DBStatus'      ; Init Screen Name
79                                     HEX          0000          ; Header End Bytes
80
81 **-----
82
83 IStart
84
85                                     LDA          AWVersion      ; AppleWorks version #
86                                     CMP          #$33          ; Is it Version 5.1?
87                                     BNE          Done          ; disregard - wrong version
88
89
90 PatchH02      JSR          imSavePatch ; call patch manager
91                                     DW          Code1          ; beginning of patch1 code ($40xx)
92                                     DW          Patch1End-PatchAdr
93 ; length of patch code
94                                     DW          $0002          ; SEG number to patch
95                                     ; ($02 = ADB Edit SEG)
96
97 PatchH01      JSR          imSavePatch ; call patch manager
98                                     DW          Code1+Patch1End-PatchAdr
99 ; beginning of patch2 code ($40xx)
100                                     DW          MoveStart-PatchAdr+Patch2End-Patch2Run
101 ; length of patch code
102                                     DW          $0001          ; SEG number to patch
103                                     ; ($01 = ADB Main SEG)
104
105 Done          RTS
106                                     ; back to Init Manager
107
108 **-----
109
110 Code1         EQU          *           ; (will be $40xx)
111
112                                     ORG          PatchAdr      ; (Patching Code is moved and run

```

```

110                                     ; @ $BB00 by Init Manager)
111
112 HookBytes      HEX      0000      ; first bytes for $02 Patch
113               LDA      #$4C      ; JMP instruction
114               STA      PatchPoint ; $251A in SEG $02
115               LDA      #Patch2Run ; low byte of new code
116               STA      PatchPoint+1
117               LDA      #>Patch2Run ; high byte of new code
118               STA      PatchPoint+2
119
120               RTS
121
122 Patch1End      EQU      *
123
124 **-----
125
126               ORG      PatchAdr    ; (Patching Code is moved and run
127                                     ; @ $BB00 by Init Manager)
128
129 MoveCall       HEX      0000      ; first bytes for $01 Patch
130               JSR      MvLeftRtn  ; move new code to run location
131               DA      #Patch2Run  ; ($91D0)
132               DA      #MoveStart  ;
133               DA      Patch2End-Patch2Run
134
135               RTS                ; patch hook-in done
136
137 **-----
138
139 MoveStart      EQU      *          ; (will be $BBxx)
140
141               ORG      Patch2Run  ; ($91D0)
142
143 PatchStart     LDA      #'::'     ; unchanged 'File' suffix
144               STA      StatusStr+1 ;
145               LDA      DTFStatus  ; current file status flag byte
146               AND      #%00000011 ; test if either Bit 0 or Bit 1 is set
147                                     ; to check for unchanged or saved
148               BEQ      :A         ; if both bits are clear, branch around
149               LDA      #MTDiamond ; mousetext 'diamond' suffix
150               STA      StatusStr+1 ;
151 :A             JSR      StrWrRtn  ;
152               DB      $04        ; column $04/04
153               DB      $00        ; row $00/00
154               DA      StatusStr  ; either colon or MTDiamond character
155               JSR      Word2Str   ; originally from $251A-251C
156               DA      CurRec     ; originally from $251D-241E
157
158               JMP      PatchPoint+5
159                                     ; go back to original code at $251F
160
161 **-----
162
163 StatusStr      Str      '::'     ; default 'File' suffix
164
165 **-----

```

```
166
167 Patch2End EQU * ;
168 SAV I.DBSTATUS
169 LST OFF
170
171 END
172
```