

```

1 *****
2 *
3 *           MaxDesk - An AppleWorks Init           *
4 *
5 *   - When quitting AppleWorks running under      *
6 *     ProDOS 8 on the Apple IIGS, the             *
7 *     MaxDesk Init disposes of the memory        *
8 *     allocated by the Apple IIGS Memory         *
9 *     Manager to the AppleWorks Desktop         *
10 *     so that restarting AppleWorks in a        *
11 *     single session does not result in         *
12 *     a diminished Desktop size.               *
13 *     (approximately 230kB reduction)           *
14 *
15 *           - BACKGROUND -                       *
16 *
17 *   Under ProDOS 8 (no tool patches) the         *
18 *   DisposeAll tool does not seem reliably to   *
19 *   work if the assigned Application Memory has *
20 *   an Aux ID of 1 (the second digit in $11xx), *
21 *   unless an Aux ID of 0 (wildcard) is used in *
22 *   the DisposeAll call. This Init patches     *
23 *   AppleWorks to use an Aux ID of 0 when quitting, *
24 *   thus correcting the problem.                *
25 *
26 *
27 *   - NOTE: This init is not required when running *
28 *     AppleWorks under GS/OS, but its           *
29 *     presence in the AW.INITS folder           *
30 *     causes no harm.                         *
31 *
32 *           Version 1.0 (for AppleWorks Version 5.1) *
33 *           (c) 2015 Hugh Hood                  *
34 *
35 *   - Seg $29 is patch-in segment, but only the exiting *
36 *     code is modified. No NEW code is required. *
37 *
38 *
39 *****
40
41           TR           ADR           ; truncate bank address
42
43           XC           ; enable 65C02 code
44           XC           ; enable 65816 code
45
46 * Equates *
47
48 MemoryID      EQU      $0FDE      ; Memory ID# stored here by AppleWorks
49 Seg00Type     EQU      $FE0       ; memory manager type
50                                     ; $41 = SEG.AM / $52 = SEG.RM (IIGS) /
51                                     ; $58 = SEG.XM / $4D = Deja IIX (Mac)
52 AWVersion     EQU      $1003      ; $33/51 = 5.1 / $28/40 = 4.0 /
53                                     ; $1E/30 = 3.0
54 DisposeAll    EQU      $1102      ; DisposeAll Tool for Memory Manager
55 MvLeftRtn     EQU      $1148      ; memory move / follow with TO/FROM/LENGTH
56 DeleteID      EQU      $2103      ; DeleteID Tool for Memory Manager

```

```

57 imSavePatch    EQU          $3006    ; Patch Manager save routine in SEG.IM
58 InitAdr        EQU          $4000    ; load address for Init files
59 PatchPoint     EQU          $65D0    ; patch point in SEG $29
60                ; (routine to Dispose of memory allocated
61                ;   by Apple IIGS Memory Manager)
62 PatchAdr       EQU          $BB00    ; load address for patch code
63                ; (NOTE: uses ProDOS I/O buffer -
64                ;   1K max length -
65                ;   $BB00 - $BEFF)
66 ToolLocate     EQU          $E10000   ; GS Tool Locator Call
67
68                ORG          InitAdr   ; ($4000)
69                TYP          $06      ; create binary file
70
71 *****
72 *           Init Header           *
73 *****
74 START
75                JMP          IStart    ; skip over header
76
77 **-----
78
79                ASC          'mb'      ; Init ID Bytes (AW 5.1)
80                DB           $0A      ; Init Version - programmer assigned
81                ; e.g. - $0A/1.0 $0B/1.1 $17/2.3
82                STR          'MaxDesk' ; Init Screen Name
83                HEX          0000     ; Header End Bytes
84
85 **-----
86
87 IStart
88
89                LDA          AVersion  ; AppleWorks version #
90                CMP          #$33     ; Is it Version 5.1?
91                BNE          Done      ; disregard - wrong version
92
93                LDA          Seg00Type ; Memory Manager Type
94                CMP          #$52     ; Is it Apple IIGS? (SEG.RM)
95                BNE          Done      ; disregard - IIGS only
96
97 PatchH29       JSR          imSavePatch ; call patch manager
98                DW          Code1     ; beginning of patch1 code ($40xx)
99                DW          MoveStart-PatchAdr+Patch1End-PatchPoint
... ; length of patch code
100                DW          $0029    ; SEG number to patch
101                ; ($29 = Organizer Save/Remove SEG)
102
103 Done           RTS                    ; back to Init Manager
104
105 **-----
106
107 Code1          EQU          *          ; (will be $40xx)
108
109                ORG          PatchAdr  ; (Patching Code is moved and run
110                ; @ $BB00 by Init Manager)
111

```

```

112 HookBytes      HEX      0000      ; first bytes for Patch
113              JSR      MvLeftRtn  ; move new code to run location
114              DA      #PatchPoint ; ($65D0)
115              DA      #MoveStart  ;
116              DA      Patch1End-PatchPoint
117
118              RTS                      ; patch done
119
120 **-----
121
122 MoveStart      EQU      *              ; (will be $BBxx)
123
124              ORG      PatchPoint     ; ($65D0)
125
126              MX      %00             ; 16-bit ACC and Registers
127
128 PatchStart    AND      #$F0FF        ; force Aux ID# to 0
129              STA      MemoryID       ; store modified Aux ID#
130              PHA                      ; push the ID# on the stack
131              NOP                      ; kills an unused patched byte
132              LDX      #DisposeAll    ; ready DisposeAll Tool
133              JSL      ToolLocate     ; Tool Locator call
134              LDA      MemoryID       ; modified by AND
135              PHA                      ; push the ID# on the stack
136              LDX      #DeleteID     ; ready DeleteID Tool
137
138 **-----
139
140              MX      %11             ; 8-bit ACC and Registers
141
142 Patch1End     EQU      *              ;
143              SAV      I.MAXDESK
144              LST      OFF
145
146              END
147

```