

```

*****
*
*          SSStatus - An AppleWorks Init
*
*      - Adds Real Time File Status Indicator to
*        Spreadsheet REVIEW/ADD/CHANGE Window -
*
*      - 'New' or 'Changed' File Indicated by Mousetext
*        Diamond Character in place of colon after
*        'File' in top line of Display.
*
*          Version 2.3 (for AppleWorks Version 5.1)
*          (c) 2015 Hugh Hood
*
*      - Seg $1E is patch-in segment, but NEW code requires
*        space at end of Seg $1D (Main), because placing
*        code at either $BB00 (default) or at most other
*        locations results in it being overwritten by
*        newly-loaded Segments. By moving the new code
*        to the Main SS Segment patch, we ensure that it
*        is always present whenever SpreadSheet is the
*        active application, since Seg $1D must be loaded.
*
*****

```

```

TR      ADR      ; truncate bank address

```

```

XC      ; enable 65C02 code

```

```

* Equates *

```

```

MTDiamond EQU $DB      ; mousetext 'diamond' character
DTFStatus EQU $C6C     ; current file status flag byte:
                        ; ($01/0001) or $03/0011) - new
                        ; ($02/0010) or $06/0110) - changed
                        ; ($04/0100) - saved
                        ; ($00/0000) - unchanged
LJCellID  EQU $802C     ; used by SS to contain CellID (e.g. 'A1')
AWVersion EQU $1003     ; $33/51 = 5.1 / $28/40 = 4.0 /
                        ; $1E/30 = 3.0
MvLeftRtn EQU $1148     ; memory move / follow with TO/FROM/LENGTH
StrWrRtn  EQU $116F     ; writes a string at a fixed location
                        ; on the screen
                        ; (follow JSR with col/row/string address)
imSavePatch EQU $3006   ; Patch Manager save routine in SEG.IM
InitAdr    EQU $4000     ; load address for Init files
PatchPoint EQU $3D5E     ; patch point in SEG $1E
Patch2Run  EQU $7ED0     ; final destination for new code
                        ; NOTE: Seg $1E runs from $3900 - $4EFE,
                        ;       but seems to use memory up through
                        ;       $51FF - Other SpreadSheet Segs
                        ;       load at $5300, $7400 and $2100
                        ;       - not even $5200 is free for long -
                        ;       - Seg $1D runs from $7400-$7EC3
                        ;       - SS work area starts $7F00
PatchAdr   EQU $BB00     ; load address for patch code
                        ; (NOTE: uses ProDOS I/O buffer -
                        ;       1K max length -
                        ;       $BB00 - $BEFF)

```

```

                ORG     InitAdr      ; ($4000)
                TYP     $06          ; create binary file

*****
*           Init Header           *
*****
START
                JMP     IStart       ; skip over header

**-----

                ASC     'mb'         ; Init ID Bytes (AW 5.1)
                DB      $17          ; Init Version - programmer assigned
                                   ; e.g. - $0A/1.0 $0B/1.1 $17/2.3
                STR     'SSStatus'   ; Init Screen Name
                HEX     0000         ; Header End Bytes

**-----

IStart

                LDA     AWVersion    ; AppleWorks version #
                CMP     #$33         ; Is it Version 5.1?
                BNE     Done         ; disregard - wrong version

PatchH1E       JSR     imSavePatch  ; call patch manager
                DW      Code1        ; beginning of patch1 code ($40xx)
                DW      Patch1End-PatchAdr
                                   ; length of patch code
                DW      $001E        ; SEG number to patch
                                   ; ($1E = ASP Edit SEG)

PatchH1D       JSR     imSavePatch  ; call patch manager
                DW      Code1+Patch1End-PatchAdr
                                   ; beginning of patch2 code ($40xx)
                DW      MoveStart-PatchAdr+Patch2End-Patch2Run
                                   ; length of patch code
                DW      $001D        ; SEG number to patch
                                   ; ($1D = ASP Main SEG)

Done           RTS                 ; back to Init Manager

**-----

Code1          EQU     *           ; (will be $40xx)

                ORG     PatchAdr     ; (Patching Code is moved and run
                                   ; @ $BB00 by Init Manager)

HookBytes      HEX     0000         ; first bytes for $1E Patch
                LDA     #$4C         ; JMP instruction
                STA     PatchPoint   ; $3D5E in SEG $1E
                LDA     #Patch2Run   ; low byte of new code
                STA     PatchPoint+1
                LDA     #>Patch2Run
                                   ; high byte of new code
                STA     PatchPoint+2

```

```

                RTS

Patch1End      EQU      *

**-----

                ORG      PatchAdr      ; (Patching Code is moved and run
                ;        @ $BB00 by Init Manager)

MoveCall       HEX      0000          ; first bytes for $1D Patch
                JSR      MvLeftRtn     ; move new code to run location
                DA       #Patch2Run    ; ($7ED0)
                DA       #MoveStart    ;
                DA       Patch2End-Patch2Run

                RTS                  ; patch hook-in done

**-----

MoveStart      EQU      *              ; (will be $BBxx)

                ORG      Patch2Run     ; ($7ED0)

PatchStart     LDA      #':'          ; unchanged 'File' suffix
                STA      StatusStr+1   ;
                LDA      DTFStatus     ; current file status flag byte
                AND      #%00000011    ; test if either Bit 0 or Bit 1 is set
                ;                    ; to check for unchanged or saved
                BEQ      :A            ; if both bits are clear, branch around
                LDA      #MTDiamond    ; mousetext 'diamond' suffix
                STA      StatusStr+1   ;
:A             JSR      StrWrRtn       ;
                DB       $04           ; column $04/04
                DB       $00           ; row $00/00
                DA       StatusStr     ; either colon or MTDiamond character
                JSR      StrWrRtn     ; originally from $3D5E-3D60
                DB       $00           ; column $00/00 {was $3D61}
                DB       $15           ; row $15/21 {was $3D62}
                DA       LJCellID      ; used by SS to contain CellID (e.g. 'A1')
                ;                    ; {was $3D63-3D64}

                JMP      PatchPoint+7
                ; go back to original code

**-----

StatusStr      Str      ':'          ; default 'File' suffix

**-----

Patch2End      EQU      *              ;
                SAV      I.SSSTATUS
                LST      OFF

                END

```