```
*************************************************************
*                                                           *
*                 WPStatus - An AppleWorks Init             *
*                                                           *
*         - Adds Real Time File Status Indicator and        *
*            Total Number of Lines Indicator to             *
*           Word Processor REVIEW/ADD/CHANGE Window -       *
*                                                           *
*     - 'New' or 'Changed' File Indicated by Mousetext      *
*         Diamond Character in place of colon after         *
*               'File' in top line of Display.              *
*                                                           *
*        - Total Number of Lines in Word Processor          *
*           Document is displayed after the filename        *
*           in top line of Display.                         *
*                                                           *
*           Version 2.4 (for AppleWorks Version 5.1)        *
*                     (c) 2015 Hugh Hood                    *
*                                                           *
*     - Seg $16 is patch-in segment, but NEW code requires  *
*        space at end of Seg $19 (Spell Check), because     *
*        code at either $BB00 (default) or at most other    *
*        locations can result in it being overwritten by    *
*        newly-loaded Segments. By moving the new code      *
*        to the Main WP Segment patch, we ensure that it    *
*        is always present whenever Word Processor is the   *
*        active application, since Seg $15 must be loaded.  *
*                                                           *
*************************************************************

              TR     ADR          ; truncate bank address

              XC                  ; enable 65C02 code

* Equates *

PtrBase       EQU    $1E          ; AuxMem address of pointer containing
                                  ;  first line in file
                                  ;   (normally = $4000)
MTRightBar    EQU    $DA          ; mousetext 'right bar' character
MTDiamond     EQU    $DB          ; mousetext 'diamond' character
MTLeftBar     EQU    $DF          ; mousetext 'left bar' 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
OldStr        EQU    $E05         ; used by WP to contain 'Line/Column' String
StrWork5      EQU    $FD7         ; 5-byte string used by newWord2Str
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    $4393        ; patch point in SEG $16
```

```
NextDSP       EQU    $7C0C       ; AuxMem address of pointer containing
                                 ;  NEXT AVAILABLE line in WP file
                                 ;  (e.g. - 1 address past pointer to
                                 ;    LAST line in WP file)
Patch2Run     EQU    $A880       ; final destination for new code
                                 ; NOTE: Seg $16 runs from $2100 - $4B89,
                                 ;       but other Segs use adjacent memory
                                 ;       up through $6DC3, leaving only
                                 ;       $3C (60) bytes free before the
                                 ;       Main WP Seg ($15) starts at $6E00.
newCC2S       EQU    $AB40       ; concatenate (2) strings
                                 ;  (first String has second added
                                 ;    at end)
newWord2Str   EQU    $AB61       ; convert HEX word to decimal string
                                 ;  (result in StrWork5) [5 character max]
                                 ;  (result is left justified)
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     $18         ; Init Version - programmer assigned
                                 ;  e.g. - $0A/1.0 $0B/1.1 $18/2.4
              STR    'WPStatus24'
              ; 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

PatchH16      JSR    imSavePatch ; call patch manager
              DW     Code1       ; beginning of patch1 code ($40xx)
              DW     Patch1End-PatchAdr
              ; length of patch code
              DW     $0016       ; SEG number to patch
                                 ;  ($16 = AWP Edit SEG)


PatchH15      JSR    imSavePatch ; call patch manager
              DW     Code1+Patch1End-PatchAdr
              ; beginning of patch2 code ($40xx)
```

```
              DW     MoveStart-PatchAdr+Patch2End-Patch2Run
              ; length of patch code
              DW     $0015       ; SEG number to patch
                                 ;  ($15 = AWP 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 $16 Patch
              LDA    #$4C        ; JMP instruction
              STA    PatchPoint  ; $4393 in SEG $16
              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 $15 Patch
              JSR    MvLeftRtn   ; move new code to run location
              DA     #Patch2Run  ; ($6DD0)
              DA     #MoveStart  ;
              DA     Patch2End-Patch2Run

              RTS                ; patch hook-in done

**-------------------------------------------------------------

MoveStart     EQU    *           ; (will be $BBxx)

              ORG    Patch2Run   ; ($6DD0)

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 $4393-$4395
```

```
                DB      $26         ; column $26/37 {was $4396}
                DB      $17         ; row $17/23 {was $4397}
                DA      OldStr      ; used by WP to contain 'Line/Column' String
                                    ;  {was $4398-$4399}
*
                LDA     NextDSP     ; next new line pointer (LSB)
                SEC                 ; prepare for subtraction
                SBC     PtrBase     ; subtract first line pointer (LSB)
                STA     NumLines    ; store result (LSB) X2
                LDA     NextDSP+1   ; next new line pointer (MSB)
                SBC     PtrBase+1   ; subtract first line pointer (MSB)
                LSR                 ; divide result (MSB) by (2)
                                    ;  (bit 0 to carry)
                STA     NumLines+1  ; store result (MSB)
                ROR     NumLines    ; divide result (LSB) by (2)
                                    ;  (carry to bit 7)
                CLC                 ; reset carry
                JSR     newWord2Str ; convert HEX word to decimal string
                DA      NumLines    ; address of word to convert
                STZ     LinesStr    ; initialize string to be printed
                JSR     newCC2S     ; concatenate (2) strings
                DA      LinesStr
                DA      LeftBarStr
                JSR     newCC2S     ; concatenate (2) strings
                DA      LinesStr
                DA      StrWork5
                JSR     newCC2S     ; concatenate (2) strings
                DA      LinesStr
                DA      RightBarStr
                JSR     newCC2S     ; concatenate (2) strings
                DA      LinesStr
                DA      SpaceStr
                LDA     #$7         ; maximum LinesStr length
                STA     LinesStr    ; truncate to max length
                JSR     StrWrRtn    ; display number of lines at top
                DB      $17         ; column $17/23
                DB      $00         ; row $00/00
                DA      LinesStr    ;

                JMP     PatchPoint+7
                ; go back to original code

**--------------------------------------------------------------

StatusStr       Str     ':'         ; default 'File' suffix

**--------------------------------------------------------------

NumLines        DS      2           ; space for calculated result

**--------------------------------------------------------------

SpaceStr        Str     '    '      ; string for pad spaces
LeftBarStr      Str     #MTLeftBar  ; string for Left Bar
RightBarStr     Str     #MTRightBar ; string for Right Bar
```

```
LinesStr      DS     12          ; space for length byte +
                                 ;  (7) characters max + (4)
                                 ;  'space' characters max to pad

**-------------------------------------------------------------

Patch2End     EQU    *           ;
              SAV    I.WPSTATUS24
              LST    OFF

              END
```