```
;====================================================================
====================================
;
VIC II REGISTER INCLUDE FILE
;====================================================================
====================================
;
Peter 'Sig' Hewett aka Retroromicon
;
- 2016
;====================================================================
====================================

;--------------------------------------------------------------------
VIC II REGISTERS
VIC_SPRITE_X_POS        =  $D000        ; Increment by 2 bytes
for the next sprite x pos
VIC_SPRITE_Y_POS        =  $D001        ; Increment by 2 bytes
for the next sprite y pos
VIC_SPRITE_X_EXTEND     =  $D010        ; Bits #0 - 7 : Extended
X bit for sprites 0-7

VIC_SCREEN_CONTROL_Y    =  $D011
VIC_SCREEN_CONTROL      =  $D011        ; Screen Control
Register 1
                                        ; Bits #0-#2: Vertical
raster scroll
                                        ; Bit #3 : Screen Height
0 = 24 rows 1 = 25 rows
                                        ; Bit #4 : 0 = Screen
off  1 = Screen on (normal function)
                                        ; Bit #5 : 0 = Text Mode
; 1 = Bitmap mode
                                        ; Bit #6 : 1 = Extended
Background Mode on
                                        ; Bit #7 : Read current
raster line (bit #8)
                                        ;          Write: Raster
line to generate interrupt at (bit #8)
                                        ;

VIC_RASTER_LINE         =  $D012        ; Read: Current raster
line (bits #0-#7)
                                        ; Write: Raster line to
generate interrupt at (bits #0-#7).
```

```
VIC_SPRITE_ENABLE       =  $D015        ; (53269) set bits 0-8
to enable repective sprite

VIC_SCREEN_CONTROL_X    =  $D016
VIC_CONTROL             =  $D016        ; Screen Control
Register 2
                                        ; Bits #0-#2 :
Horizontal raster scroll
                                        ; Bit #3 : Screen Width
; 0 = 38 cols 1 = 40 cols
                                        ; Bit #4 : 1 =
Multicolor mode on
                                        ; Default : $C8
(%11001000)

VIC_MEMORY_CONTROL      =  $D018        ; MEMORY SETUP REGISTER
BITS
                                        ; Bits #1-#3 in text
mode are a pointer to character memory relative to VIC_BANK
($DD00)
                                        ;       %000, 0 : $0000-
$07FF
                                        ;       %001, 1 : $0800-
$0FFF
                                        ;       %010, 2 : $1000-
$17FF
                                        ;       %011, 3 : $1800-
$1FFF
                                        ;       %100, 4 : $2000-
$27FF
                                        ;       %101, 5 : $2800-
$2FFF
                                        ;       %110, 6 : $3000-
$37FF
                                        ;       %111, 7 : $3800-
$3FFF
                                        ; Values %010 and %011
in VIC Bank #0 and #2 select Character ROM instead
                                        ;
                                        ; In bitmap mode,
pointer to bitmap memory (bit #13) relative to VIC_BANK address
($DD00)
                                        ;       %0xx, 0 : $0000-
$1FFFF
                                        ;       %1xx, 4 : $2000-
$3FFFF
                                        ;
```

```
                                        ; Bits #4-#7 : Pointer
to screen memory (bits #10-#13) relative to VIC_BANK address
($DD00)
                                        ;       %0000, 0  :
$0000 - $03FF
                                        ;       %0001, 1  :
$0400 - $07FF
                                        ;       %0010, 2  :
$0800 - $0BFF
                                        ;       %0011, 3  :
$0C00 - $0FFF
                                        ;       %0100, 4  :
$1000 - $13FF
                                        ;       %0101, 5  :
$1400 - $17FF
                                        ;       %0110, 6  :
$1800 - $1BFF
                                        ;       %0111, 7  :
$1C00 - $1FFF
                                        ;       %1000, 8  :
$2000 - $23FF
                                        ;       %1001, 9  :
$2400 - $27FF
                                        ;       %1010, 10 :
$2800 - $2BFF
                                        ;       %1011, 11 :
$2C00 - $2FFF
                                        ;       %1100, 12 :
$3000 - $33FF
                                        ;       %1101, 13 :
$3400 - $37FF
                                        ;       %1110, 14 :
$3800 - $3BFF
                                        ;       %1111, 15 :
$3C00 - $3FFF

VIC_INTERRUPT_CONTROL   =  $D01A        ; (53274) Interrupt
control register
                                        ;  Bit #0 1 = Raster
interupt enabled
                                        ;  Bit #1 1 = Sprite -
Background collision interupt enabled
                                        ;  Bit #2 1 = Sprite -
Sprite collision interrupt enabled
                                        ;  Bit #3 1 = Light pen
interrupt enabled
```

```
VIC_SPRITE_MULTICOLOR   =  $D01C        ; (53276) Sprite
Multicolor mode register
                                        ;  #bit - set individual
sprites to multicolor
                                        ;  Bit #x = 0 - SpriteX
is single color 1 = Sprite x is multicolor
                                        ;

VIC_BORDER_COLOR        =  $D020        ; (53280) Border color
VIC_BACKGROUND_COLOR    =  $D021        ; (53281) Background
color
VIC_CHARSET_MULTICOLOR_1 = $D022        ; (53282) Extra
Background Color 1 - Multicolor 1
VIC_CHARSET_MULTICOLOR_2 = $D023        ; (53283) Extra
Background Color 2 - Multicolor 2
VIC_CHARSET_MULTICOLOR_3 = $D024        ; (53284) Extra
Background Color 3
VIC_SPRITE_MULTICOLOR_1 =  $D025        ; (53285) Sprite extra
color 1
VIC_SPRITE_MULTICOLOR_2 =  $D026        ; (53286) Sprite extra
color 2
VIC_SPRITE_COLOR        =  $D027        ; (53287) Sprite Color
($D027 - $D02E = Sprites 0 - 7)

;-----------------------------------------------------------------
------------------ CIA REGISTERS
PORT_A                  = $DC00         ; CIA PORT A - Joystick
#2
JOY_2                   = $DC00         ; Keyboard Matrix
columns and Joystick #2
                                        ; Read Bits:
                                        ;       Bit #0 0 = Port
2 Joystick up pressed
                                        ;       Bit #1 0 = Port
2 Joystick down pressed
                                        ;       Bit #2 0 = Port
2 Joystick left pressed
                                        ;       Bit #3 0 = Port
2 Joystick right pressed
                                        ;       Bit #4 0 = Port
2 Joystick fire pressed
                                        ; Write Bits:
                                        ;       Bit #x : 0 =
Select keyboard matrix column #x
                                        ;       Bit #6-#7 :
Padle selection %01 = Paddle 1; #10 = Paddle #2
```

```
INT_CONTROL             = $DC0D         ; Interrupt control and
status register
                                        ; Read Bits     #0 - 1 =
Timer A underflow occurred
                                        ;               #1 - 1 =
Timer B underflow occurred
                                        ;               #2 - 1 =
TOD is equal to alarm time
                                        ;               #3 - 1 =
A complete byte has been received into or
                                        ;
sent from serial shift register
                                        ;               #4 -
Signal level on FLAG pin, datasette input
                                        ;               #7 - An
interrupt has been generated
                                        ;
                                        ; Write Bits    #0 - 1 =
Enable interrupts generated by timer A underflow
                                        ;               #1 - 1 =
Enable interrupts generated by timer B underflow
                                        ;               #2 - 1 =
Enable TOD alarm interrupt
                                        ;               #3 - 1 =
Enable interrupts generated by a byte having been
                                        ;
recieved/sent via serial shift register
                                        ;               #4 - 1 =
Enable interrupts generated by positivy edge on FLAG pin
                                        ;               #7 -
Fill bit ; bits #0-#6 that are set to 1, get their values from
                                        ;
this bit; bits #0-#6, that are set to 0, are left unchanged

VIC_BANK                = $DD00
CIA_PRA                 = $DD00         ; CIA#2 - PORT_A, serial
bus access
                                        ; Bits #0-#1 : VIC bank
values
                                        ;       %00 - Bank #3 -
$C000 - $FFFF
                                        ;       %01 - Bank #2 -
$8000 - $BFFF
                                        ;       %10 - Bank #1 -
$4000 - $7FFF
```

```
                                        ;       %11 - Bank #0 -
$0000 - $3FFF
                                        ; Bit #2 - RS232 TXD
line, output bit
                                        ; Bit #3 - Serial bus
ATN OUT; 0 = high; 1 = low
                                        ; Bit #4 - Serial bus
CLOCK OUT; 0 - high; 1 = low
                                        ; Bit #5 - Serial bus
DATA OUT; 0 - low ; 1 = high
                                        ; Bit #6 - Serial bus
CLOCK IN; 0 = low; 1 = high
                                        ; Bit #7 - Serial bus
DATA IN; 0 = low; 1 = high

;---------------------------------------------------------------
-----------------------------------
;
ZERO PAGE
;---------------------------------------------------------------
-----------------------------------
PROC_PORT       = $0001         ; Bits #0 - #2 : Configuration
for memory areas $A000-$BFFF, $D000-$DFFF and $E000-$FFFF
                                ;       Values :  %x00 :  RAM
visible in all 3 areas
                                ;                 %x01 :  RAM
visible at $A000-$BFFF and $E000-$FFFF
                                ;                 %x10 :  RAM
visible at $A000-$BFFF ; KERNAL ROM visible at $E000-$FFFF
                                ;                 %x11 :  BASIC
ROM visible at $A000-$BFFF; KERNAL ROM visible at $E000-$FFFF
                                ;                 %1xx :  I/O
area visible at $D000-$DFFF (Except for the value %100, see
above)
                                ;
                                ; Bit #3 : Datasette output
signal level
                                ; Bit #4 : Datasette button
status;  0 = One or more of PLAY,RECORD,FFWD or REW pressed
                                ;
1 = No button pressed
                                ; Bit #5 : Datasette motor
control; 0 = on; 1 = off
                                ;
                                ; Default: #$37, %00110111
;---------------------------------------------------------------
-----------------------------------
```

```
;
COLORS
;-------------------------------------------------------------------
-----------------------------------
COLOR_BLACK     = 0
COLOR_WHITE     = 1
COLOR_RED       = 2
COLOR_CYAN      = 3
COLOR_VIOLET    = 4
COLOR_GREEN     = 5
COLOR_BLUE      = 6
COLOR_YELLOW    = 7
COLOR_ORANGE    = 8
COLOR_BROWN     = 9
COLOR_LTRED     = 10
COLOR_GREY1     = 11
COLOR_GREY2     = 12
COLOR_LTGREEN   = 13
COLOR_LTBLUE    = 14
COLOR_GREY3     = 15
```