How to Recognize Video Mode and Generate Free Running Synchronization Signals Using TSC8051C1/C2 Microcontrollers

Description

The TSC8051C1 is an application specific microcontroller for autosync monitor and digital control application. It includes the TEMIC static 8–bit 80C51 CPU core with 8 Kbytes of ROM and 256 bytes of RAM, 12x8–bit PWM channels, buffered HSYNC and VSYNC outputs, a watchdog timer and a multimaster I^{2}C controller.

This application note describes how to automatically recognize video mode by measuring the period and polarity of horizontal and vertical synchronization signals; it also explains how to generate free running synchronization signal for burn–in purpose.

In the rest of the application note, the use of words Hsync and Vsync means horizontal synchronization signal and vertical synchronization signal respectively.

Typical Autosync Monitor Application

The introduction of the TSC8051C1 in CRT monitors allows manufacturer and final user to get maximum flexibility.

- Automatic parameters adjustment during factory set–up.
- Auto–alignment capabilities.
- Saving of factory default parameters.
- Versatile frequency range up to 100KHz.
- More adjustment parameters are available to the user.
- Automatic video mode recognition that allows automatic monitor adjustment to the values previously saved by user.
- On chip I^{2}C bus controller allows Access bus implementation and so monitor adjustment by the PC’s Keyboard.

Figure 1 shows a block diagram of a typical autosync monitor designed with the TSC8051C1.

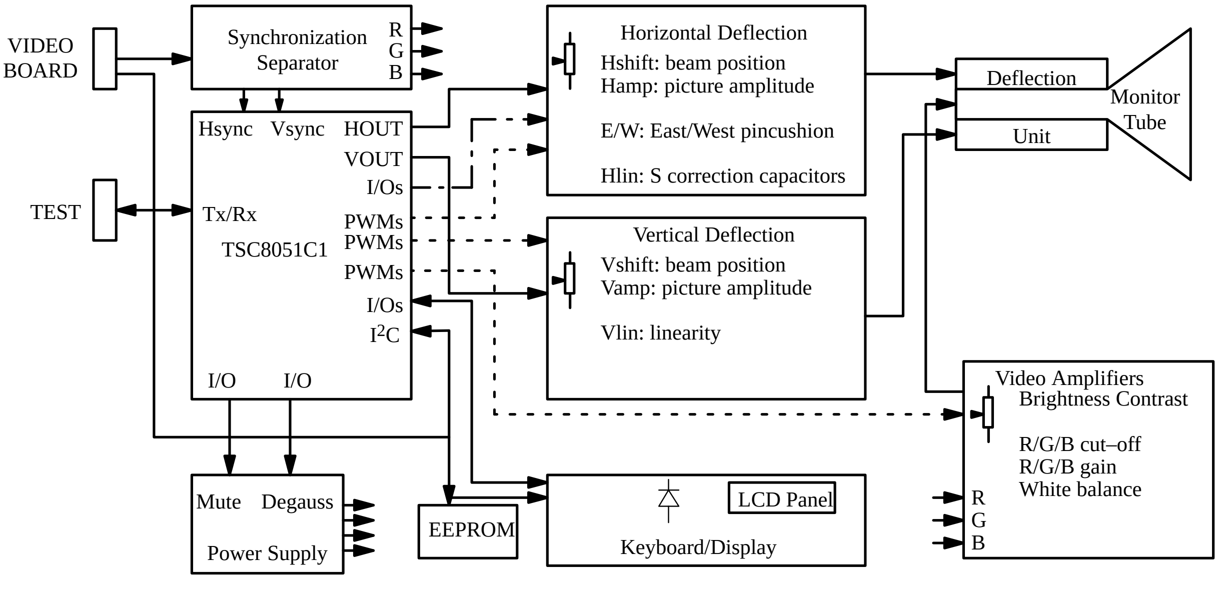

Figure 1. Autosync monitor block diagram with the TSC8051C1

Hardware Description

TSC8051C1 implements some special features to allow video mode recognition without adding any external components.

- Special Hsync and Vsync inputs.
  - Vsync can generate an interrupt on either falling or rising edge. As 8051 core samples inputs one time per machine cycle, pulse duration less than Tosc x 12 (1µs using 12 MHz crystal) are not 100% detected. To allow Hsync pulses counting (duration>150ns), pulses are lengthened up to 1 cycle period to be sampled by the 8051 core. Figure 2 and Figure 3 show the VSYNC and HSYNC input block diagrams.
  - These features are programmable through EICON SFR (address E4h).

MSB **EICON SFR E4h** **LSB**

–	–	–	–	–	T0L	T0S	I0L

Symbol	Position	Name and Function
I0L	EICON.0	INT0/VSYNC input Level bit. Setting this bit inverts INT0/VSYNC input signal. Clearing it allows standard use of INT0/VSYNC input.
T0S	EICON.1	T0/HSYNC input Selection bit. Setting this bit allows short pulse capture. Clearing it allows standard use of T0/HSYNC input.
T0L	EICON.2	T0/HSYNC input Level bit. Setting this bit allows positive pulse capture. Clearing it allows negative pulse capture.

- Special Hsync and Vsync outputs.
- TSC8051C1 implements programmable Hsync and Vsync outputs. User can disable and can invert these outputs to provide good polarity to deflection stages.
- These features are programmable through SOCR SFR (address E5h).

MSB **SOCR SFR E5h** **LSB**

–	–	VOS	HOS	VOP	VOE	HOP	HOE

Symbol	Position	Name and Function
HOE	SOCR.0	HSYNC Output Enable bit. Setting this bit enables the HSYNC signal.
HOP	SOCR.1	HSYNC Output Polarity bit. Setting this bit inverts the HSYNC output.
VOE	SOCR.2	VSYNC Output Enable bit. Setting this bit enables the VSYNC signal.
VOP	SOCR.3	VSYNC Output Polarity bit. Setting this bit inverts the VSYNC output.
HOS	SOCR.4	HSYNC Output Selection bit. Setting this bit selects the VSYNC output, clearing it selects P3.5 SFR bit.
VOS	SOCR.5	VSYNC Output Selection bit. Setting this bit selects the VSYNC output, clearing it selects P3.3 SFR bit.

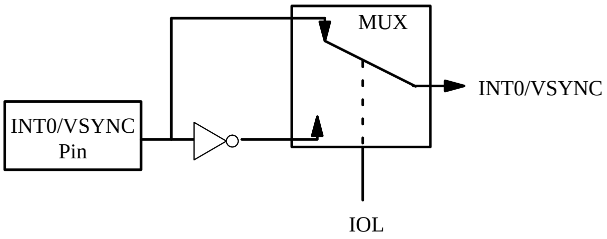

Figure 2. INT0/VSYNC input block diagram

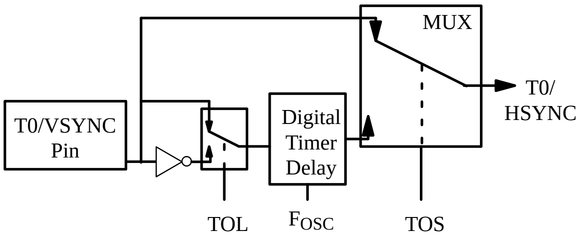

Figure 3. T0/HSYNC input block diagram

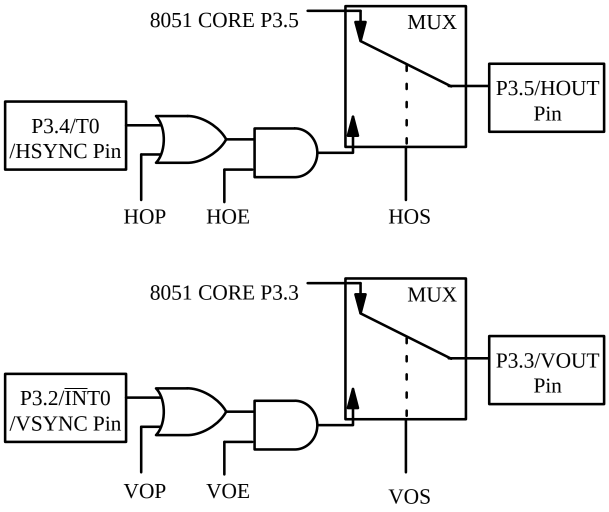

Figure 4. HSYNC and VSYNC outputs block diagram

Video Mode Recognition Description

Vsync input is programmed to generate an interrupt each time a falling edge appears on Vsync.

Vsync period measurement, Vsync polarity detection and Hsync pulse counting are performed using Timer 0. Figure 5 shows the Timer 0 block diagram in mode 1.

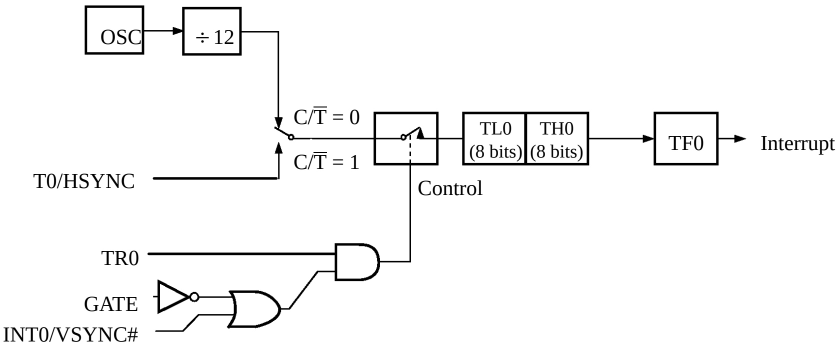

Figure 5. Timer/Counter 0 in mode 1: 16–bit Counter

The measurement cycle is divided in 3 operations (3 Hsync frames):

Timer 0 is reset and Vsync interrupt is enabled.

- Vsync frequency measurement:
- In first interrupt of the cycle, timer 0 is programmed to be used as free running timer with fosc/12 clock. TR0 bit is set to start counting (GATE bit is reset). In the second interrupt, TR0 bit is reset to stop counting. At this time, TH0 and TL0 registers contain a representative value of the Vsync period (in µs if 12MHz crystal is used) (see Figure 6).

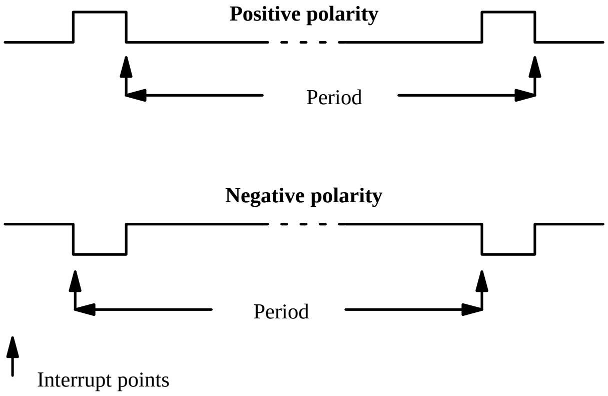

Figure 6. Vsync frequency measurement

- Vsync polarity detection:
- In the second interrupt of the cycle, timer 0 is programmed to be used as gated timer with fosc/12 clock. GATE bit is set and timer counts only during Vsync high level. In the third interrupt, TR0 bit is reset to stop counting. At this time, TH0 and TL0 registers contain a representative value of the Vsync high level duration (in µs if 12MHz crystal is used). If this duration is higher than the Vsync period divided by 2 then, Vsync has a negative polarity else it has a positive polarity (see Figure 7).

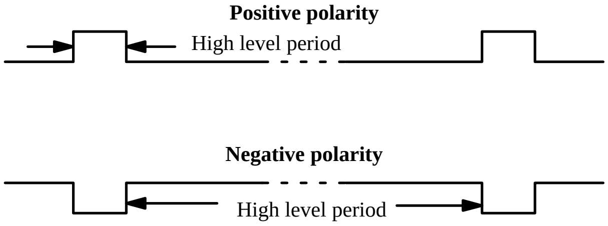

Figure 7. Vsync polarity detection

- Hsync polarity detection:
- In the second interrupt of the cycle, during the Vsync high level duration measurement, the Hsync signal is sampled 16 times, if number of high level samples is greater than 8 then, Hsync has a negative polarity else

it has a positive polarity (see Figure 8). Hsync input filter is then set to accept negative pulses or positive pulses respectively.

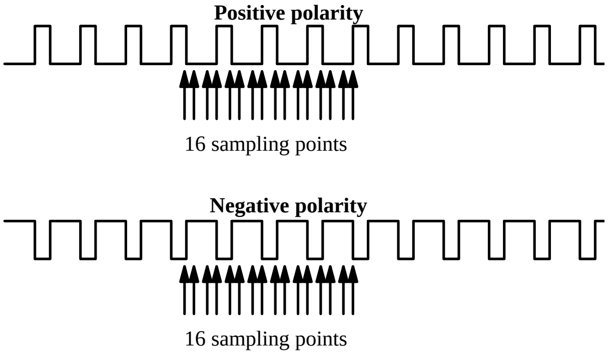

Figure 8. Hsync polarity detection

- Hsync frequency measurement:
- In the third interrupt of the cycle, timer 0 is programmed to be used as external event counter with Hsync clock. TR0 bit is set to start counting. In the fourth interrupt (last of the measurement cycle), TR0 bit is reset to stop counting. At this time, TH0 and TL0 registers contain a representative value of the Hsync period that is the number of Hsync pulses during a Vsync period (see Figure 9). A flag is set to inform main program of the end of cycle.

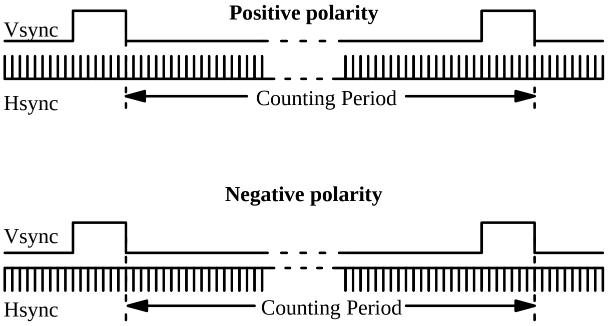

Figure 9. Hsync frequency measurement

When one cycle is completed, the main program checks the values and determines whether if the video mode has changed or not. If yes, the actions to take are listed hereafter:

- Depending on the Hsync frequency, S correction capacitors have to be updated.
- Some PWM values are updated.
- Video mute is activated.
- A research is made in EEPROM to find if the same video mode is already stored.
- If yes recall user set–up (update PWM values) from EEPROM, else default set–up is applied and the video mute is released.

Free Running Generation Description

During manufacturing burn–in, monitors are powered, but no video source is connected to the monitor. To force deflection stages' activity, free running Hsync and Vsync are output.

The software solution for free running generation, offers to the user a maximum of flexibility to program the best frequencies according to the deflection stages.

Two examples are proposed. In the first one, Vsync is a 60.1Hz negative polarity signal with 66µs pulses and Hsync is a 41.7KHz negative polarity signal with 2µs pulses, in the second one Vsync is a 72Hz positive polarity signal with 58µs pulses and Hsync is a 62.5KHz positive polarity signal with 1µs pulses.

Software Description

The software proposed hereafter is divided in two main routines:

- The Vsync Interrupt service routine.
- The Hsync/Vsync free running generation routine.

All the routines are based on a 12MHz oscillator operation; so 1 machine cycle has exactly 1µs duration. In the example, deflection stages are considered having a negative polarity synchronization input. User can program positive Hsync/Vsync outputs, by modifying the software as follows:

```
mov           A,#HOUT_VOUT_ENA
mov           C,Vpol_m
cpl           C     ; set VOP bit for positive
mov           ACC.3,C               ; polarity on Vout
mov           C,Hpol_m
cpl           C     ; set HOP bit for positive
mov           ACC.1,C               ; polarity on Hout
mov           SOCR,A                ; update Vout/Hout polarity
```

Vsync interrupt service routine has the highest priority. Due to the sampling clock, the Vsync period has a basic precision of 1µs. Depending on the instruction executed during the interrupt activation the measured period may be increased up to 4µs. The validation of a new detected video mode is effective only when the difference between the new measured and the previously saved period/counting is significant. This is achieved by the Check_diff subroutine.

The reception of a character on serial port activates one of the two Hsync/Vsync free running generation routine: ‘1’ for the first example, ‘2’ for the second one.

As the generation of the synchronization signals uses 100% of the CPU time, the only ways to disable generation are to clear the activation flag during an interrupt service routine (in the example, the flag is cleared in Vsync interrupt when a video source is input) or to apply a reset.

The listing includes the file reg51c1.inc that is the TSC8051C1 register declarations.

After a new video mode recognition, parameters are stored in the following variables:

`Vpol_s`	`Vsync polarity`
`Hpol_s`	`Hsync polarity`
`Vperl_s`	`Vsync period high order byte`
`Vperh_s`	`Vsync period low order byte`
`Hcntl_s`	`Hsync count high order byte`
`Hcnth_s`	`Hsync count low order byte`

The table hereafter presents different video modes and their associated parameters.

MODE	H. Frequency (KHz)	V. Frequency (Hz)	H. Polarity (Hpol_s)	V. Polarity (Vpol_s)	H. Count (Hcnt_s)	V.period (Vper_s)
EGA 640x350	31.5	70	+ (1)	– (0)	1C2h	37CEh
CGA 640x400	31.5	70	– (0)	+ (1)	1C2h	37CEh
VGA 640x480	31.5	60	– (0)	– (0)	20Dh	411Bh
VGA plus 800x600	35.5	56	+ (1)	– (0)	276h	45C1h
SVGA 800x600	37.8	60	+ (1)	+ (1)	276h	411Bh
VESA 800x600	48	72	+ (1)	+ (1)	29Ch	3641h
8514/a 1024x768	35.5	87	+ (1)	– (0)	198h	2CE6h
1280x1024	63.5	60	+ (1)	+ (1)	422h	411Bh

```
1                 ; TEMIC 1996.
2                 ; Demonstration program for video mode recognition
3                 ; and free running generation with TSC 8051C1
4
5                 $RB (0,1)                              ; bank 0 and 1 reserved
            6     $INCLUDE  (reg51c1.inc)  ; register declarations
            7     ; TEMIC 1996.
            8     ; Register declarations for TSC 8051C1 microcontroller
            9     ; Rev. A
            10
            11    ; BYTE Registers
0080        12    P0               DATA     080H
0090        13    P1               DATA     090H
00A0        14    P2               DATA     0A0H
00B0        15    P3               DATA     0B0H
            16
00D0        17    PSW              DATA     0D0H
00E0        18    ACC              DATA     0E0H
00F0        19    B                DATA     0F0H
0081        20    SP               DATA     081H
0082        21    DPL              DATA     082H
0083        22    DPH              DATA     083H
0087        23    PCON             DATA     087H
0088        24    TCON             DATA     088H
0089        25    TMOD             DATA     089H
008A        26    TL0              DATA     08AH
008B        27    TL1              DATA     08BH
008C        28    TH0              DATA     08CH
008D        29    TH1              DATA     08DH
00A8        30    IE               DATA     0A8H
00B8        31    IP               DATA     0B8H
0098        32    S0CON            DATA     098H
0099        33    S0BUF            DATA     099H
            34
00D8        35    S1CON            DATA     0D8H
00D9        36    S1STA            DATA     0D9H
00DA        37    S1DAT            DATA     0DAH
00DB        38    S1ADR            DATA     0DBH
            39
00AF        40    MSCON            DATA     0AFh
00E4        41    EICON            DATA     0E4h
00E5        42    SOCR             DATA    0E5h
00E6        43    HWDR             DATA    0E6h
00DF        44    PWMCON           DATA    0DFh
00E7        45    MXCRo            DATA    0E7h
00D7        46    MXCR1            DATA    0D7h
00EC        47    PWM0             DATA    0ECh
00ED        48    PWM1             DATA    0EDh
00EE        49    PWM2             DATA    0EEh
00EF        50    PWM3             DATA    0EFh
00F4        51    PWM4             DATA    0F4h
00F5        52    PWM5             DATA    0F5h
00F6        53    PWM6             DATA    0F6h
00F7        54    PWM7             DATA    0F7h
00FC        55    PWM8             DATA    0FCh
00FD        56    PWM9             DATA    0FDh
00FE        57    PWM10            DATA    0FEh
00FF        58    PWM11            DATA    0FFh
            59
            60
            61    ; BIT Registers
            62    ; PSW
00D7        63    CY               BIT     0D7H
```

```
00D6         64   AC               BIT     0D6H
00D5         65   F0               BIT     0D5H
00D4         66   RS1              BIT     0D4H
00D3         67   RS0              BIT     0D3H
00D2         68   OV               BIT     0D2H
00D0         69   P                BIT     0D0H
             70
             71   ; TCON
008F         72   TF1              BIT     08FH
008E         73   TR1              BIT     08EH
008D         74   TF0              BIT     08DH
008C         75   TR0              BIT     08CH
008B         76   IE1              BIT     08BH
008A         77   IT1              BIT     08AH
0089         78   IE0              BIT     089H
0088         79   IT0              BIT     088H
             80
             81   ; IE
00AF         82   EA               BIT     0AFH
00AD         83   ES1              BIT     0ADH
00AC         84   ES0              BIT     0ACH
00AB         85   ET1              BIT     0ABH
00AA         86   EX1              BIT     0AAH
00A9         87   ET0              BIT     0A9H
00A8         88   EX0              BIT     0A8H
             89
             90   ; IP0
00BD         91   PS1              BIT     0BDH
00BC         92   PS0              BIT     0BCH
00BB         93   PT1              BIT     0BBH
00BA         94   PX1              BIT     0BAH
00B9         95   PT0              BIT     0B9H
00B8         96   PX0              BIT     0B8H
             97
             98   ; P3
00B7         99   RD               BIT     0B7H
00B7         100  SDA              BIT     0B7H
00B6         101  WR               BIT     0B6H
00B6         102  SCL              BIT     0B6H
00B5         103  T1               BIT     0B5H
00B5         104  HOUT             BIT     0B5H
00B4         105  T0               BIT     0B4H
00B4         106  HSYNC            BIT     0B4H
00B3         107  INT1             BIT     0B3H
00B3         108  VOUT             BIT     0B3H
00B2         109  INT0             BIT     0B2H
00B2         110  VSYNC            BIT     0B2H
00B1         111  TXD              BIT     0B1H
00B0         112  RXD              BIT     0B0H
             113
             114  ; S0CON
009F         115  SM0              BIT     09FH
009E         116  SM1              BIT     09EH
009D         117  SM2              BIT     09DH
009C         118  REN              BIT     09CH
009B         119  TB8              BIT     09BH
009A         120  RB8              BIT     09AH
0099         121  TI               BIT     099H
0098         122  RI               BIT     098H
             123
             124  ; S1CON
00D85             CR0              BIT     0D8H
00D9         126  CR1              BIT     0D9H
```

```
00DA          AA               BIT     0DAH
00DB          SI               BIT     0DBH
00DC          STO              BIT     0DCH
00DD          STA              BIT     0DDH
00DE          ENS1             BIT     0DEH




              ; CONSTANT DEFINITION
              ; -------------------
0080          WDT_PER          EQU     80h           ; 2s watchdog period

              HOUT_VOUT_SET    EQU     00101000b     ; Hout/Vout=1
00D7          HOUT_VOUT_CLR    EQU     11010111b     ; Hout/Vout=0
0035          HOUT_VOUT_ENA    EQU     00110101b     ; Hout/Vout enabled
00CF          HOUT_VOUT_DIS    EQU     11001111b     ; Hout/Vout disabled

0002          EICON_H_NEG      EQU     00000010b     ; negative Hsync selection
0006          EICON_H_POS      EQU     00000110b     ; positive Hsync selection

0008          VSYNC_DIFF       EQU     8             ; 7us=Vsync diff authorised
0002          HSYNC_DIFF       EQU     2             ; 1 pulse=Hsync diff authorised

              ; BIT VARIABLE DEFINITION
              ; -----------------------
----                           BSEG AT 20h
0020          Vpol_m:          DBIT    1             ; measured Vsync polarity
0021          Hpol_m:          DBIT    1             ; measured Hsync polarity
0022          Vpol_s:          DBIT    1             ; saved Vsync polarity
0023          Hpol_s:          DBIT    1             ; saved Hsync polarity
0024          End_cycle:       DBIT    1             ; measuring end cycle flag (1)
0025          Free_run:        DBIT    1             ; Free running generation flag (1)

              ; DATA VARIABLE DEFINITION
              ; ------------------------
----                           DSEG AT 30h
0030          Isr_state:       DS      1             ; Interrupt state flag
0031          Vperl_m:         DS      1             ; measured period high order byte
0032          Vperh_m:         DS      1             ; measured period low order byte
0033          Hcntl_m:         DS      1             ; measured count high order byte
0034          Hcnth_m:         DS      1             ; measured count low order byte
0035          Vperl_s:         DS      1             ; saved period high order byte
0036          Vperh_s:         DS      1             ; saved period low order byte
0037          Hcntl_s:         DS      1             ; saved count high order byte
0038          Hcnth_s:         DS      1             ; saved count low order byte

0039          Stack:           DS      10h           ; 16 bytes stack



              ;=====================================================================
              ;                        BEGIN CODE
              ;=====================================================================

                               USING   0             ; RB0 used by default
----                           CSEG
0000                           ORG     0000h         ; reset address
0000 0106                      ajmp    Reset

0003                           ORG     0003h         ; VSYNC (INT0) interrupt
0003 0200FC                    ljmp    Vsync_isr

```

```

             ;=====================================================================
             ;                        INITIALISATION
             ;=====================================================================

0006 758138  Reset:           mov     SP,#Stack-1   ; stack pointer initialisation

0009 78FF                     mov     R0,#0FFh      ; Internal RAM initialisation
000B 7600    Ram_init:        mov     @R0,#00H
000D D8FC                     djnz    R0,Ram_init

000F 758DE6                   mov     TH1,#0E6h     ; T1 used as baud rate generator
0012 758B00                   mov     TL1,#00h      ; at 1200 bauds with 12MHz crystal
0015 758921                   mov     TMOD,#21h     ; T0 16b counter, T1 8b autoreload
0018 758841                   mov     TCON,#41h     ; T1 run, INT0 falling edge

001B 758700                   mov     PCON,#00h     ; SMOD=0
001E 759852                   mov     S0CON,#52h    ; 8-bit UART, Rx enabled

0021 75A881                   mov     IE,#81h       ; IE0 enabled
0024 75B801                   mov     IP,#01h       ; IE0 high priority

0027 75E680                   mov     HWDR,#WDT_PER ; watchdog activation


             ;========================================================================
             ;                        MAIN PROGRAM
             ;========================================================================

002A 202410  Wait_sync:       jb      End_cycle,Check_mode
002D 75E680                   mov     HWDR,#WDT_PER ; watchdog refresh

                              ; here must be inserted the
                              ; man-machine interface control

0030 3098F7                   jnb     RI,Wait_sync  ; example for free running
0033 C298                     clr     RI
0035 E599                     mov     A,S0BUF

0037 B43104                   cjne    A,#’1’,test_car
003A 11B3                     acall   H_V_sync_gen_1; Free running generation
003C 012A                     ajmp    Wait_sync

003E B432E9  test_car:        cjne    A,#’2’,Wait_sync
0041 3104                     acall   H_V_sync_gen_2
                                                    ; Free running generation
0043 012A                     ajmp    Wait_sync

0045 E4      Check_mode:      clr     A
0046 A220                     mov     C,Vpol_m
0048 92E0                     mov     ACC.0,C
004A A222                     mov     C,Vpol_s
004C 9400                     subb    A,#00
004E 702A                     jnz     Mode_changed  ; Vsync polarity changed

0050 A221                     mov     C,Hpol_m
0052 92E0                     mov     ACC.0,C
0054 A223                     mov     C,Hpol_s
0056 9400                     subb    A,#00
0058 7020                     jnz     Mode_changed  ; Hsync polarity changed

005A A831                     mov     R0,Vperl_m
005C A932                     mov     R1,Vperh_m
```

```
 005E AA35                         mov     R2,Vperl_s
 0060 AB36                         mov     R3,Vperh_s
 0062 7C08                         mov     R4,#VSYNC_DIFF
 0064 11A0                         acall   Check_diff    ; compare new & old period
 0066 7012                         jnz     Mode_changed  ; Vsync period changed

 0068 A833                         mov     R0,Hcntl_m
 006A A934                         mov     R1,Hcnth_m
 006C AA37                         mov     R2,Hcntl_s
 006E AB38                         mov     R3,Hcnth_s
 0070 7C02                         mov     R4,#HSYNC_DIFF
 0072 11A0                         acall   Check_diff    ; compare new & old counting
 0074 7004                         jnz     Mode_changed  ; Vsync period changed

 0076 C224                         clr     End_cycle     ; a new cycle can start
 0078 012A                         ajmp    Wait_sync

 007A             Mode_changed:

                  ; user define    setb    Video_mute    ; video mute during mode change
                  ; user define    acall   Cs_select     ; S correction capacitors update
                                                         ; depending on Vsync period

 007A 7435                         mov     A,#HOUT_VOUT_ENA
 007C A220                         mov     C,Vpol_m      ; set VOP bit for negative
 007E 92E3                         mov     ACC.3,C       ; polarity on Vout
 0080 A221                         mov     C,Hpol_m      ; set HOP bit for negative
 0082 92E1                         mov     ACC.1,C       ; polarity on Hout
0084 F5E5                          mov     SOCR,A        ; update Vout/Hout polarity

 0086 A220                         mov     C,Vpol_m
 0088 9222                         mov     Vpol_s,C      ; save new Vsync polarity
 008A 853135                       mov     Vperl_s,Vperl_m
 008D 853236                       mov     Vperh_s,Vperh_m
                                                         ; save new Vsync period

 0090 A221                         mov     C,Hpol_m
 0092 9223                         mov     Hpol_s,C      ; save new Hsync polarity
 0094 853337                       mov     Hcntl_s,Hcntl_m
 0097 853438                       mov     Hcnth_s,Hcnth_m
                                                         ; save new Hsync period

                  ;    Here must be inserted the research of this video mode in EEPROM
                  ;    if it is already stored, then recall user’s screen parameters
                  ;    if not, recall factory default screen parameters.
                  ;    This new mode will be stored in EEPROM after user adjustments

 009A 31F9                         acall   Out_results   ; send results to serial port

                  ; user define    clr     Video_mute    ; end of video mute
 009C C224                         clr     End_cycle     ; a new cycle can start
 009E 012A                         ajmp    Wait_sync


                  ;========================================================================
                  ;                        SUBROUTINES
                  ;========================================================================

                  ;------------------------------------------------------------------------
                  ;
                  ;      This subroutine checks if the absolute difference of two words
                  ;      is less than a given byte value
                  ;
```

```
              ;      Inputs:  R0:     word 1 low order byte
              ;                R1:     word 1 high order byte
              ;                R2:     word 2 low order byte
              ;                R3:     word 2 high order byte
              ;                R4:     limit of difference
              ;
              ;      Outputs: A:      if A = 0 the difference is less than the limit
              ;                        else the difference is greater than or equal to
                                       the limit
              ;------------------------------------------------------------------------

00A0 C3       Check_diff:      clr     C
00A1 E9                        mov     A,R1
00A2 9B                        subb    A,R3          ; A=MSB difference
00A3 700D                      jnz     End_check     ; MSB not equal

00A5 E8                        mov     A,R0
00A6 9A                        subb    A,R2          ; A=LSB difference
00A7 5003                      jnc     Check_pos

00A9 C3                        clr     C             ; negative difference
00AA EA                        mov     A,R2
00AB 98                        subb    A,R0          ; A=LSB difference

00AC B50400   Check_pos:       cjne    A,AR4,$+3
00AF 5001                      jnc     End_check
00B1 E4                        clr     A
00B2          End_check:       ret



              ;------------------------------------------------------------------------
              ;
              ;     This subroutine generates free Running synchronization Signals
              ;
              ;     Hout = 41.7KHz with 2us negative pulses
              ;     Vout = 60.1Hz with 66us negative pulses
              ;
              ;------------------------------------------------------------------------

00B3 E5E5     H_V_sync_gen_1:  mov     A,SOCR
00B5 54CF                      anl     A,#HOUT_VOUT_DIS
00B7 F5E5                      mov     SOCR,A        ; select P3.3/P3.5 as Vout/Hout
00B9 D225                      setb    Free_run      ; set flag (cleared in Vsync isr)

00BB 53B0D7   V_pulse_1:       anl     P3,#HOUT_VOUT_CLR
00BE 00                        nop
00BF D2B5                      setb    HOUT          ; 2us neg pulse on Hout
00C1 790A                      mov     R1,#10
00C3 D9FE                      djnz    R1,$          ; 20us tempo

00C5 C2B5     clr              HOUT
00C7 00                        nop
00C8 D2B5                      setb    HOUT          ; 2us neg pulse on Hout
00CA 790A                      mov     R1,#10
00CC D9FE                      djnz    R1,$          ; 20us tempo

00CE C2B5                      clr     HOUT
00D0 00                        nop
00D1 D2B5                      setb    HOUT          ; 2us neg pulse on Hout
00D3 7907                      mov     R1,#7
00D5 D9FE                      djnz    R1,$          ; 14us tempo
00D7 D2B3                      setb    VOUT          ; 66us neg pulse on Vout
```

```

00D9 78E6                    mov     R0,#230       ; 230 * 3 Hsync pulses
00DB 00      H_pulse_1:      nop
00DC 00                      nop
00DD 75E680                  mov     HWDR,#WDT_PER ; watchdog refresh

00E0 C2B5                    clr     HOUT
00E2 00                      nop
00E3 D2B5                    setb    HOUT          ; 2us neg pulse on Hout
00E5 790A                    mov     R1,#10
00E7 D9FE                    djnz    R1,$          ; 20us tempo

00E9 C2B5                    clr     HOUT
00EB 00                      nop
00EC D2B5                    setb    HOUT          ; 2us neg pulse on Hout
00EE 790A                    mov     R1,#10
00F0 D9FE                    djnz    R1,$          ; 20us tempo

00F2 C2B5                    clr     HOUT
00F4 00                      nop
00F5 D2B5                    setb    HOUT          ; 2us neg pulse on Hout
00F7 7906                    mov     R1,#6
00F9 D9FE                    djnz    R1,$          ; 12us tempo

00FB 302505                  jnb     Free_run,End_gen_1
                                                   ; Free running enabled ?

00FE D8DB                    djnz    R0,H_pulse_1
0100 00                      nop
0101 01BB                    ajmp    V_pulse_1
0103 22      End_gen_1:      ret



            ;--------------------------------------------------------------------------
             ;
             ;       This subroutine generates free Running synchronization Signals
             ;
             ;       Hout = 62.5KHz with 1us positive pulses
             ;       Vout = 72Hz with 58us positive pulses
             ;
            ;--------------------------------------------------------------------------

0104 E5E5    H_V_sync_gen_2: mov     A,SOCR
0106 54CF                    anl     A,#HOUT_VOUT_DIS
0108 F5E5                    mov     SOCR,A        ; select P3.3/P3.5 as Vout/Hout
010A D225                    setb    Free_run      ; set flag (cleared in Vsync isr)

010C 53B0D7  V_pulse_2:      orl     P3,#HOUT_VOUT_SET
010F D2B5                    clr     HOUT          ; 1us pos pulse on Hout
0111 00                      nop
0112 7906                    mov     R1,#6
0114 D9FE                    djnz    R1,$          ; 12us tempo

0116 C2B5                    setb    HOUT
0118 D2B5                    clr     HOUT          ; 1us pos pulse on Hout
011A 00                      nop
011B 7906                    mov     R1,#6
011D D9FE                    djnz    R1,$          ; 12us tempo

011F C2B5                    setb    HOUT
0121 D2B5                    clr     HOUT          ; 1us pos pulse on Hout
0123 00                      nop
0124 7906                    mov     R1,#6
```

```
 0126 D9FE                      djnz     R1,$          ; 12us tempo

 0128 C2B5                      setb     HOUT
 012A D2B5                      clr      HOUT          ; 1us pos pulse on Hout
 012C 00                        nop
 012D 7903                      mov      R1,#3
 012F D9FE                      djnz     R1,$          ; 6us tempo
 0131 D2B3                      clr      VOUT          ; 58us pos pulse on Vout

 0133 78D8                      mov      R0,#216       ; 216 * 4 Hsync pulses
 0135 00        H_pulse_2:      nop
 0136 00                        nop
 0137 75E680                    mov      HWDR,#WDT_PER ;watchdog refresh

 013A C2B5                      setb     HOUT
 013C D2B5                      clr      HOUT          ; 1us pos pulse on Hout
 013E 00                        nop
 013F 7906                      mov      R1,#6
 0141 D9FE                      djnz     R1,$          ; 12us tempo

 0143 C2B5                      setb     HOUT
 0145 D2B5                      clr      HOUT          ; 1us pos pulse on Hout
 0147 00                        nop
 0148 7906                      mov      R1,#6
 014A D9FE                      djnz     R1,$          ; 12us tempo

 014C C2B5                      setb     HOUT
 014E D2B5                      clr      HOUT          ; 1us pos pulse on Hout
 0150 00                        nop
 0151 7906                      mov      R1,#6
 0153 D9FE                      djnz     R1,$          ; 12us tempo

 0155 C2B5                      setb     HOUT
 0157 D2B5                      clr      HOUT          ; 1us pos pulse on Hout
 0159 00                        nop
 015A 7902                      mov      R1,#2
 015C D9FE                      djnz     R1,$          ; 4us tempo

 015E 302505                    jnb      Free_run,End_gen_2
                                                       ; Free running enabled ?

 0161 D8D2                      djnz     R0,H_pulse_2
 0163 00                        nop
 0164 210C                      ajmp     V_pulse_2
 0166 22        End_gen_2:      ret




              ;---------------------------------------------------------------------------
                ;
                ;       This routine is the VSYNC interrupt service routine
                ;
              ;---------------------------------------------------------------------------
                                USING    1             ; RB1 used in interrupt

0167 C225       Vsync_isr:      clr      Free_run      ; stop Free running
0169 302401                     jnb      End_cycle,Vsync_beg_isr
                                                       ; Cycle not yet treated
 016C 32                        reti

 016D C0E0      Vsync_beg_isr:  push     ACC
 016F C0D0      push            PSW
```

```
0171 75D008                    mov     PSW,#08h      ; RB1 selection
0174 E530                      mov     A,Isr_state   ; load ISR state

0176 B4000F  State_0:          cjne    A,#00,state_1
0179 758921                    mov     TMOD,#21h     ; T0 : free running 16-bit timer
017C 758A00                    mov     TL0,#00h
017F 758C00                    mov     TH0,#00h
0182 D28C                      setb    TR0           ; start measuring Vsync period
0184 0530                      inc     Isr_state
0186 21F3                      ajmp    Vsync_end_isr

0188 B40133  State_1:          cjne    A,#01,state_2
018B 00                        nop                   ; nop are inserted for timing
018C 00                        nop                   ; compensation between start
018D 00                        nop                   ; and stop counting
018E 00                        nop
018F C28C                      clr     TR0           ; stop T0
0191 858A31                    mov     Vperl_m,TL0
0194 858C32                    mov     Vperh_m,TH0   ; store Vsync period
0197 758929                    mov     TMOD,#29h     ; T0 : gated 16-bit timer
019A 758A00                    mov     TL0,#00h
019D 758C00                    mov     TH0,#00h
01A0 D28C                      setb    TR0           ; start measuring Vsync high level

01A2 E4                        clr     A             ; Hsync polarity detection
01A3 7810                      mov     R0,#16        ; 16 samples
01A5 A2B4    S1_1:             mov     C,HSYNC       ; read pin level
01A7 3400                      addc    A,#00         ; store state
01A9 D8FA                      djnz    R0,S1_1
01AB B40800                    cjne    A,#08,S1_2
01AE 9221    S1_2:             mov     Hpol_m,C      ; store Hsync polarity (0 = neg)
01B0 4005                      jc      S1_3
01B2 75E402                    mov     EICON,#EICON_H_NEG
                                                     ; negative Hsync pulses selection
01B5 21BA                      ajmp    S1_4
01B7 75E406  S1_3:             mov     EICON,#EICON_H_POS
                                                     ; positive Hsync pulses selection

01BA 0530    S1_4:             inc     Isr_state
01BC 21F3                      ajmp    Vsync_end_isr

01BE B4021B  State_2:          cjne    A,#02,state_3
01C1 758925                    mov     TMOD,#25h     ; T0 : 16-bit counter
01C4 758A00                    mov     TL0,#00h
01C7 A88C                      mov     R0,TH0        ; save MSB
01C9 758C00                    mov     TH0,#00h
01CC D28C                      setb    TR0           ; start counting Hsync pulses

01CE E532                      mov     A,Vperh_m     ; Vsync polarity detection
01D0 C3                        clr     C
01D1 13                        rrc     A             ; A = Vsync period / 20
01D2 B50800                    cjne    A,AR0,S2_1    ; test only high order byte
01D5 B3      S2_1:             cpl     C
01D6 9220                      mov     Vpol_m,C      ; store Vsync polarity (0 = neg)
01D8 0530                      inc     Isr_state
01DA 21F3                      ajmp    Vsync_end_isr

01DC 00      state_3:          nop                   ; nop are inserted for timing
01DD 00                        nop                   ; compensation between start
01DE 00                        nop                   ; and stop counting
01DF 00                        nop
01E0 00                        nop
01E1 00                        nop
```

```
 01E2 00                        nop
 01E3 00                        nop
 01E4 00                        nop
 01E5 00                        nop
 01E6 C28C                      clr      TR0            ; stop T0
 01E8 858A33                    mov      Hcntl_m,TL0
 01EB 858C34                    mov      Hcnth_m,TH0    ; store Hsync pulse count
 01EE 753000                    mov      Isr_state,#00  ; reset ISR state to start a new
                                                        cycle
 01F1 D224                      setb     End_cycle      ; set flag for the main program

 01F3 D0D0     Vsync_end_isr:   pop      PSW
 01F5 D0E0                      pop      ACC
 01F7 32                        reti

                                USING    0              ; RB0 used by default


               ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
               ;                                                                       +
               ;                The subroutines hereafter are only used for
                                demonstration program+
               ;                                                                       +
               ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

               ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
               ; Send measured parameters on serial port                               +
               ; Vsync period, polarity                                                +
               ; Hsync counting, polarity                                              +
               ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 01F8 900256   Out_results:     mov      DPTR,#Vsync_msge
 01FB 512F                      acall    Out_msge       ; display Vsync message
 01FD E536                      mov      A,Vperh_s
 01FF 5139                      acall    Out_byte
 0201 E535                      mov      A,Vperl_s
 0203 5139                      acall    Out_byte       ; display Vsync period
 0205 7420                      mov      A,#' '
 0207 514E                      acall    Out_char
 0209 742D                      mov      A,#'-'
 020B A222                      mov      c,Vpol_s
 020D 5002                      jnc      Vpol_neg
 020F 742B                      mov      A,#'+'
 0211 514E     Vpol_neg:        acall    Out_char       ; display Vsync polarity

0213 90025F                     mov      DPTR,#Hsync_msge
0216 512F                       acall    Out_msge       ; display Hsync message
0218 E538                       mov      A,Hcnth_s
021A 5139                       acall    Out_byte
021C E537                       mov      A,Hcntl_s
021E 5139                       acall    Out_byte       ; display Hsync count
0220 7420                       mov      A,#' '
0222 514E                       acall    Out_char
0224 742D                       mov      A,#'-'
0226 A223                       mov      c,Hpol_s
0228 5002                       jnc      Hpol_neg
022A 742B                       mov      A,#'+'
022C 514E      Hpol_neg:        acall    Out_char       ; display Hsync polarity
022E 22                         ret

               ;++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
               ; Send a message on serial port                                         +
               ; DPTR is the message address                                           +
               ; a NUL character is the end of message                                 +
              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 022F E4      Out_msge:        clr      A
 0230 93                       movc     A,@A+DPTR
 0231 6005                     jz       Out_end       ; last character
 0233 A3                       inc      DPTR
 0234 514E                     acall    Out_char
 0236 412F                     ajmp     Out_msge
 0238 22      Out_end:         ret

              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              ; Send a hexadecimal byte on serial port                               +
              ; A is the byte to send                                                +
              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 0239 C0E0    Out_byte:        PUSH     ACC
 023B C4                       SWAP     A
 023C 5143                     ACALL    Out_nib
 023E D0E0                     POP      ACC
 0240 5143                     ACALL   Out_nib
 0242 22                       ret

              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              ; Send a hexadecimal nibble on serial port                             +
              ; A is the nibble to send                                              +
              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 0243 540F    Out_nib:         ANL      A,#0FH
 0245 2490                     ADD      A,#90H
 0247 D4                       DA       A
 0248 3440                     ADDC     A,#40H
 024A D4                       DA       A
 024B 514E                     acall    Out_char
 024D 22                       ret

              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              ;Send an ASCII character on serial port                                +
              ; A is the character to send                                           +
              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 024E 3099FD  Out_char:        jnb      TI,Out_char
 0251 C299                     clr      TI
 0253 F599                     mov      S0BUF,A
 0255 22                       ret

              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
              ; Messages definition                                                  +
              ;+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
0256 0A0D5673 Vsync_msge:      DB       0Ah,0Dh,’Vsync ’,0
 025A 796E6320
 025E 00
025F 0A0D4873 Hsync_msge:      DB       0Ah,0Dh,’Hsync ’,0
 0263 796E6320
 0267 00
```