PRODUCT NAME
TM58XX

TITLE
How to use the external clock setting function (TMR0) in the TM58XX series

APPLICATION NOTE
1. Introduction
2. DEMO code
3. Diagrams of the applicable circuits

1. Introduction

In the programming of the TM58 series, the external clock setting function of (TMR0) is used occasionally. In order to explain how the external clock setting function of (TMR0) (the EXT_CLK pin of TM58 is the external clock input pin) is used, the MCU of the TM58 series (using model TM58P10 as an example) will be introduced below. The special function register that is related to the external clock setting function of (TMR0) is SELECT (it does not take up any RAM address; the setting of its corresponding bit is set by the SELECT command), which is defined as follows:

SELECT：

Bit	Symbol	Description				
		PS2	PS1	PS0	TMR0 rate	WDT rate
2~0	PS2~PS0	0	0	0	1：2	1：1
		0	0	1	1：4	1：2
		0	1	0	1：8	1：4
		0	1	1	1：16	1：8
		1	0	0	1：32	1：16
		1	0	1	1：64	1：32
		1	1	0	1：128	1：64
		1	1	1	1：256	1：128
3	PSA	PSA: Prescaler assignment bit 1: Prescaler assigned to WDT 0: Prescaler assigned to TMR0				
4	EDGE0	EDGE0: TRM0 source signal edge control bit 1: Increment when H -> L transition on external clock 0: Increment when L -> H transition on external clock				
5	SUR0	SUR0: TMR0 clock source bit 1: External clock input 0: (Internal clock)/4 or instruction cycle				

NOTE:
The select register is the write register. The setting of its control bit is completed by executing the select command, during which the content of the accumulator is sent into the select register.

An example of the setting process is shown below:

```
movla   00h
select
```

After executing the command given above, the content of the select register will become 00h. Meanwhile, the prescaler is assigned to TMR0, and the prescaling ratio is 1:1, TMR0 uses the internal command cycle for counting.

2. DEMO Code

The following DEMO code and the diagrams of circuits indicate the process in which TM58P10 makes use of the external clock setting function:
(For the functions of the programming, please read the functional description in the programming lines).

```
;=========================================
;Mode: Advanced
;Crystal vibration: 4MHZ
; WATCHDOG: DISABLE
;Function: Programming for the external clock setting function of (TMR0)
;The system uses a delay command to create a clock signal with the cycle of 10uS, and
  exports the clock signal to the external clock input port of tmr0 via PORTB,0 port, so that
  TMR0 can carry out time setting that lasts 1S by using the external clock signal. The
  system controls the LED light by using the period of time setting, which makes it go on
  and off 1S, respectively. TMR0 sets time by using the method of interrupt.
;
;
;=========================================
;Definitions of the special function register
;=========================================
indf            equ     00h
tmr0            equ     01h
pc              equ     02h
status          equ     03h
fsr             equ     04h
porta           equ     05h
portb           equ     06h
wakeup          equ     20h
irqm            equ     21h
irqf            equ     22h
;
;
;=========================================
;Bit definitions of the destination register
;=========================================
```

```
w               equ     00h
f               equ     01h
;
;
;=========================================
; Bit definitions of the status register
;=========================================
c               equ     00h
dc              equ     01h
z               equ     02h
pd              equ     03h
to              equ     04h
pa0             equ     05h
pa1             equ     06h
pa2             equ     07h
;
;
;=========================================
; Bit definitions of the interrupt mask register (irqm)
;=========================================
intm            equ     07h
exintm          equ     02h
tmr0m           equ     00h
;
;
;=========================================
; Bit definitions of the interrupt flag register (irqf)
;=========================================
tmr0f           equ     00h
exintf          equ     02h
;
;
;=========================================
; Bit definitions of the select register
;=========================================
ps0             equ     00h
ps1             equ     01h
ps2             equ     02h
psa             equ     03h
rte             equ     04h
rts             equ     05h
;
;
;=========================================
; Bit definitions of the WAKEUP register
;=========================================
wdts            equ     07h
wue             equ     06h
eis             equ     05h
puh3            equ     03h
puh2            equ     02h
```

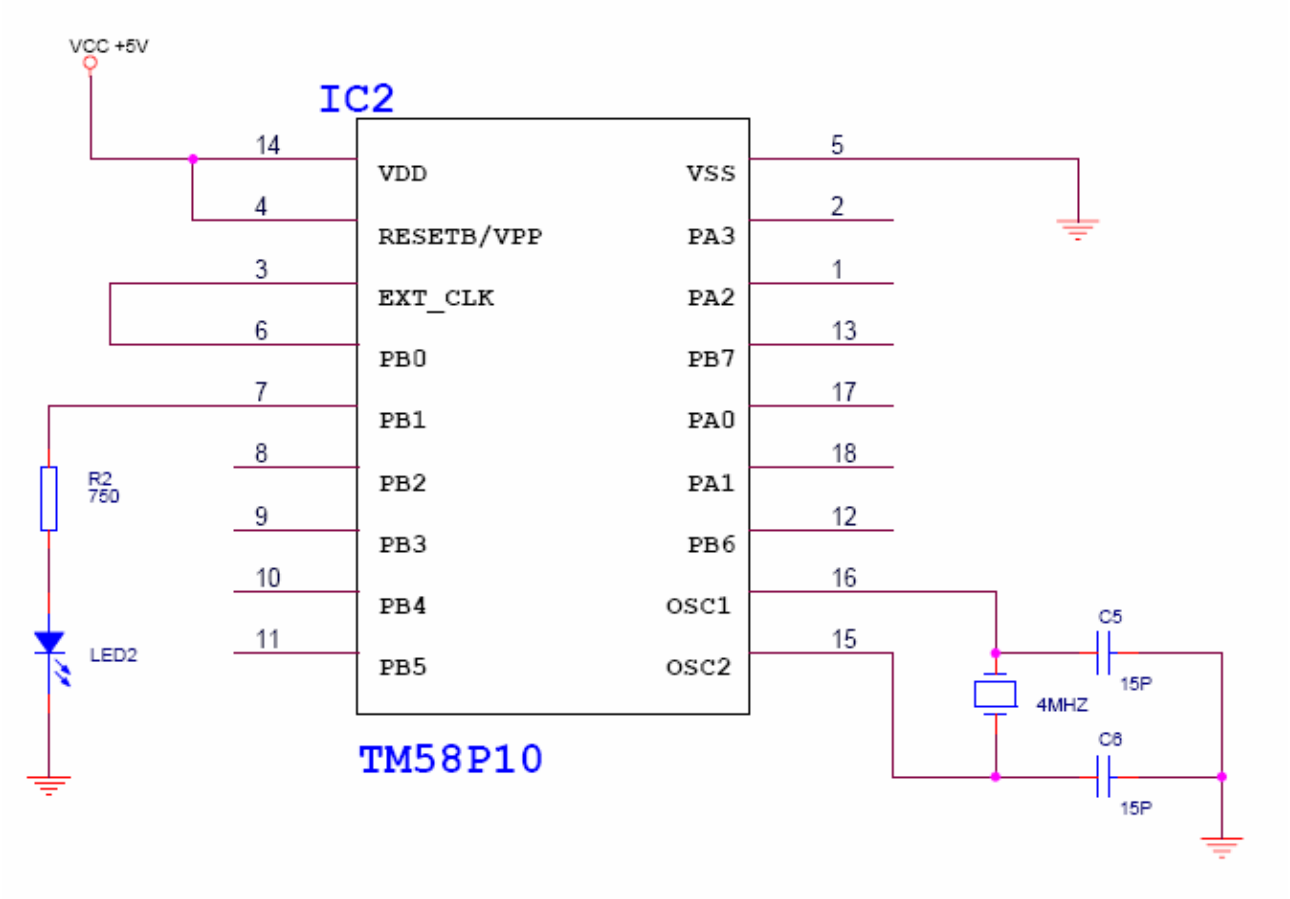
```
puh1            equ     01h
puh0            equ     00h
;
;
;=========================================
;Bit definitions
;=========================================
count           equ     08h
a_temp          equ     09h
status_temp     equ     0ah
;
;
;=========================================
;portb port     I/O definitions
;=========================================
wave            equ     00h
led             equ     01h
;
;
;=========================================
;program start here
;=========================================
        org     3ffH            ; Resets vector address
        goto    main
        org     3feh            ; Interrupts entrance address of the subprogram
        goto    time_int
        org     000h            ; The initialization address of programs in ROM
;=========================================
;MAIN PROGRAM
;=========================================
main:                           ; initialization
        clrm    count           ; Clears count
        movla   02h             ; Turns on led lights
        movam   portb
        clra
        iodir   portb           ;Sets portb port at export status
        movla   22h
        select                  ; Assigns tmr0 clock source as the external clock, positive edge
                                ; triggered
                                ; Assigns prescaler to tmr0, the prescaling ratio is 1:8
        movla   b'10000001'     ; Allows tmr0 interrupt function
        movam   irqm
        nop
        nop
;
;
start:                          ; Program section for generating 10uS tmr0 external clock
                                ; signal
        bsm     portb,wave      ; Generating external clock signal positive edge
        nop
        nop
```

```
        nop
        nop
        nop
        nop
        bcm     portb,wave    ; Generating external clock signal negative edge
        goto    start
;
;
;=========================================
; TMR0  interrupt subprogram（interrupts once every 20mS）
;=========================================
time_int:
        movam   a_temp        ; Sends the content of the accumulator into a_temp for
                              ; protection
        swapm   status,w      ; Exchanges the H-L half bytes in the status register and sends
                              ; the result into the accumulator
        movam   status_temp   ; Sends the result into status_temp for protection
        movla   06h           ; Resets the value of tmr0 to 06h, so that an interrupt occurs
                              ; every 250*8*10uS=20000uS at tmr0
        movam   tmr0          ;
        incm    count         ; adds 1 to the count value
        movla   32h
        xoram   count,w
        btmss   status,z      ; Determines whether it has reached 1S=50*20mS
        goto    end_int       ;1s not reached, turns to end_int for execution
        clrm    count         ;1s reached, clears count
        btmss   portb,led     ; Determines the on/off status of led lights
        goto    turn_on_led   ; led is off, turns on the light
turn_off_led:
        bcm     portb,led     ; Turns off led
        goto    end_int       ; Turns to end_int for execution
turn_on_led:
        bsm     portb,led     ; Turns on led
end_int:
        bcm     irqf,tmr0f    ; Clears tmr0 interrupt request flag
        swapm   status_temp,w; Exchanges the H-L half bytes in the status_temp and sends
                              ; the result into the accumulator
        movam   status        ; Sends the content of accumulator into the status register to
                              ; restore the status before entering an interrupt
        swapm   a_temp,f      ; Exchanges the H-L half bytes in the a_temp and sends the
                              ; result into a_temp
        swapm   a_temp,w      ; Exchanges the H-L half bytes in the a_temp,
                              ; and sends the result into the accumulator to restore the
                              ; content of the accumulator before entering an interrupt
        reti                  ; Returns from interrupt, and the program will automatically
                              ; turn on an interrupt after executing the command
;
;=========================================
        end                   ; Program ends
```

```
;=========================================
; Example
;=========================================
```

; After powering on the system, the led lights enter the loop of: on 1S---> off 1S---> on 1S. The 1S time setting period is implemented via the interrupt function, which is in turn launched when tmr0 uses the external clock signal as the clock source.

3. Diagrams of the applicable circuits



The APN circuitry diagram showing the external clock setting function of TMR0 in TM58 series.