PRODUCT NAME

TM58XX

TITLE

the way of TM58XX drives LCD directly with IO

APPLICATION NOTE

1.program
2.LCD Layout
3.application circuit diagram

```
;=========================================
;Title:TM58xx Application note
;file name----lcddisplay.asm
;
;Revision History:
;Rev:        Date:              Reason:
;1.0         5/8/2004           original
;
;
;=========================================
;
;
; Crystal oscillator:4MHZ
; WATCHDOG：  DISABLE
; MODE: ADVANCED
; Measure low-voltagly (LV): NO USE
;Function:LCD realize the display of the value of the counter
; The program design a counter, counter value per second to add by 1, the max of the counter count is 19999. After overflowing in counter value, the counter is counted from scratch.
; The counter value reveal through LCD,character M reveal Glimmerlly as second  while revealing counter value;
;
;
;=======================================
   include "d:\tm58p20.inc"    ; introduce include file
;
;
;=========================================
; position definition
;=========================================
buf1            equ 08h         ; the unit of counter value is buffered
buf2            equ 09h         ; the ten unit of counter value is buffered
buf3            equ 0ah         ; the hundred unit of counter value is buffered
buf4            equ 0bh         ; the thousand unit of counter value is buffered
buf5            equ 0ch         ; the 10 thousand unit of counter value is buffered
;---------------------------------
sec_acc         equ 0eh         ;1S Accumulator


time_temp       equ 0fh
time_buf        equ 10h
ms500_acc       equ 11h         ;0.5S Accumulator

;---------------------------------
seg1_buf        equ 12h         ;seg1 section code buffer
seg2_buf        equ 13h         ;seg2 section code buffer
seg3_buf        equ 14h         ;seg3 section code buffer
seg4_buf        equ 15h         ;seg4 section code buffer
seg5_buf        equ 16h         ;seg5 section code buffer
```

```
seg6_buf        equ 17h         ;seg6 section code buffer
seg7_buf        equ 18h         ;seg7 section code buffer
seg8_buf        equ 19h         ;seg8 section code buffer
;---------------------------------
com_dis_flag    equ 1ah         ; LCD COM port display and control register

m_flash_flag    equ 1bh         ; character M glitter as second display control register

;---------------------------------
portb_buf       equ 1ch         ;portb  buffer
portc_buf       equ 1dh         ;portc  buffer
;---------------------------------
io_buf          equ 1eh         ;portc i/o directional control buffer
;
;
;=======================================
;portc definition
;=======================================
com1            equ 07h
com2            equ 06h
com3            equ 05h
com4            equ 04h
;
;
;=======================================
;=======================================
;program start here
;=======================================
;=======================================
        org         7FFH    ;replacement vector

        lgoto       main
        org         000H
;=======================================
;LCD seg1 display section code table
;=======================================
seg1_table:
        addam pc
        retla       07h     ;"0"
        retla       00h     ;"1"
        retla       05h     ;"2"
        retla       01h     ;"3"
        retla       02h     ;"4"
        retla       03h     ;"5"
        retla       07h     ;"6"
        retla       01h     ;"7"
        retla       07h     ;"8"
        retla       03h     ;"9"
;
;
```

```
;=======================================
;LCD seg2 display section code table
;=======================================
seg2_table:
        addam pc
        retla       0dh         ;"0"
        retla       05h         ;"1"
        retla       0bh         ;"2"
        retla       0fh         ;"3"
        retla       07h         ;"4"
        retla       0eh         ;"5"
        retla       0eh         ;"6"
        retla       05h         ;"7"
        retla       0fh         ;"8"
        retla       0fh         ;"9"
;
;
;=======================================
;LCD seg3 display section code table
;=======================================
seg3_table:
        addam pc
        retla       0fh         ;"0"
        retla       00h         ;"1"
        retla       0dh         ;"2"
        retla       09h         ;"3"
        retla       02h         ;"4"
        retla       0bh         ;"5"
        retla       0fh         ;"6"
        retla       01h         ;"7"
        retla       0fh         ;"8"
        retla       0bh         ;"9"
;
;
;=======================================
;LCD seg4 display section code table
;=======================================
seg4_table:
        addam pc
        retla       05h         ;"0"
        retla       05h         ;"1"
        retla       03h         ;"2"
        retla       07h         ;"3"
        retla       07h         ;"4"
        retla       06h         ;"5"
        retla       06h         ;"6"
        retla       05h         ;"7"
        retla       07h         ;"8"
```

```
        retla       07h         ;"9"
;
;
;=======================================
;LCD seg5 display section code table
;=======================================
seg5_table:
        addam       pc
        retla       0fh         ;"0"
        retla       00h         ;"1"
        retla       0dh         ;"2"
        retla       09h         ;"3"
        retla       02h         ;"4"
        retla       0bh         ;"5"
        retla       0fh         ;"6"
        retla       01h         ;"7"
        retla       0fh         ;"8"
        retla       0bh         ;"9"
;
;
;=======================================
;LCD seg6 display section code table
;=======================================
seg6_table:
        addam       pc
        retla       05h         ;"0"
        retla       05h         ;"1"
        retla       03h         ;"2"
        retla       07h         ;"3"
        retla       07h         ;"4"
        retla       06h         ;"5"
        retla       06h         ;"6"
        retla       05h         ;"7"
        retla       07h         ;"8"
        retla       07h         ;"9"
;
;
;=======================================
;LCD seg7 display section code table
;=======================================
seg7_table:
        addam       pc
        retla       0fh         ;"0"
        retla       00h         ;"1"
        retla       0dh         ;"2"
        retla       09h         ;"3"
        retla       02h         ;"4"
        retla       0bh         ;"5"
        retla       0fh         ;"6"
        retla       01h         ;"7"
```

```
        retla       0fh         ;"8"
        retla       0bh         ;"9"
;
;
;=====================================
;LCD seg8 display section code table
;=====================================
seg8_table:
        addam       pc
        retla       05h         ;"0"
        retla       05h         ;"1"
        retla       03h         ;"2"
        retla       07h         ;"3"
        retla       07h         ;"4"
        retla       06h         ;"5"
        retla       06h         ;"6"
        retla       05h         ;"7"
        retla       07h         ;"8"
        retla       07h         ;"9"
;
;
;=====================================
;get the section code subprogram which corresponds to LCD com1
;=====================================
com1_seg:
        clrm        portb_buf
        btmsc       seg1_buf,0
        bsm         portb_buf,0
        btmsc       seg2_buf,0
        bsm         portb_buf,1
        btmsc       seg3_buf,0
        bsm         portb_buf,2
        btmsc        seg4_buf,0
        bsm         portb_buf,3
        btmsc       seg5_buf,0
        bsm         portb_buf,4
        btmsc       seg6_buf,0
        bsm         portb_buf,5
        btmsc       seg7_buf,0
        bsm         portb_buf,6
        btmsc       seg8_buf,0
        bsm         portb_buf,7
        ret
;
;
;=====================================
; get the section code subprogram which corresponds to LCD com2
;=====================================
com2_seg:
        clrm        portb_buf
```

```
        btmsc   seg1_buf,1
        bsm     portb_buf,0
        btmsc   seg2_buf,1
        bsm     portb_buf,1
        btmsc   seg3_buf,1
        bsm     portb_buf,2
        btmsc   seg4_buf,1
        bsm     portb_buf,3
        btmsc   seg5_buf,1
        bsm     portb_buf,4
        btmsc   seg6_buf,1
        bsm     portb_buf,5
        btmsc   seg7_buf,1
        bsm     portb_buf,6
        btmsc   seg8_buf,1
        bsm     portb_buf,7
        ret
;
;
;=====================================
; get the section code subprogram which corresponds to LCD com3
;=====================================
com3_seg:
        clrm    portb_buf
        btmsc   seg1_buf,2
        bsm     portb_buf,0
        btmsc   seg2_buf,2
        bsm     portb_buf,1
        btmsc   seg3_buf,2
        bsm     portb_buf,2
        btmsc   seg4_buf,2
        bsm     portb_buf,3
        btmsc   seg5_buf,2
        bsm     portb_buf,4
        btmsc   seg6_buf,2
        bsm     portb_buf,5
        btmsc   seg7_buf,2
        bsm     portb_buf,6
        btmsc   seg8_buf,2
        bsm     portb_buf,7
        ret
;
;
;=====================================
;get the section code subprogram which corresponds to LCD com4
;=====================================
com4_seg:
        clrm    portb_buf
        btmsc   seg1_buf,3
```

```
        bsm     portb_buf,0
        btmsc    seg2_buf,3
        bsm     portb_buf,1
        btmsc   seg3_buf,3
        bsm     portb_buf,2
        btmsc   seg4_buf,3
        bsm     portb_buf,3
        btmsc   seg5_buf,3
        bsm     portb_buf,4
        btmsc   seg6_buf,3
        bsm     portb_buf,5
        btmsc   seg7_buf,3
        bsm     portb_buf,6
        btmsc   seg8_buf,3
        bsm     portb_buf,7
        ret
;
;
;=======================================
;取反 display section code subprogram
;=======================================
comp_seg_buf:
        comm    seg1_buf,f
        comm    seg2_buf,f
        comm    seg3_buf,f
        comm    seg4_buf,f
        comm    seg5_buf,f
        comm    seg6_buf,f
        comm    seg7_buf,f
        comm    seg8_buf,f
        ret
;
;
;=======================================
; judge time subprogram
;=======================================
read_time:
        movm    tmr0,w
        xoram   time_buf,w
        movam   time_temp
        xoram   time_buf,f      ;tmr0==>time_buf
        btmss   time_temp,7     ;Is 128*32us/1000=4.096ms ?
        ret                     ;4.096ms arrive
        incm    sec_acc         ;if 4.096ms arrives,1s counter value add 1
        incm    ms500_acc       ;0.5s counter value add 1
        movla   07ah
        xoram   ms500_acc,w
        btmss   status,z        ;Is 0.5s arrive?
        lgoto   dis             ;0.5s not arrive
```

```
        clrm    ms500_acc       ;if 0.5sarrive,clear ms500_acc
        comm    m_flash_flag,f  ;the control flag of character M glitter as second ,
                                ;m_flash_flag
        movla   0f4h
        xoram   sec_acc,w
        btmss   status,z        ; Is 1s arrive?
        Lgoto   dis             ;1s is not arrive
        clrm    sec_acc         ;1s arrives ,clear sec_acc
        movla   09h
        xoram   buf1,w
        btmss   status,z        ; the units place (BUF1) counts the value whether is 9?

        lgoto   add_buf1        ;is not 9
        clrm    buf1            ;is 9,clear unit place(BUF1)
        movla   09h
        xoram   buf2,w
        btmss   status,z        ;Is decade units（BUF2）counter value equal to  9?
        lgoto   add_buf2        ;not equal to 9
        clrm    buf2            ;equalt to 9,clear decade units(BUF2)
        movla   09h
        xoram   buf3,w
        btmss   status,z        ;Is hundred units（BUF3）counte value equal to 9?
        lgoto   add_buf3        ; not equal to 9
        clrm    buf3            ; equalt to 9,clear hundred units(BUF3)
        movla   09h
        xoram   buf4,w
        btmss   status,z        ; Is thousand units（BUF4）counte value equal to 9?
        lgoto   add_buf4        ; not equal to 9
        clrm    buf4            ; equalt to 9,clear thousand units(BUF4)
        movla   01h
        xoram   buf5,w
        btmss   status,z        ; Is ten thousand units（BUF5）counte value equal to 1?
        lgoto   add_buf5        ;not equal to 1
        clrm    buf5            ; equalt to 1,clear thousand units(BUF5)
        lgoto   dis
add_buf1:
        incm    buf1            ; the units add 1
        lgoto   dis
add_buf2:
        incm    buf2            ; the decade units add 1
        lgoto   dis
add_buf3:
        incm    buf3            ; the hundred units add 1
        lgoto   dis
add_buf4:
        incm    buf4            ; the thousand units add 1
        lgoto   dis
add_buf5:
```

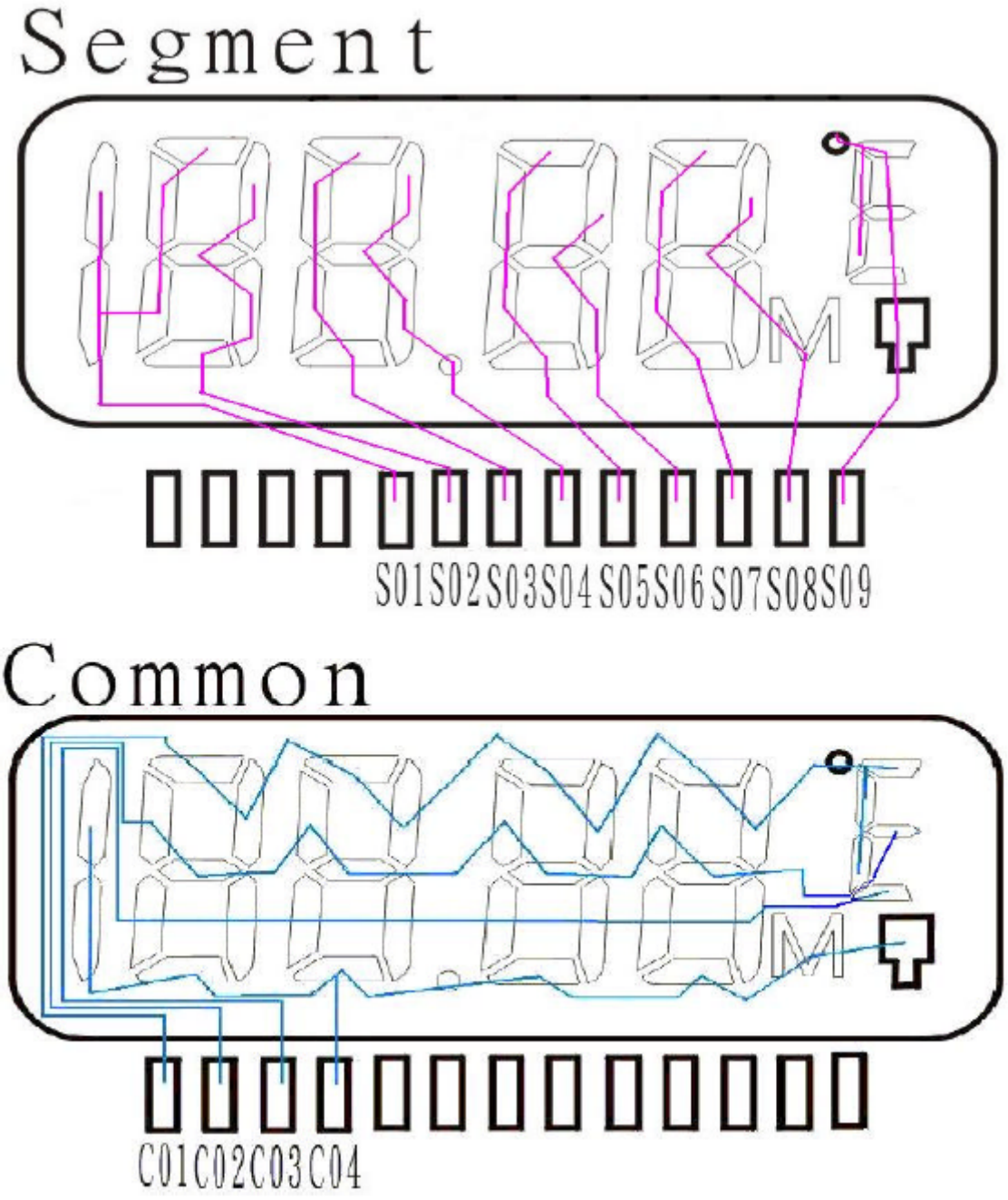
```
        incm    buf5            ; the ten thousand units add 1
dis:
        lcall   display
        ret
;
;
;=======================================
; display subprogram
;=======================================----------22
98+798
display:
        movla   0ffh
        iodir   portc
        movm    buf1,w
        lcall   seg7_table
        movam   seg7_buf        ; get section code which units replay seg7 displaying
        movm    buf1,w
        lcall   seg8_table
        movam   seg8_buf        ; get section code which decade units replay seg8
                                ;displaying
        movm    buf2,w
        lcall   seg5_table
        movam   seg5_buf        ; get section code which decade units replay seg5
                                ;displaying
        movm    buf2,w
        lcall   seg6_table
        movam   seg6_buf        ; get section code which decade units replay seg6
                                ;displaying
        movm    buf3,w
        lcall   seg3_table
        movam   seg3_buf        ;get section code which hundred units replay seg3
                                ;displaying
        movm    buf3,w
        lcall   seg4_table
        movam   seg4_buf        ; get section code which hundred units replay seg4
                                ;displaying
        movm    buf4,w
        lcall   seg1_table
        movam   seg1_buf        ; get section code which thousand units replay seg1
                                ;displaying
        movm    buf4,w
        lcall   seg2_table
        movam   seg2_buf        ; get section code which thousand units replay seg2
                                ;displaying
        btmss   m_flash_flag,0  ; Can it display character M?
        lgoto   dis_bit5        ; can`t display character M
        movla   b'00001000'     ; can display character M
        ioram   seg8_buf,f      ; send the section code of character M into
                                ;seg8_buf
```

```
dis_bit5:
        movm    buf5,w
        btmsc   status,z        ; Is the highest unit display 1?
        lgoto   dis_judge       ; the highest unit can`t display 1
        movla   b'00001000'     ; the highest unit display 1
        ioram   seg1_buf,f      ; send 1section code into seg1_buf
dis_judge:
        btmsc   com_dis_flag,0
        lgoto   dis_com1_h
        btmsc   com_dis_flag,1
        lgoto   dis_com1_l
        btmsc   com_dis_flag,2
        lgoto   dis_com2_h
        btmsc   com_dis_flag,3
        lgoto   dis_com2_l
        btmsc   com_dis_flag,4
        lgoto   dis_com3_h
        btmsc   com_dis_flag,5
        lgoto   dis_com3_l
        btmsc   com_dis_flag,6
        lgoto   dis_com4_h
dis_com4_l:
        bcm     com_dis_flag,7
        bsm     com_dis_flag,0
        lcall   com4_seg
        bcm     portc_buf,com4
        movla   0efh
        movam   io_buf
        lgoto   dis_lcd
dis_com4_h:
        bcm     com_dis_flag,6
        bsm     com_dis_flag,7
        lcall   comp_seg_buf
        lcall   com4_seg
        bsm     portc_buf,com4
        movla   0efh
        movam   io_buf
        lgoto   dis_lcd
dis_com3_l:
        bcm     com_dis_flag,5
        bsm     com_dis_flag,6
        lcall   com3_seg
        bcm     portc_buf,com3
        movla   0dfh
        movam   io_buf
        lgoto   dis_lcd
dis_com3_h:
        bcm     com_dis_flag,4
```

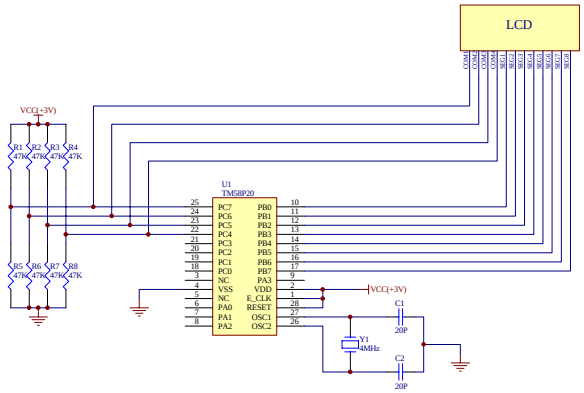
```
        bsm     com_dis_flag,5
        lcall   comp_seg_buf
        lcall   com3_seg
        bsm     portc_buf,com3
        movla   0dfh
        movam   io_buf
        lgoto   dis_lcd
dis_com2_l:
        bcm     com_dis_flag,3
        bsm     com_dis_flag,4
        lcall   com2_seg
        bcm     portc_buf,com2
        movla   0bfh
        movam   io_buf
        lgoto   dis_lcd
dis_com2_h:
        bcm     com_dis_flag,2
        bsm     com_dis_flag,3
        lcall   comp_seg_buf
        lcall   com2_seg
        bsm     portc_buf,com2
        movla   0bfh
        movam   io_buf
        lgoto   dis_lcd
dis_com1_l:
        bcm     com_dis_flag,1
        bsm     com_dis_flag,2
        lcall   com1_seg
        bcm     portc_buf,com1
        movla   7fh
        movam   io_buf
        lgoto   dis_lcd
dis_com1_h:
        bcm     com_dis_flag,0
        bsm     com_dis_flag,1
        lcall   comp_seg_buf
        lcall   com1_seg
        bsm     portc_buf,com1
        movla   7fh
        movam   io_buf
dis_lcd:
        movm    portc_buf,w
        movam   portc
        movm    portb_buf,w
        movam   portb
        movm    io_buf,w
        iodir   portc
        ret
```

```
;
;
;=======================================
;main program
;=======================================
main:
        clrm    buf1
        clrm    buf2
        clrm    buf3
        clrm    buf4
        clrm    buf5
;------------------------------
        clrm    sec_acc
        clrm    ms500_acc
;------------------------------
        movla   01h
        movam   com_dis_flag
        movla   0ffh
        movam   m_flash_flag
        movla   00h
        iodir   portb
        movla   04h             ;预分频器 point to tmr0,分频比 1:32
        select
        movm    tmr0,w
        movam   time_buf
        lcall   display
start:
        lcall   read_time
        lgoto   start
;
;
;=======================================
        end
;
;
;=========================================
;example
;=======================================
;After system electrify,LCD start displaying ‘0000’,then the value of
display  add 1 percent second.When the value displaying add to ‘1999’,
the add 1 LCD start displaying ‘0000’ again.The character M display as
second glitter as second.
```

LCD Layout


Segment
S01S02S03S04S05S06S07S08S09
Common
C01C02C03C04

the application circuit diagram



tenx	Title:		
	Size: B	Number:	Revision:
	Date:		Sheet of
	File:		Drawn by: