PRODUCT NAME

TM58XX

TITLE

TM58XX 用 HT1621 驱动 LCD 的方法

APPLICATION NOTE

1. 说明
2. 程序
3. LCD Layout
4. 应用线路图

1. 对 HT1621 初始化：

主要是对 HT1621 使用频率、音频选择、是否打开蜂鸣器等进行选择

先使能片选位(CS\=0)，ID 为 4 即格式命令为 100

然后送相应的命令字，送完后将 CS\脚置 1，即不对 HT1621 操作时 CSP 为 1。

2. 清屏：

HT1621 内部有 32*4 位的 RAM 单元，也可以说是输出缓冲区，将其清 0，即 LCD 不显示。

3. 输出：

Demo 板使用的 LCD 为五个显示位，本程序实现后两位固定输出 dE，前三位输出 000-->111...-->FFF 后动态停机

输出数据时先置片选位有效(CS\=0)－－＞然后送 ID 为 5（101）即命令格式为写操作－－＞接着将欲显段地址送 1621－－＞将要显的数据送 1621 输出－－＞最后关闭片选位即 CS\脚置 1。

其它说明：

(1)对 1621 写操作问题：

对 1621 操作时先置片选位有效（CSP＝0），不对 1621 操作时置片选位无效（CSP＝1）。

(2)注意地址、数据或命令的长度问题。

```
;---------------------------
;Title:         Write HT1621
;Function:
;Note:         The voltage applied to Vlcd pin must be lower than Vdd.
;Clock:         4m
;Config word: xt_osc & _wdt_off & _cpt_on & _type_general & _lv_don't use
;Data:         2004_5_9


;        list   p=tm58p20
;---------------------------
indf              equ     00h
tmr0              equ     01h
pc                equ     02h
status            equ     03h
fsr               equ     04h
porta             equ     05h
portb             equ     06h
csp               equ     4
wrp               equ     5
dtp               equ     6
portc             equ     07h
wakeup            equ     20h
irqm              equ     21h
irqf              equ     22h
;---------------------------
;status  bits define
c                 equ     00h
dc                equ     01h
z                 equ     02h
pd                equ     03h
to                equ     04h
pa0               equ     05h
pa1               equ     06h
pa2               equ     07h
;---------------------------
;option bits define
ps0               equ     00h
ps1               equ     01h
ps2               equ     02h
psa               equ     03h
rte               equ     04h
rts               equ     05h
;---------------------------
cnt               equ     08h
id                equ     09h
temp              equ     0ah
cmd               equ     0bh
```

```
temp1           equ     0ch
addr            equ     0dh
dat             equ     0eh
temper          equ     0fh


        org     7ffh
        lgoto   main
        org     00h
        nop
;--------------------------
lcd_tableh
      addam     pc,1
      retla     b'00000101'   ;0
      retla     b'00000000'   ;1
      retla     b'00000110'   ;2
      retla     b'00000010'   ;3
      retla     b'00000011'   ;4
      retla     b'00000011'   ;5
      retla     b'00000111'   ;6
      retla     b'00000000'   ;7
      retla     b'00000111'   ;8
      retla     b'00000011'   ;9
      retla     b'00000111'   ;a
      retla     b'00000111'   ;b
      retla     b'00000110'   ;c
      retla     b'00000110'   ;d
      retla     b'00000111'   ;e
      retla     b'00000111'   ;f
      retla     b'00000111'   ;f

lcd_tablel
      addam     pc,1
      retla     b'00001111'   ;0
      retla     b'00000110'   ;1
      retla     b'00001011'   ;2
      retla     b'00001111'   ;3
      retla     b'00000110'   ;4
      retla     b'00001101'   ;5
      retla     b'00001101'   ;6
      retla     b'00000111'   ;7
      retla     b'00001111'   ;8
      retla     b'00001111'   ;9
      retla     b'00000111'   ;a
      retla     b'00001100'   ;b
      retla     b'00001000'   ;c
      retla     b'00001110'   ;d
      retla     b'00001001'   ;e
      retla     b'00000001'   ;f
```

```
        retla       b'00000001'     ;f

;---------------------------
dispwriteid_:
        movla       3
        movam       cnt
        bcm         status,c
        rlm         id,m
        rlm         id,m
        rlm         id,m
        rlm         id,m
        rlm         id,m
        movm        id,a
        movam       temp
        lcall       dispwritebits_
        ret
;---------------------------
dispwritecmd_:
        movla       8
        movam       cnt
        movm        cmd,a
        movam       temp
        lcall       dispwritebits_
        ret
;---------------------------
dispwritebits_:
        rlm         temp,m
        btmsc       status,c
        lgoto       highs
low:
        bcm         portb,dtp       ;写数据'0'
        bcm         portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        bsm         portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        nop
        lgoto       loop1
highs:
        bsm         portb,dtp       ;写数据'1'
```

```
        bcm     portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        bsm     portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
loop1:
        decmsz  cnt,m
        lgoto   dispwritebits_
        ret
;---------------------------
dispwritebit_:
        movla   1
        movam   cnt
        movla   80h
        movam    temp
        lcall   dispwritebits_
        ret
;---------------------------
writeto46addr_:
        movla   7
        movam   cnt
        bcm     status,c
        rlm     addr,m
        movm    addr,a
        movam   temp
        lcall   dispwritebits_
        ret

dispwriteaddr_:
        movla   6
        movam   cnt
        bcm     status,c
        rlm     addr,m
        rlm     addr,m
        movm    addr,a
        movam   temp
        lcall   dispwritebits_
        ret
;---------------------------
dispwritedata_:
```

```
        movla   4
        movam   cnt
        movm    dat,a
        movam    temp
        lcall   dispwritebits1_
        ret
;---------------------------
;input: temp,cnt
;func:only for write dat low nibble
;-------------
dispwritebits1_:
loop00:
        rrm     temp,m
        btmsc   status,c
        lgoto   high11
low00:
        bcm     portb,dtp
        bcm     portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        nop
        bsm     portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        nop
        lgoto   loop10
high11:
        bsm     portb,dtp
        bcm     portb,wrp
        nop
        nop
        nop
        nop
        nop
        nop
        nop
        bsm     portb,wrp
        nop
        nop
        nop
        nop
```

```
        nop
        nop
        nop
        nop
loop10:
        decmsz    cnt,m
        lgoto     loop00
        ret
;--------------------------
initdisp_:
;turn on lcd display    id:100
        bcm       portb,csp       ;使能 ht1621
        movla     4
        movam     id
        lcall     dispwriteid_

        movla     3
        movam     cmd             ;turn on lcd bias gererator
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     b'00011000'   ;rc 256k
        movam     cmd
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     b'00101001'   ;select bias ,duty
        movam     cmd             ;1/4duty,1/3bias
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     b'11100011'   ;normal  mode
        movam     cmd
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     1               ;turn sys osc on
        movam     cmd
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     b'01100000'   ;set  buzzer 2khz
        movam     cmd
        lcall     dispwritecmd_
        lcall     dispwritebit_

        movla     b'00001000'   ;close buzzer
        movam     cmd
        lcall     dispwritecmd_
```

```
        lcall       dispwritebit_

        movla       b'00000101'    ;close watch dog
        movam       cmd
        lcall       dispwritecmd_
        lcall       dispwritebit_

        bsm         portb,csp      ;不对 HT1621 操作时 CSP 为 1
        nop
        ret
;--------------------------
clearall_:
        movla       d'24'          ;清除地址 08h 后的 24 个单元
        movam       temp1          ;08h 前的单元因为本程序未用,可以不理会
        bcm         portb,csp
        movla       d'5'
        movam       id
        lcall       dispwriteid_   ;3 位 id
        movla       d'8'
        movam       addr
        lcall       dispwriteaddr_;6 位地址
bbb:
        clrm        dat
        lcall       dispwritedata_ ;8 位数据
        decmsz      temp1,1
        lgoto       bbb
        bsm         portb,csp
        nop
        nop
        ret
;--------------------------
;seg10 seg9     显示第 1 位数据
;seg8  seg13    显示第 2 位数据
;seg17 seg21    显示第 3 位数据
;seg25 seg29    显示第 4 位数据
;seg31 seg30    显示第 5 位数据
;前三位数据显示从 000-->111-->...FFF
;后两位数据为固定显示 dE
output
seg10:
        movla       d'5'           ;D'5'为 101 即为写命令
        movam       id
        bcm         portb,csp      ;片选位有效
        lcall       dispwriteid_
        movla       d'10'
        movam       addr           ;将欲显段地址送 HT1621
        lcall       dispwriteaddr_
```

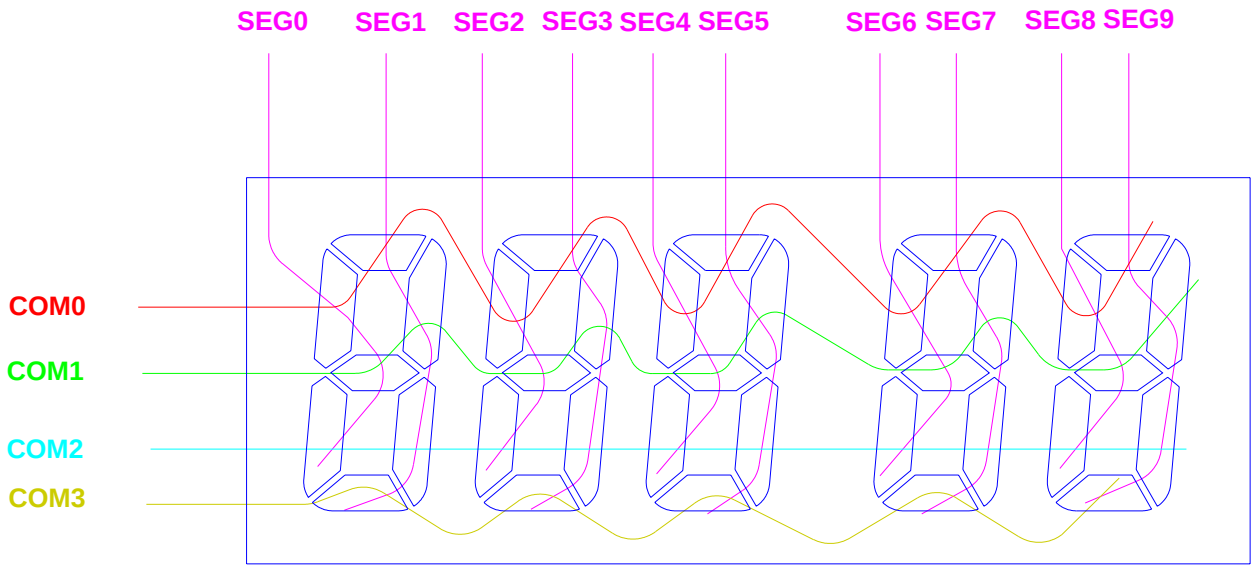
```
        movm    temp1,a
        lcall   lcd_tableh      ;将欲显数据高半位送 HT1621
        movam   dat             ;因为 HT1621 的 RAM 为四位长
        lcall   dispwritedata_  ;所以分两次送
        bsm     portb,csp
        nop
        nop
seg9:
        movla   d'5'
        movam   id
        bcm     portb,csp
        lcall   dispwriteid_
        movla   d'9'
        movam   addr
        lcall   dispwriteaddr_
        movm    temp1,a
        lcall   lcd_tablel      ;将欲显数据低半位送 HT1621
        movam   dat
        lcall   dispwritedata_
        bsm     portb,csp
        nop
        nop
seg8:
        movla   d'5'            ;以下显示位处理方法同上
        movam   id
        bcm     portb,csp
        lcall   dispwriteid_
        movla   d'8'
        movam   addr
        lcall   dispwriteaddr_
        movm    temp1,a
        lcall   lcd_tableh
        movam   dat
        lcall   dispwritedata_
        bsm     portb,csp
        nop
        nop
seg13:
        movla   d'5'
        movam   id
        bcm     portb,csp
        lcall   dispwriteid_
        movla   d'13'
        movam   addr
        lcall   dispwriteaddr_
        movm    temp1,a
        lcall   lcd_tablel
        movam   dat
```

```
        lcall   dispwritedata_
        bsm     portb,csp
        nop
        nop
seg17:
        movla   d'5'
        movam   id
        bcm     portb,csp
        lcall   dispwriteid_
        movla   d'17'
        movam   addr
        lcall   dispwriteaddr_
        movm    temp1,a
        lcall   lcd_tableh
        movam   dat
        lcall   dispwritedata_
        bsm     portb,csp
        nop
        nop
seg21:
        movla   '5'
        movam   d
        bcm     ortb,csp
        lcall   dispwriteid_
        movla   '21'
        movam   ddr
        lcall   dispwriteaddr_
        movm    emp1,a
        lcall   cd_tablel
        movam   at
        lcall   dispwritedata_
        bsm     ortb,csp
        nop
        nop
seg25:
        movla   '5'
        movam   d
        bcm     ortb,csp
        lcall   dispwriteid_
        movla   '25'
        movam   ddr
        lcall   dispwriteaddr_
        movla   dh
        lcall   cd_tableh
        movam   at
        lcall   dispwritedata_
        bsm     ortb,csp
        nop
```

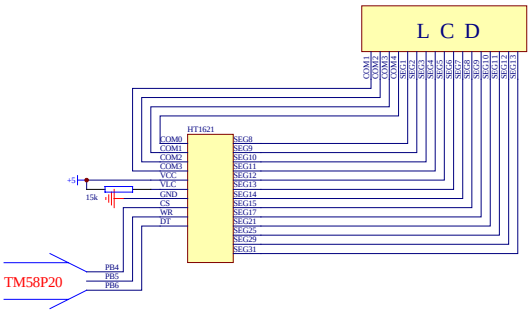
```
        nop
seg29:
        movla   '5'
        movam   d
        bcm     ortb,csp
        lcall   dispwriteid_
        movla   '29'
        movam   ddr
        lcall   dispwriteaddr_
        movla   dh
        lcall   cd_tablel
        movam   at
        lcall   dispwritedata_
        bsm     ortb,csp
        nop
        nop
seg31:
        movla   '5'
        movam   d
        bcm     ortb,csp
        lcall   dispwriteid_
        movla   '31'
        movam   ddr
        lcall   dispwriteaddr_
        movla   eh
        lcall   cd_tableh
        movam   at
        lcall   dispwritedata_
        bsm     ortb,csp
        nop
        nop
seg30:
        movla   '5'
        movam   d
        bcm     ortb,csp
        lcall   dispwriteid_
        movla   '30'
        movam   ddr
        lcall   dispwriteaddr_
        movla   eh
        lcall   cd_tablel
        movam   at
        lcall   dispwritedata_
        bsm     ortb,csp
        bcm     ortb,wrp
        bcm     ortb,dtp
        ret
;--------------------------
```

```
main:
        clrm        orta            ;I/O 口初始化
        movla       0h
        iodir       rta

        clrm        ortb
        movla       0h
        iodir       ortb

        clrm        ortc
        movla       0h
        iodir       ortc

        clrm        mr0
        movla       7h
        select
start:
        lcall       initdisp_       ;初始化 1621
        lcall       clearall_       ;清屏

        clrm        temp1
        lp0
        lcall       output          ;输出 000-->111...-->fff 后动态停机
        incm        temp1,  1
        movla       10h
        subam       temp1,  0
        btmss       status, c
        goto        lp0
aaa:
        lgoto       aaa
;---------------------------
        end
;---------------------------
```

LCD Layout


SEG0
SEG1
SEG2
SEG3
SEG4
SEG5
SEG6
SEG7
SEG8
SEG9
COM0
COM1
COM2
COM3

应用线路图



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