

HT48 & HT46 MCU 對 HT1621 的讀寫 -- 使用 C 語言

文件編碼：HA0103T

簡介

HT1621 是一款 128 個位元組的 LCD 控制元件，內部 RAM 直接對應 LCD 的顯示單元。相對應的軟體使它適用於包括 LCD 模組和顯示次系統在內的多功能應用。主控制器與 HT1621 介面只需 4 到 5 根線。內置的省電模式極大的降低了功耗。本文介紹 HT48R05A-1 微控制器來控制 HT1621，並介紹如何 Turn ON 及 Turn OFF LCD 所有位元。

應用說明

對於 HT1621 操作之前應該給它發送旗標碼，表明要求工作在哪種狀態。旗標的定義如下表：

操作	狀態	旗標碼
讀	資料	110
寫	資料	101
讀 - 修改 - 寫	資料	101
控制	命令	100

為了點亮 LCD，必須先給出兩個控制指令：SYSTEM ENABLE 和 LCD ON。SYSTEM ENABLE 指令碼是：1000000001X(X 為 Don't care bit)。LCD ON 指令碼是：1000000011X(X 為 Don't care bit)。操作結束可以用 SYSTEM DISABLE 來關閉 LCD。

由於是串列通信，資料應該先出現在 Data Input 腳位，然後給出一個寫入允許信號 (WR)，輸入一位元資料，接著輸入第二位元...直到全部寫入。

對 RAM 區不連續寫資料過程如下：

1	0	1	A5	A4	A3	A2	A1	A0	D0	D1	D2	D3	結束位	下一過程
---	---	---	----	----	----	----	----	----	----	----	----	----	-----	------

先發送旗標碼 101，表明下面要進行寫入操作。然後發送位址碼 A5~A0，用 D0~D3 指定相對應的位元，就可以對 LCD 相對應的位元操作了。讀的過程除了旗標位元不同，其餘類似。連續讀寫時，給出起始位址，操作結束位址自動加 1。

程式說明

本例介紹如何 Turn ON 及 Turn OFF LCD 全部位元組。程式流程如下：

系統初始化→1621 啟動→點亮 LCD 全部位元元→關閉 HT1621。

電路圖：參照 HT1621 的規格書。

```
#include <ht48r05a-1.h>
#define uchar unsigned char

#define BIAS 0x52
#define SYSEN 0x02
#define LCDON 0x06
#define LCDOFF 0x04

#define cs _pa0
#define wr _pa2
#define dat _pa3

void SendBit_1621(uchar data,uchar cnt) //data的高 cnt 位元寫入
HT1621，高位在前
{
    uchar i;
    for(i =0; i <cnt; i ++)
    {
        if((data&0x80)==0) dat=0;
        else dat=1;
        wr=0;
        _nop();
        wr=1;
        data<<=1;
    }
}

void SendDataBit_1621(uchar data,uchar cnt) //data的低 cnt 位寫入
HT1621，低位在前
{
    uchar i;
    for(i =0; i <cnt; i ++)
    {
        if((data&0x01)==0) dat=0;
        else dat=1;
        wr=0;
        _nop();
        wr=1;
        data>>=1;
    }
}
```

```

void SendCmd(uchar command)
{
    cs=0;
    SendBit_1621(0x80,4); //寫入旗標碼"100"和 9 位 command 命令，由於
    SendBit_1621(command,8); //沒有使有到更改時鐘輸出等命令，爲了編程方便
    cs=1; //直接將 command 的最高位寫"0"
}

void Write_1621(uchar addr,uchar data)
{
    cs=0;
    SendBit_1621(0xa0,3); //寫入旗標碼"101"
    SendBit_1621(addr,6); //寫入 addr 的高 6 位元
    SendDataBit_1621(data,4); //寫入 data 的低 4 位元
    cs=1;
}

void WriteAll_1621(uchar addr,uchar *p,uchar cnt)
{
    uchar i;
    cs=0;
    SendBit_1621(0xa0,3); //寫入旗標碼"101"
    SendBit_1621(addr,6); //寫入 addr 的高 6 位元
    for(i =0; i <cnt; i ++,p++) //連續寫入資料
    {
        SendDataBit_1621(*p,8);
    }
    cs=1;
}

void main()
{
    uchar a[10];
    a[0]=0x21;a[1]=0x43;a[2]=0x65;a[3]=0x87;a[4]=0xa9;
    _pac=0; //將口設爲輸出
    SendCmd(BIAS); //設置偏壓和工作週期
    SendCmd(SYSEN); //打開系統振盪器
    SendCmd(LCDON); //打開 LCD 偏壓發生器
    Write_1621(0x24,0x01); //0x24: (位址)的高 6 位有效，
    //0x01: (資料)的低 4 位元有效
    WriteAll_1621(0,a,5); //0: (起始位址)高 6 位有效，
    //a: (寫入資料的起始位址) 8 位元都有效，
    //6:爲寫入的位元組數。連續寫入"123456789a"
    SendCmd(LCDOFF); //關閉 LCD 顯示
}

```