



Reference Program for LCD Modules

Version No.

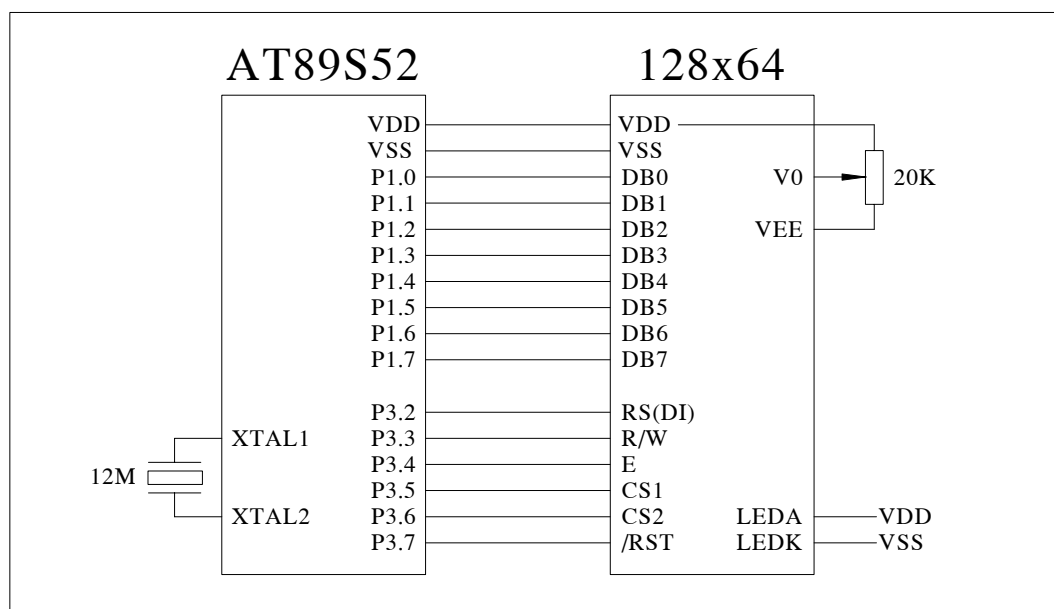
A00

Type

COB

Remark: 128x64 Graphic Type LCM, with S6B0108 or compatible IC

1. Interface



2. Instruction Code

Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Function
Display ON/OFF	L	L	L	L	H	H	H	H	H	L/H	Controls the display on or off. Internal status and display RAM data is not affected. L:OFF, H:ON
Set Address (Y address)	L	L	L	H	Y address (0~63)						Sets the Y address in the Y address counter.
Set Page (X address)	L	L	H	L	H	H	H		Page (0~7)		Sets the X address at the X address register.
Display Start Line (Z address)	L	L	H	H	Display start line (0~63)						Indicates the display data RAM displayed at the top of the screen.
Status Read	L	H	B U S Y	L	O N / O F F	R E S E T	L	L	L	L	Read status. BUSY L: Ready H: In operation ON/OFF L: Display ON H: Display OFF RESET L: Normal H: Reset
Write Display Data	H	L	Write Data								Writes data (DB0:7) into display data RAM. After writing instruction, Y address is increased by 1 automatically.
Read Display Data	H	H	Read Data								Reads data (DB0:7) from display data RAM to the data bus.

Remark: For the detail instruction for the control function, please refer to the related manual data book for controller IC.

3. Reference Program

```
//===== 89S52 MCU, 12M Oscillator =====  
#include <reg51.h>  
#include <intrins.h>  
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <math.h>  
#define uint unsigned int  
#define uchar unsigned char  
#define xchar unsigned char code  
  
sbit RS  = P3^2;  
sbit WRR = P3^3;  
sbit E   = P3^4;  
sbit CS1 = P3^5;  
sbit CS2 = P3^6;  
sbit RST = P3^7;  
sbit BF  = ACC^7;  
//-----  
xchar xshm[1024]={  
0xFF, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01, 0x01,  
0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0xFF, 0xFF, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x20, 0xF0, 0xF0, 0x10, 0x18, 0x18, 0x10, 0x00, 0x00,  
0x00, 0xFC, 0xFC, 0xF8, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xE0, 0xE0,  
0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0xC0, 0xC0, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0xE0, 0xE0,  
0xE0, 0x00, 0x00, 0x00, 0x00, 0x80, 0xF0, 0x70, 0x30, 0x00, 0x40, 0xC0, 0xC0, 0xF8, 0xF8, 0xF0,  
0xC0, 0xC0, 0xC0, 0x00, 0x00, 0xF8, 0xF8, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xF8, 0xF8, 0xF0, 0x00, 0x00, 0x00, 0x00, 0x00,  
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20,  
0x20, 0x20, 0x20, 0x20, 0x20, 0xA0, 0xE0, 0xE0, 0x70, 0x70, 0x20, 0x00, 0x00, 0x00, 0x00, 0xFF,  
0xFF, 0x00, 0x00, 0x04, 0x04, 0x04, 0xC4, 0xF4, 0xFF, 0xFF, 0xC4, 0xC6, 0x86, 0x36, 0x3C, 0x3F,  
0x03, 0xFF, 0xFF, 0xFF, 0xF0, 0x1C, 0x0E, 0x07, 0x03, 0x01, 0x00, 0x00, 0x00, 0x00, 0xFF, 0xFF,  
0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0xFF, 0xFF, 0x40, 0x40, 0x40, 0x40, 0x40, 0x40, 0xFF, 0xFF,  
0xFF, 0x00, 0x00, 0x02, 0x03, 0xC1, 0xF0, 0xFC, 0xFE, 0x0E, 0x04, 0x47, 0x4F, 0x4F, 0x47, 0x4F,
```

[illegible]

```

{
    WRR=1;
    do
    {
        P1=0xff;E=1;delayms(4);ACC=P1;E=0;
    }while(BF==1);
}
//-----
void wcomd_1(uchar wc1)
{
    CS1=1;CS2=0;RS=0;
    rbusy();
    WRR=0;P1=wc1;E=1;delayms(4);E=0;
}
//-----
void wdata_1(uchar wd1)
{
    CS1=1;CS2=0;RS=0;
    rbusy();
    RS=1;WRR=0;P1=wd1;E=1;delayms(4);E=0;
}
//-----
void wcomd_2(uchar wc2)
{
    CS1=0;CS2=1;RS=0;
    rbusy();
    WRR=0;P1=wc2;E=1;delayms(4);E=0;
}
//-----
void wdata_2(uchar wd2)
{
    CS1=0;CS2=1;RS=0;
    rbusy();
    RS=1;WRR=0;P1=wd2;E=1;delayms(4);E=0;
}
//-----
void initial()
{
    wcomd_1(0xc0);wcomd_2(0xc0);
    wcomd_1(0x3f);wcomd_2(0x3f);
}
//-----
void disp_all(uchar ch1,uchar ch2)
{
    uchar i,j;
    for(j=0;j<8;j++)
    {
        wcomd_1(0xb8+j);wcomd_2(0xb8+j);
        wcomd_1(0x40);wcomd_2(0x40);
    }
}

```

```

    for(i=0;i<32;i++)
    {
        wdata_1(ch1);wdata_2(ch1);
        wdata_1(ch2);wdata_2(ch2);
    }
}
}
//-----
void disp_bmp(xchar *str)
{
    uchar i,j;
    for(j=0;j<8;j++)
    {
        wcomd_1(0xb8+j);wcomd_2(0xb8+j);
        wcomd_1(0x40);wcomd_2(0x40);
        for(i=0;i<64;i++)
        {
            wdata_1(str[i+j*128]);
            wdata_2(str[i+64+j*128]);
        }
    }
}
//-----
////////// main //////////
void main(void)
{
    RST=0;delay(10);
    RST=1;delay(2);
    initial();
    while(1)
    {
        disp_bmp(xshm);      delay(100);
        disp_all(0xff,0xff);  delay(100);
        disp_all(0x00,0x00);  delay(100);
        disp_all(0x55,0xaa);  delay(100);
        disp_all(0xaa,0x55);  delay(100);
        disp_all(0xff,0x00);  delay(100);
        disp_all(0x00,0xff);  delay(100);
    }
}

```