

LM6065

液晶显示模块应用参考

深圳市拓普微科技发展有限公司

版本	描述	日期	编者
0.1	新版本	2006-01-20	周侃侃
0.2	修改 2.3 指令集：寄存器 F0H 的指令说明	2006-02-15	周侃侃



目 录

1 简介	3
2 应用	3
2.1 接口	3
2.2 指令操作	3
2.3 指令集	4
2.4 显示屏幕	5
2.5 软件流程图	5
2.6 应用举例	6

1 简介

LM6065模块是240×64全点阵STN液晶模组，内置LCD升压电路，白色背光，具有亮度高、对比度高，宽显示视角的特点。该模块内置RA8802控制器可内建512KByte的字型码，支持文字与绘图两种混和显示，可显示中文字型（内置简体字库GB 码）、数字符号、英日欧文等字母，使用者只要透过MCU对LM6065写入中/英文字型码，就可以直接在LCD 面板上显示中英文字型。大大节省了产品开发时间，使显示更方便。

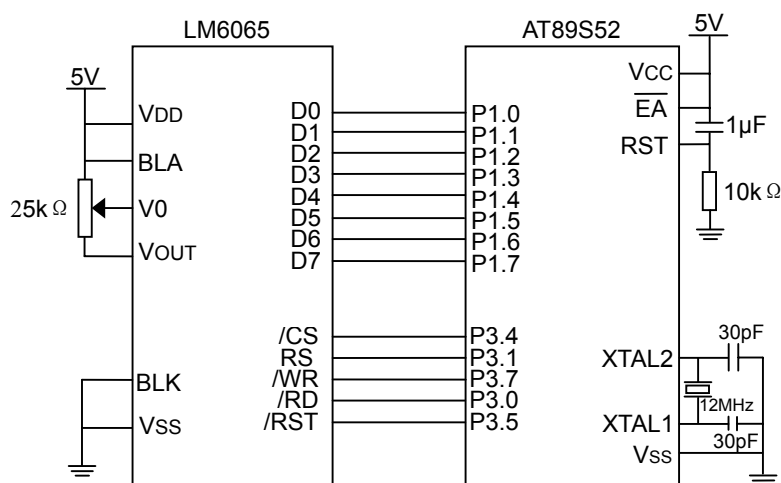
模块特点：

- ✧高对比度，STN 型 LCD 屏
- ✧高效率白色 LED 背光
- ✧内嵌繁/简体中文字库（7602 个汉字）
- ✧单 5V/3V 供电

2 应用

2.1 接口

以 8 位并行通信为例，模块与单片机 AT89S52 的接口如下图所示。



序号	符号
1	VOUT
2	V0
3	VSS
4	VDD
5	BUSY
6	NC
7	/CS
8	RS
9	/WR
10	/RD
11	D0
...	...
18	D7
19	/RST
20	NC
21	BLA
22	BLK

2.2 指令操作

本模块的操作是由对应的指令寄存器的值控制的，寄存器的地址和它的值共同完成了一个指令操作。一个完整的指令操作如下。

步骤	/RD	/WR	RS	DATA (D0~D7)
1	1	0	0	寄存器地址
2	1	0	0	寄存器的值

2.3 指令集

寄存器地址	寄存器名称	R/W	参数								指令说明		
			D7	D6	D5	D4	D3	D2	D1	D0			
00H	LCR	R/W	PW		SR	RTM	CG	DP	DK	DV	PW:电源模式 11=正常模式;10=等待模式;01=睡眠模式;00=关闭模式 SR:软件复位 1=寄存器全部复位(除 DDRAM);0=无效 RTM:当一条指令操作>2ms 时,自动重置 1=使能;0=无效 CG:文字/图形模式选择 1=文字模式;0=图形模式 DP:显示开关控制 1=开;0=关 DK:闪烁模式 1=全屏闪烁(时间由 BTR 设定);0=正常 DV:屏幕反白模式 1=正常;0=全屏反白显示		
08H	MIR	R/W	--	CKN	DISP	PLR	--	--	CKB		CKN:CLK_OUT 输出 1=使能;0=禁止 DISP:切换视窗模式 1=工作视窗;0=显示视窗 PLR:中断(INT)和忙位(BUSY)有效电平 1=高电平;0=低电平 CKB:时钟频率选择 00=1MHz;01=2MHz;10=4MHz;11=8MHz		
10H	CCR	R/W	ARI	ALG	WDI	WBC	AIX	CP	CK	CSD	ARI:光标自动移位(读状态) 1=使能;0=禁止 ALG:中/英文字对齐 1=使能;0=禁止 WDI:数据存于显示内存模式 1=正常;0=反相 WBC:设定粗体字 1=粗体;0=正常 AIX:光标自动移位(写状态) 1=使能;0=禁止 CP:光标显示开关 1=开;0=关 CK:光标闪烁设定 1=光标闪烁(时间由 BTR 设定);0=光标不闪烁 CSD:光标宽度 1=自动调整,半码/全码;0=8pix		
18H	CSCR	R/W	CR				DY				CR:光标高度 DY:行距设定		
20H	AWRR	R/W	--	--	X				A			X:工作视窗右边位置设定 A:显示视窗右边位置设定 = 显示列数/8 - 1	
28H	DWRR	R/W	--	--	Y				B			Y:工作视窗底边位置设定 B:显示视窗底边位置设定 = 显示行数-1	
30H	AWBR	R/W	SS				C				SS:工作视窗左边位置设定 C:显示视窗左边位置设定,通常设为:00H		
38H	DWBR	R/W	SC				D				SC:工作视窗顶边位置设定 D:显示视窗顶边位置设定,通常设为:00H		
40H	AWLR	R/W	--	--	RS				RC			RS:光标列地址设定 RC:光标行地址设定	
48H	DWLR	R/W	--	--	BT				CK			BT:光标闪烁时间设置,闪烁时间=设定值/帧频 CK:设定液晶工作时钟周期 = 系统频率 × 8 / (列数 × 行数 × 帧频)	
50H	AWTR	R/W	FD				TNS				BNK		FD:写入到显示内存的数据设置 TNS:字库转换 1=允许;0=禁止 BNK:内/外部字库选择 1=外部字库;0=内部字库
58H	DWTR	R/W	RM				FDA				ASC		RM:字库语种选择 01=繁体字库;10=简体字库 FDA:重复写入寄存器 E0H 的数据 FD 到 DDRAM 1=开始写入;0=未动作 ASC:文字码的类别选择 1=ASCII 码;0=数据>=A0H,GB/BIG5 码
60H	CPXR	R/W	ABS				00				01		ABS:ASCII 码区块选择 00=ASCII 码 0 区;01=ASCII 码 1 区; 10=ASCII 码 2 区;11=ASCII 码 3 区
70H	CPYR	R/W	RM				FDA				ASC		RM:字库语种选择 01=繁体字库;10=简体字库 FDA:重复写入寄存器 E0H 的数据 FD 到 DDRAM 1=开始写入;0=未动作 ASC:文字码的类别选择 1=ASCII 码;0=数据>=A0H,GB/BIG5 码
80H	BTR	R/W	BT				CK				FD		BT:光标闪烁时间设置,闪烁时间=设定值/帧频 CK:设定液晶工作时钟周期 = 系统频率 × 8 / (列数 × 行数 × 帧频) FD:写入到显示内存的数据设置
90H	SCCR	R/W	RM				FDA				ASC		RM:字库语种选择 01=繁体字库;10=简体字库 FDA:重复写入寄存器 E0H 的数据 FD 到 DDRAM 1=开始写入;0=未动作 ASC:文字码的类别选择 1=ASCII 码;0=数据>=A0H,GB/BIG5 码
E0H	PDR	R/W	RM				FDA				ASC		RM:字库语种选择 01=繁体字库;10=简体字库 FDA:重复写入寄存器 E0H 的数据 FD 到 DDRAM 1=开始写入;0=未动作 ASC:文字码的类别选择 1=ASCII 码;0=数据>=A0H,GB/BIG5 码
F0H	FCR	R/W	TNS	BNK	RM	FDA	ASC	ABS			TNS:字库转换 1=允许;0=禁止 BNK:内/外部字库选择 1=外部字库;0=内部字库 RM:字库语种选择 01=繁体字库;10=简体字库 FDA:重复写入寄存器 E0H 的数据 FD 到 DDRAM 1=开始写入;0=未动作 ASC:文字码的类别选择 1=ASCII 码;0=数据>=A0H,GB/BIG5 码 ABS:ASCII 码区块选择 00=ASCII 码 0 区;01=ASCII 码 1 区; 10=ASCII 码 2 区;11=ASCII 码 3 区		

备注:

- *1 建议在写入控制数据前检查控制器忙位;
- *2 更多显示控制细节,请查阅 RAIO 8802 技术手册。

2.4 显示屏幕

1,1 (D7)	2,1 (D6)	3,1 (D5)	4,1 (D4)	5,1 (D3)	236,1 (D4)	237,1 (D3)	238,1 (D2)	239,1 (D1)	240,1 (D0)
1,2 (D7)	2,2 (D6)	3,2 (D5)	4,2 (D4)	5,2 (D3)	236,2 (D4)	237,2 (D3)	238,2 (D2)	239,2 (D1)	240,2 (D0)
1,3 (D7)	2,3 (D6)	3,3 (D5)	4,3 (D4)	5,3 (D3)	236,3 (D4)	237,3 (D3)	238,3 (D2)	239,3 (D1)	240,3 (D0)
:	:	:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:	:	:
:	:	:	:	:	:	:	:	:	:	:
1,62 (D7)	2,62 (D6)	3,62 (D5)	4,62 (D4)	5,62 (D3)	236,62 (D4)	237,62 (D3)	238,62 (D2)	239,62 (D1)	240,62 (D0)
1,63 (D7)	2,63 (D6)	3,63 (D5)	4,63 (D4)	5,63 (D3)	236,63 (D4)	237,63 (D3)	238,63 (D2)	239,63 (D1)	240,63 (D0)
1,64 (D7)	2,64 (D6)	3,64 (D5)	4,64 (D4)	5,64 (D3)	236,64 (D4)	237,64 (D3)	238,64 (D2)	239,64 (D1)	240,64 (D0)

显示屏幕（顶视图）

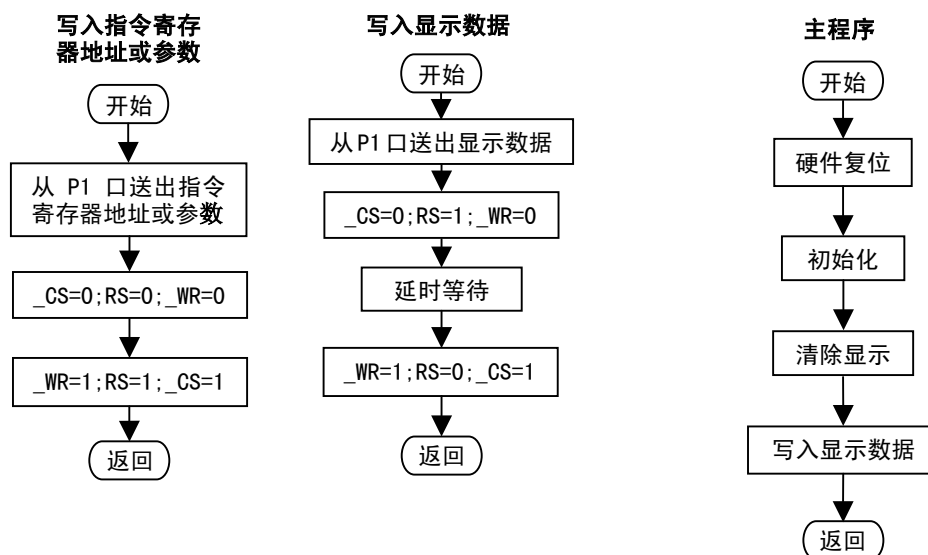
点 1,1(x,y)是屏幕左上角的顶点；

点 240,64(x,y)是屏幕右下角的顶点。

备注：

*1 想进一步了解显示屏幕，请查阅 RAI0 8802 技术手册。

2.5 软件流程图



2.6 应用举例

参考程序

```
//本程序为 LM6065 模块的基本驱动程序
//显示结果为我公司文本 LOGO
#include <stdio.h>
#include <reg52.h>
#include <intrins.h>

#define uchar unsigned char//0~255
#define uint unsigned int //0~65535

#define LCDBUS      P1
sbit  _RES   =  P3^5;
sbit  RS     =  P3^1;
sbit  _WR    =  P3^7;
sbit  _RD    =  P3^0;
sbit  _CS    =  P3^4;

//-----
// 显示数据
//-----
// TOPWAY Text LOGO
uchar code TextLogo[]={
"深圳市拓普微科技开发有限公司  "
" TOPWAY Technology Co., Ltd.  "
"ABCDEFGHIJKLMNOPQRSTUVWXYZABCD"
"123456789012345678901234567890"
};

//-----
//延时
//-----
void delaysms(uint m)          //时钟频率 12MHz 时, 大约 1ms
{
    uint j;
    uint i;

    for(i=0; i<m; i++)
        for(j=0; j<109; j++)
            _nop_();
}

//-----
//指令和数据操作
//-----
void SdCmd(uchar Command)    //送指令
{
    _WR = 1;                //设置控制信号
    _RD = 1;
    RS = 0;                 //指令通道
    LCDBUS= Command;
    _CS = 0;               //打开片选
    _nop_();
    _WR = 0;
}
```

```

    _nop_();
    _WR = 1;
    _nop_();
    _CS = 1;           //关片选
    _nop_(); _nop_();
    _nop_(); _nop_();
    _nop_(); _nop_();
    _nop_(); _nop_();
}

void SdData(uchar DData) //写显示数据
{
    _WR = 1;           //设置控制信号
    _RD = 1;
    RS = 1;           //选择数据通道
    LCDBUS = DData;
    _CS = 0;           //打开片选
    _nop_();
    _WR = 0;
    _nop_();
    _WR = 1;
    _nop_();
    _CS = 1;           //关片选
    _nop_(); _nop_();
    _nop_(); _nop_();
    _nop_(); _nop_();
    _nop_(); _nop_();
}

//-----
//显示文本
//-----
void WriteTextScreen(uchar *TxtData)
//显示数据为(15x2)x4 = 120byte
//设置为文本显示
{
    uchar TempData;
    uchar i,j;
    SdCmd(0x60); SdCmd(0x00); //设置x地址为0
    SdCmd(0x70); SdCmd(0x00); //设置y地址为0
    SdCmd(0x00); SdCmd(0xCD); //正常供电模式,文本显示,
    //显示开,正常模式

    for (j=0; j<4; j++)
    {
        for(i=0; i<30; i++)
        {
            TempData=*(TxtData+(j*30)+i);
            SdData(TempData);
        }
    }
}

//-----
//初始化模块
//-----
void initLCDM(void)

```



```
{
    _RES    = 1;
    _RES    = 0;
    delays(250);
    _RES    = 1;
    delays(100);

    SdCmd(0x00);SdCmd(0xCD); //LCD 控制寄存器 Controller Register (LCR)
    SdCmd(0x08);SdCmd(0x73); //通用寄存器 (切换窗口模式等) Misc. Register (MIR)
    SdCmd(0x10);SdCmd(0xE9); //光标控制寄存器 Cursor Control Register (CCR), 字体相关
    SdCmd(0x18);SdCmd(0x20); //光标大小控制寄存器 Cursor Size Control Register (CSCR)
    SdCmd(0x20);SdCmd(0x1D); //活动窗右边界寄存器 Active Window Right Register (AWRR)
    SdCmd(0x28);SdCmd(0x1D); //显示窗右边界寄存器 (240 列) Display Window Right Register (DWRR)
    SdCmd(0x30);SdCmd(0x3F); //活动窗底部边界寄存器 Active Window Bottom Register (AWBR)
    SdCmd(0x38);SdCmd(0x3F); //显示窗底部边界寄存器 (64 行) Display Window Bottom Register (DWBR)
    SdCmd(0x40);SdCmd(0x00); //活动窗左边界寄存器 Active Window Left Register (AWLR)
    SdCmd(0x50);SdCmd(0x00); //活动窗顶边界寄存器 Active Window Top Register (AWTR)
    SdCmd(0x48);SdCmd(0x00); //显示窗左边界寄存器 Display Window Left Register (DWLR)
    SdCmd(0x58);SdCmd(0x00); //显示窗口顶部寄存器 Display Window Top Register (DWTR)
    SdCmd(0x60);SdCmd(0x00); //X 方向光标寄存器 Cursor Position X Register (CPXR)
    SdCmd(0x70);SdCmd(0x00); //Y 方向光标寄存器 Cursor Position Y Register (CPYR)
    SdCmd(0x80);SdCmd(0x23); //闪烁时间寄存器 Blink Time Register (BTR)
    SdCmd(0x90);SdCmd(0x3E); //移位时钟控制寄存器 Shift Clock Control Register (SCCR)
    SdCmd(0xE0);SdCmd(0x00); //数据模式寄存器 Pattern Data Register (PDR)
    SdCmd(0xF0);SdCmd(0xA0); //字体控制寄存器 Font Control Register (FCR)
    delays(20);
}

//-----
//主程序
//-----
void main()
{
    SP=0x60;
    EA=0;           //关闭中断
    _CS=1;         //设置控制信号
    _RD=1;
    _WR=1;
    RS=1;
    LCDBUS=0xff;   //拉高数据口
    initLCDM();

    while(1)
    {
        WriteTextScreen(TextLogo);
        delays(2000);
    }
} //end of program
```