

# LM24064K

## 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	2006-08-04	淮俊霞



## 目 录

1 简介 .....	3
2 应用 .....	3
2.1 接口 .....	3
2.2 指令操作 .....	3
2.3 主要寄存器及参数表 .....	4
2.4 参数总结 .....	5
2.5 显示像素映射图 .....	6
2.6 流程图 .....	6
参考程序.....	7

## 1 简介

本公司生产的 LM24064K 系列产品为 240×64 点阵的图形显示模块，内置 RA8803 控制器。模块不仅可以显示单一的文本、图形，而且可以实现双图层的（“或”、“异或”、“同或”、“与”四种逻辑关系）合成显示和四阶灰度的效果。文本模式下能够实现大小字体的混编（最大字体为 64×64），中英文自动对齐，行距设定等功能。

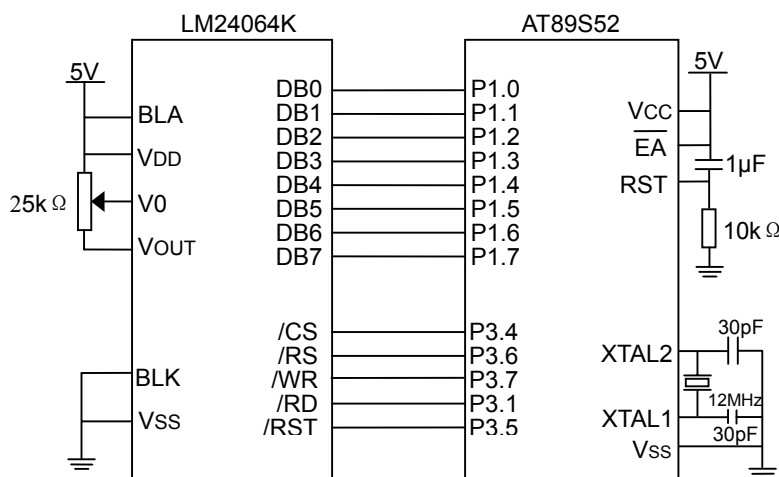
模块特点：

- ✧单电源供电，内置升压电路
- ✧白色 LED 背光
- ✧内嵌简体中文字库（7602 个汉字）
- ✧双图层显存（2×9.6k 显示存储器）
- ✧可自定义 16 个 16×16 的字符

## 2 应用

### 2.1 接口

模块与单片机 AT89S52 的接口，采用 I/O 方式的 8 位并行通信，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	DB0
...	...
18	DB7
19	/RST
20	NC
21	BLA
22	BLK

\*注：致能 BUSY 输出需设置寄存器[A0H]的 Bit3:0 任一为"1"

### 2.2 指令操作

操作	控制状态			指令代码							
	RS	/RD	/WR	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
写寄存器命令	0	1	0	寄存器地址码							
				参数							
读寄存器数据命令	0	0	1	寄存器地址码							
数据写操作	1	1	0	写数据							
数据读操作	1	0	1	读数据							



## 2.3 主要寄存器及参数表

寄存器	缺省值	指令参数								
		DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	
00H	C9H	电源模式(Power Mode) 11: 正常模式 00: 关闭模式		软件复位 1: 复位 0: 正常模式		0	显示模式 1: 文本模式 0: 图形模式	显示开关 1: 开 0: 关	屏幕闪烁 1: 是 0: 否	显示类型 1: 正向 0: 负向
01H	F0H	1	1	1	1	0	0	时钟速度选择 00: 3MHZ 01: 4MHZ 10: 8MHZ 11: 12MHZ		
03H	80H	1	0	0	0	高低位反转 显示数据 1: 是 0: 否	垂直移动 1: 允许 0: 禁止	水平移动 1: 允许 0: 禁止	平移模式 1: 水平移动 0: 垂直移动	
10H	6FH	读数据时, 光标自动移位 1: 是 0: 否	中英文对齐 1: 是 0: 否	正反相存储 数据选择 1: 正相 0: 反相	字体设置 1: 粗体 0: 正常	写数据时, 光标自动移位 1: 是 0: 否	光标显示 1: 开 0: 关	光标闪烁 1: 是 0: 否	光标宽度 1: 8 或 16 0: 固定 8	
11H	22H	光标的高度				行距				
12H	91H	图形模式下, 光标自动移位方向选择 1: 向右 0: 向下	图层显示模式选择 001: 只显示 Page 1 (单一上层显示) 010: 只显示 Page 2 (单一下层显示) 011: 双图层显示 000: 灰阶显示			在双图层模式下, Page1 与 Page 2 的逻辑关系 00: "OR" 01: "XOR" 10: "NOR" 11: "AND"		选择读写操作的图层 00: Page 0 (512B SRAM) 01: Page 1 (9.6KB SRAM) 10: Page 2 (9.6KB SRAM) 11: 同时存取两层		
20H	27H	0	0	设定工作视窗右边位置						
21H	27H	0	0	设定显示视窗右边位置						
30H	EFH	设定工作视窗底部位置								
31H	EFH	设定显示视窗底部位置								
40H	00H	0	0	设定工作视窗左边位置						
41H	00H	0	0	设定显示视窗左边位置						
50H	00H	设定工作视窗顶部位置								
51H	00H	设定显示视窗顶部位置								
60H	00H	0	0	设定光标 X 地址						
70H	00H	设定光标 Y 地址								
71H	00H	水平移动时, 块移动的起始行								
72H	EFH	水平移动时, 块移动的结束行								
81H	00H	0	1	0	0	0	0	0	0	
90H	04H	设定移位时钟								
E0H	00H	(1) 图形模式下, 若寄存器[F0]Bit3 为 1, 本寄存器的数据会被全部填写到 DDRAM 内, 之后[F0]Bit3 被置 0。 (2) 灰阶模式下, 用来控制灰阶的显示效果, "0" 与 "1" 的数目代表显示的比率。								
F0H	92H	1	0	字型 ROM 的选择 00: 简体字型(256KB) 01: 繁体字型(512KB) 10: 简体字型(512KB)		写资料到 DDRAM 1: 开始写入 0: 未动作	文本模式下 强制 ASCII 解码 1: 是 0: 否	4 种 ASCII 区块选择 00: 区块 0 01: 区块 1 10: 区块 2 11: 区块 3		
F1H	0FH	字型水平方向的大小 00: 一倍 01: 二倍 10: 三倍 11: 四倍		字型垂直方向的大小 00: 一倍 01: 二倍 10: 三倍 11: 四倍		1	1	1	1	

- 注: 1. 更详细的寄存器及其参数说明请参考 RA8803 控制器的应用手册  
2. 为了获得更好的显示效果, 建议设置寄存器[81H]的值为 40H

## 2.4 参数总结

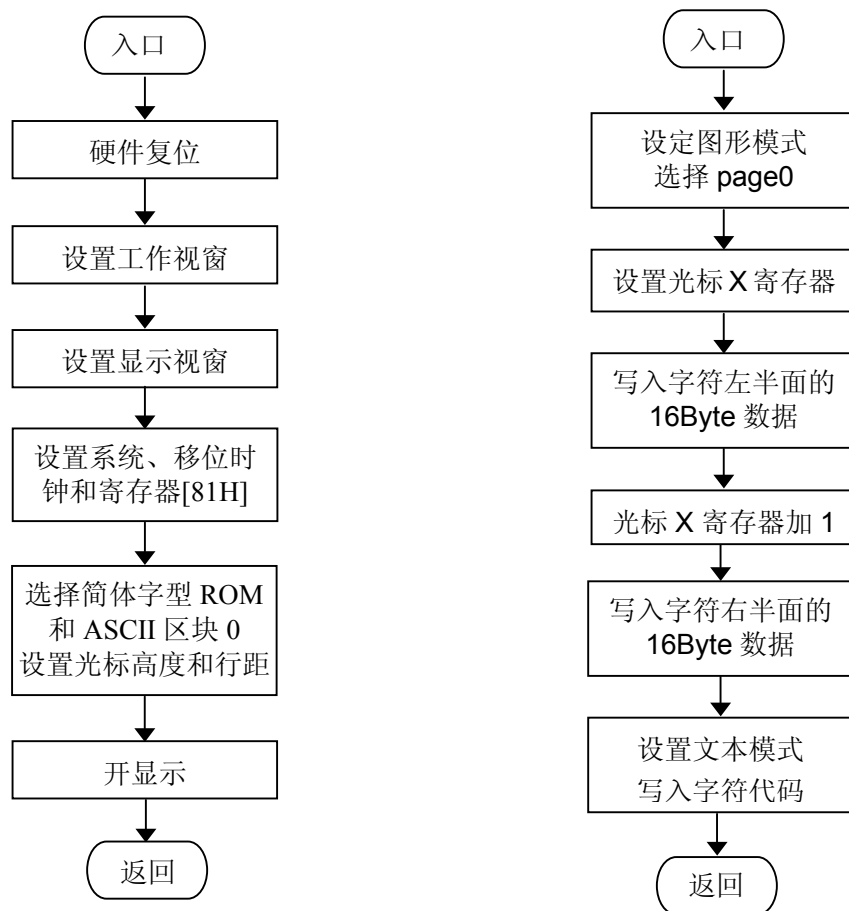
单页显示	双图层显示
选择页码 [12H]=0x91 或 0x11 // Page 1 [12H]= 0xA2 或 0x22 // Page 2 设定显示模式 [00H]=0xCD // 文本模式 [00H]=0xC5 // 图形模式 定位光标位置 [60H]=0x** [70H]=0x** 写入数据 (1) 图形模式下, 直接写入数据 (2) 文本模式下, 先写入代码的高字节, 再写入代码的低字节  <b>图形功能:</b> 1) 光标自动移位方向: [12H] Bit7 位 2) 自动写入数据: [E0H]、[F0H]Bit3 位  <b>文本功能:</b> 1) 文本的设置 行距选择: [11H] Bit3:0 位 字体选择: [10H] Bit4 位 字型大小: [F1H] Bit7:4 位 中/英文文字对齐: [10H] Bit6 位 字符负向显示: [10H] Bit5 位 字符高低位反转: [03H] Bit3 位 水平移动: [03H] Bit1:0 位; [71H]、[72H] 可以设置 任意两 Common 区块之间的移动 垂直移动: [03H] Bit2、0 位  2) 光标的设置 定位: [60H]、[70H] 光标显示: [10H] Bit2 位 光标闪烁: [10H] Bit1 位 光标宽度: [10H] Bit0 位 光标高度: [11H] Bit7:4 位	选择 page1 进行读写操作 [12H]=0x91 或 0x11 设定显示模式 [00H] Bit3 位 定位光标位置 [60H]=0x** [70H]=0x** 写入数据  选择 page2 进行读写操作 [12H]= 0xA2 或 0x22 设定显示模式 [00H] Bit3 位 定位光标位置 [60H]=0x** [70H]=0x** 写入数据  <b>两图层“或”关系</b> [12H]=0xB3 //Page1 RAM “OR” Page2 RAM <b>两图层“异或”关系</b> [12H]=0xB7 //Page1 RAM “XOR” Page2 RAM <b>两图层“同或”关系</b> [12H]=0xBB //Page1 RAM “NOR” Page2 RAM <b>两图层“与”关系</b> [12H]=0xBF //Page1 RAM “AND” Page2 RAM  <b>灰阶显示:</b> 提高系统时钟 [01H] Bit1:0 位 灰阶对比设置 [E0H]=0x** 提高帧频 [90H]=0x** 显示灰阶 [12H]=0x00 关显示 [00H] Bit2 位 开显示 [00H] Bit2 位 延迟 1 秒

## 2.5 显示像素映射图

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)	...	...	236,239 (D4)	237,239 (D3)	238,239 (D2)	239,239 (D1)	240,239 (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)

上图中 1,1(x,y)对应于显示屏左上角像素的位置, 240,64(x,y)对应于显示屏右下角像素的位置。

## 2.6 流程图



初始化流程图

自定义字符流程图



## 参考程序

```
//-----  
//本程序的演示结果为:  
//“拓普微 LM24064K”  
//-----  
#include <reg52.h>  
#include <intrins.h>  
  
#define uchar unsigned char //0~255  
#define uint unsigned int //0~65535  
  
#define lcd_bus P1  
sbit _CS = P3^4;  
sbit RS = P3^6;  
sbit _WR = P3^7;  
sbit _RD = P3^1;  
sbit _RST = P3^5; //端口设置  
  
//-----  
//延时子程序  
//-----  
void Delay(uint m)  
{  
    uint i;  
    uint j;  
    for(i=0;i<m;i++)  
        for(j=0;j<109;j++)  
            _nop_();  
}  
  
//-----  
//写入寄存器地址或参数  
//-----  
void SdCmd(uchar command)  
{  
    lcd_bus = command;  
    _CS = 0;  
    RS = 0;  
    _WR = 0;  
    _RD = 1;  
    RS = 1;  
    _CS = 1;  
}  
  
//-----  
//写入数据  
//-----  
void SdData(uchar ddata)  
{  
    lcd_bus = ddata;  
    _CS = 0;  
    RS = 1;  
    _WR = 0;  
    _nop_();_nop_();_nop_();  
    _WR = 1;  
    RS = 0;  
    _CS = 1;  
}  
  
//-----  
//显示文本  
//-----  
void ShowText(uchar *text)  
{  
    while(*text>0)  
    {  
        SdData(*text);  
        text++;  
    }  
}
```



```
//-----  
//初始化  
//-----  
void Init()  
{  
    _RST= 0;  
    Delay(250);  
    _RST= 1;  
    Delay(100);  
  
    SdCmd(0x20);SdCmd(0x1d);          //设置工作视窗  
    SdCmd(0x30);SdCmd(0x3f);  
    SdCmd(0x40);SdCmd(0x00);  
    SdCmd(0x50);SdCmd(0x00);  
  
    SdCmd(0x21);SdCmd(0x1d);          //设置显示视窗  
    SdCmd(0x31);SdCmd(0x3f);  
    SdCmd(0x41);SdCmd(0x00);  
    SdCmd(0x51);SdCmd(0x00);  
  
    SdCmd(0x01);SdCmd(0xf0);          //系统时钟  
    SdCmd(0x90);SdCmd(0x15);          //移位时钟  
    SdCmd(0x81);SdCmd(0x40);          //寄存器[81H]  
    SdCmd(0xf0);SdCmd(0xa0);          //简体中文, ASCII 0  
    SdCmd(0x11);SdCmd(0x00);          //光标高度和行距  
    SdCmd(0x00);SdCmd(0xcd);          //文本模式, 开显示  
}  
  
//-----  
//清屏  
//-----  
void CleanScreen()  
{  
    uchar i,j;  
    SdCmd(0x60);SdCmd(0x00);  
    SdCmd(0x70);SdCmd(0x00);  
    for(i=0;i<64;i++)  
        {  
            for(j=0;j<30;j++)  
                SdData(0x00);  
        }  
}  
  
//-----  
//主程序  
//-----  
main()  
{  
    lcd_bus=0xff;  
  
    _CS    = 1;  
    _RD    = 1;  
    _WR    = 1;  
    RS     = 1;  
  
    Init();  
    CleanScreen();  
    SdCmd(0xf1);SdCmd(0x1f);          //设置显示字型大小  
    SdCmd(0x60);SdCmd(0x07);          //设置显示起始位置 X 坐标  
    SdCmd(0x70);SdCmd(0x0f);          //设置显示起始位置 Y 坐标  
    ShowText("拓普微 LM24064K ");    //显示汉字  
    Delay(100);  
    while(1);                          //end of program  
}
```