

OLED 显示模块与 C8051F 单片机的接口设计

摘要: OLED 作为新一代显示技术, 广泛用于各种仪器仪表的显示终端, 实时显示字符、汉字、曲线等信息。文中介绍一种点阵式 OLED 模块 VGS12864E 的结构特征、指令系统; 给出它与 Cygnal C8051F020 单片机的间接访问接口电路设计, 以及显示模块的硬件驱动和显示 16×8 点阵西文字符的 Keil C51 程序代码, 并对相关代码进行注释。由于此款显示模块的指令系统与液晶显示驱动控制器 HD61202 兼容, 故程序代码也可作为指令系统与它兼容的液晶显示编程的参考。

关键词: 点阵显示模块 OLED C8051F 接口设计

有机发光显示 OLED (Organic Light Emitting Display) 是比液晶显示技术更为先进的新一代平板显示技术, 是被业界公认为最具发展前景的下一代显示技术。它与液晶显示技术相比, 具有超轻薄、高亮度、广视角、自发光、响应速度快、适应温度范围宽、抗震强、功耗低、可实现柔软显示等优越性能, 可广泛应用于通信、计算机、消费电子、工业应用、商业、交通等领域。下面以 VGS12864E 显示模块为例, 介绍 C8051F020 单片机与它的接口设计及软件编程方法。

1 VGS12864E 显示模块

VGS12864E 是 128×64 行点阵的 OLED 单色、字符、图形显示模块。模块内藏 64×64 的显示数据 RAM, 其中的每位数据都对应于 OLED 屏上一个点的亮、暗状态; 其接口电路和操作指令简单, 具有 8 位并行数据接口, 读写时序适配 6800 系列时序, 可直接与 8 位微处理器相连; 与 Intel 8080 时序的 MCU 连接时需要进行时序转换。

2 显示模块结构

2.1 模块框图

VGS12864E 显示模块显示屏为 128 列、64 行, 使用 1 片有 64 行输出的行驱动器和 2 片列驱动控制器, 其中每片列驱动器有 64 路输出。行驱动器与 MCU 没有关系, 只要提供电源就能产生驱动信号和同步信号, 模块的外部信号仅与列驱动器有关。列驱动器内置 64×64 位显示存储器, RAM 被分为 8 页, 每页 8 行; 显示屏上各像素点显示状态与显示存储器各位数据一一对应, 显示存储器的数据直接作为图形显示的驱动信号, 为“1”显示, 为“0”不显示。图 1 为模块的逻辑电路接口框图。

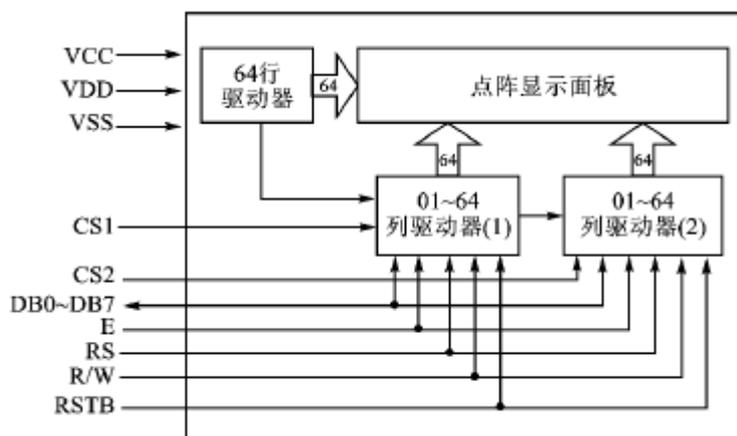


图 1VGS12864E 逻辑框图

2.2 模块引脚功能及指令系统

模块引脚功能如表 1 所列。模块的指令系统与液晶显示驱动控制器 HD61202 兼容，共有 7 条指令。这里不作详细描述，仅列出表 2 指令列表。其中，前两条为显示状态设置类指令，其余的为读写操作类指令。

表 1VGS12864E 引脚功能

引脚序号	引脚符号	功能
1	GND	逻辑电路电源地(0V)
2	VCC	逻辑电路正电源(5V)
3	VDD	OLED 驱动电源(9V)
4	RS(输入)	数据/指令寄存器选择
5	R/W(输入)	读/写信号选择
6	E(输入)	使能 OLED
7~14	DB0~DB7(三态)	数据总线
15	CS1(输入)	左半屏片选
16	CS2(输入)	右半屏片选
17	RSTB(输入)	复位
18~20	NC	无

表 2VGS12864E 指令列表

指令名称	W	RS	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
显示开/关	0	0	0	0	1	1	1	1	1	1/0
显示起始行	0	0	1	1	显示起始行 0~63					
页设置	0	0	1	0	1	1	1	页号 0~7		
列地址设置	0	0	0	1	显示列地址 0~63					
读状态	1	0	BUSY(1)	0	On(1)/Off(0)	REST(1)	0	0	0	0
写数据	0	1	写数据							
读数据	1	1	读显示数据							

3 显示模块与 Cygnal 单片机硬件接口设计

VGS12864E 的接口连接方式有两种:一种是直接访问方式,另一种是间接访问方式。不论哪种方式,要访问模块都必须先读取状态寄存器内容,判断“忙”标志,不忙时才可以访问。直接访问方式是将模块接口作为存储器或 I/O 设备直接挂在 MCU 总线上,MCU 以访问存储器或 I/O 设备方式对模块进行操作。间接访问方式是 MCU 通过软件模拟控制时序对模块进行操作。这里介绍的是 Cygnal C8051F020 单片机与 VGS12864E 的间接访问接口设计。

C8051F020 是美国 Cygnal 公司推出的一种混合信号 SoC 型 8 位单片机,是集成度很高的混合信号系统级的芯片。它具有 100 脚的 TQFP 封装,功耗低,供电电压为 2.7~3.3 V,全部 I/O、RST、JTAG 引脚均耐 5 V 电压;有高速、流水线结构的 8051 兼容的 CIP51 内核(可达 25 MIPS)。该 MCU 具有 P0~P7 共 64 个通用 I/O 端口,每个端口引脚都可以被配置为推挽输出或漏级开路输出。对于 VGS12864E,由于其工作电压是 5 V,而 C8051F020 的工作电压是 3.3 V,所以要 C8051F020 的输出能更好地驱动 5 V 输入的 OLED,需要对系统进行额外配置。除了将对应端口的输出方式设置为“漏极开路”外,还应在电路上将每个端口通过一个上拉电阻接到 5 V 电源,这样可以保证 C8051F020 的逻辑“1”输出能够被提升到 5 V。接口电路如图 2 所

示。

4 软件编程

软件编程采用 Keil C51 语言, 包括显示模块硬件的驱动程序(即写指令和写数据), 显示模块初始化和清屏等通用子程序以及西文字符的显示实现程序。由于此模块指令系统与液晶显示驱动控制器 HD61202 兼容, 故这些程序具有较高的通用性。对于字符汉字显示, 该模块的字符库数据特点是以列数据形式编制, 即 1 个字节数据表示 1 列 8×1 的数据, 和通常的字符库相比, 该字符是旋转了 90° 的字模数据。使用 Zimo21.exe 取字模软件, 并设置提取方式为纵向取模, 可以很方便地取得所需的中西文字模。由于该显示模块的每一列 8×1 的数据是低位在前, 高位在后, 为进一步实现中西文的正确显示, 还需对通过上述方法取得的字模的每个字节的高低位进行一次对调, 程序中使用了 一个数组 UpsetChar[] 来实现。程序中使用的西文字符是 16×8 点阵, 汉字是 16×16 点阵。

Cygnal 单片机通过间接访问方式控制 OLED。其 I/O 端口需要进行配置, 配置如下:

```
void C8051F020_output_config() {
    P2MDOUT = 0x00; //配置 P2.5(lcd_wr), P2.6(lcd_rd),
                  //P2.7(lcd_rs)为推挽输出方式
    P74OUT = 0xf3; //配置 P5.0~P5.7(lcd_d0 lcd_d7)
                  //为推挽输出方式
}
```

对显示模块进行写指令操作和写数据操作分为写左半屏和写右半屏。写左右半屏的差别仅在于置位相应的片选信号, 而写指令与写数据的差别在于写数据时置位 RS 寄存器选择信号。初始化操作完成显示位置的确定和打开显示。现以写左半屏命令和写左半屏数据为例, 说明各操作函数。

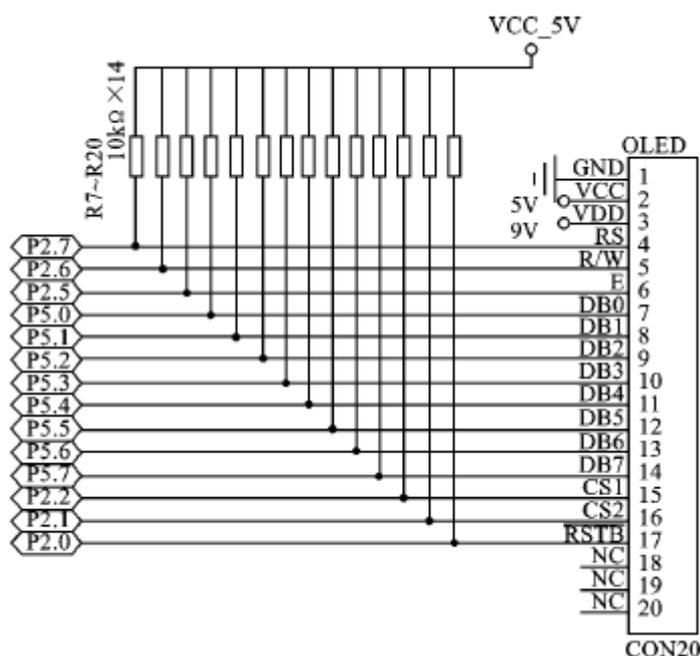


图 2C8051F020 与 VGS12864E 间接访问方式接口电路

(1) 写左半屏命令

```

void wr_command1() {
    cs1=1; cs1=1; cs1=1;    //选择左半屏
    cs2=0; cs2=0; cs2=0;
    read_status();        //读 BUSY 位状态
    r_w=0; r_w=0; r_w=0;
    P5=com;                //将命令字节送 I/O 口
    e=1; e=1; e=1;
    e=0; e=0; e=0;        //在 E 下降沿, 命令字节被写
                           //入列驱动器
}

```

(2) 写左半屏数据

```

void wr_data1() {
    cs1=1; cs1=1; cs1=1;    //选择左半屏
    cs2=0; cs2=0; cs2=0;
    read_status();        //读 BUSY 位状态
    d_i=1; d_i=1; d_i=1;
    r_w=0; r_w=0; r_w=0;
    P5=dat;                //将数据字节送 I/O 口
    e=1; e=1; e=1;
    e=0; e=0; e=0;        //在 E 下降沿, 数据字节被写
                           //入列驱动器
}

```

其中, 读状态位的函数采用查询标志位的方式, 即

```

void read_status() reentrant {
    uchar busy;
    uchar temp;
    d_i=0; d_i=0; d_i=0;    //进行指令操作
    r_w=1; r_w=1; r_w=1;    //进行读操作
    do {
        P5=0xff;
        e=1; e=1; e=1;
        busy=P5;            //读入 P5 端口状态
        e=0; e=0; e=0;
        temp=busy&0x80;
    }while(temp!=0);
}

```

(3) OLED 初始化

```

void init_lcd() {
    com=0xc0;                //从第 0 行开始
    wr_command1();
    wr_command2();
}

```

```
com=0x3f;          //打开显示
wr_command1();
wr_command2();
}
```

(4) OLED 清屏

```
void clear_lcd() {
    uchar column1;
    uchar page;
    for(page=0;page<8;page++) {
        com=(0xb8+page); //设置页号
        wr_command1();
        wr_command2();
        com=0x40;        //设置起始列为 0, 写操作完
                        //后列地址计数器自动加 1

        wr_command1();
        wr_command2();
        for(column1=0;column1<64;column1++) {
            //清左半屏

            dat=0;
            wr_data1();
        }
        for(column1=64;column1<128;column1++) {
            //清右半屏

            dat=0;
            wr_data2();
        }
    }
}
```

(5) 显示 16×8 字符的程序

```
void lcd_write_char_code(uchar page8,uchar column8,uchar* block168) reentrant {
    uchar column1;
    set_position(page8,column8); //设置所写字符起始页位置
    for(column1=0;column1<8;column1++) {
        dat=block168[column1];
            //从字库中取出上半页 8×8 点阵字模数据
        dat=UpsetChar[dat];
            //将每个字节数据高低位进行对调

        if(column8<=7)
            //如果设置的所写位置在左半屏, 调用写左半屏
            //数据的函数
            wr_data1();
        else
```

```
        //如果设置的所写位置在右半屏，调用写右半屏
        //数据的函数
        wr_data2();
    }
    page8++;
    set_position(page8, column8); //设置所写字符下半页位置
    for(column1=8; column1<16; column1++) {
        dat=block168[column1];
        //从字库中取出下半页 8×8 点阵字模数据
        dat=UpsetChar[dat];
        //将每个字节数据高低位进行对调
        if(column8<=7)
            //如果设置的所写位置在左半屏，调用写左半屏
            //数据的函数
            wr_data1();
        else
            //如果设置的所写位置在右半屏，调用写右半屏数据的函数
            wr_data2();
    }
}
```

5 结论

OLED 显示技术有着广泛的应用前景。采用此技术的 VGS12864E 显示模块具有与 MCU 接口方便、显示功能强和编程简单等优点，具有广泛的应用价值。上面介绍的接口设计与软件编程已成功应用于数据采集系统中的系统工作参数的显示，效果良好；再配合按键控制进行反显、参数设置、翻页等，建立了良好的人机交互界面。