

## 中文点阵液晶显示驱动器 RA8816 及其应用

北京中显电子有限公司诚意奉献

内置中文字库的点阵液晶显示驱动器 RA8816 具有极强的通用性。RA8816 将多种实用的外围电路集成在芯片内部，使得应用系统的硬件电路和软件设计大大简化。利用 RA8816 构成的人机交互系统，不仅更容易控制，而且还能降低系统成本，是构成微机控制系统人机交互的理想方案。本文给出了 RA8816 详细的硬件配置方法以及关键的控制程序。最后结合一个具体的屏幕滚动实例，给出了 RA8816 控制方法。实验结果表明，RA8816 构成的人机交互系统可以使硬件电路更加紧凑、可靠，软件设计更加方便。

### 引言

液晶显示器由于它具有功耗低、体积小、重量轻、显示的信息量大，能显示各种类型的信息，如字符、图形(包括汉字、曲线、表格等)，正越来越广泛地应用于各个领域，尤其是在便携设备中，更是占据着绝对主流的地位。

对于简单的信息显示，通常用段码式 LCD 驱动器，如盛群的 HT1621 等。而对于复杂的信息显示，则必须由点阵式 LCD 驱动器来驱动，最常见的点阵式 LCD 驱动器中，有不带中文字库的东芝 LCD 驱动器 T6963[1-4]，也有带中文字库的矽创 LCD 驱动器 ST7920[5-7]。对于较大的应用系统，由于系统通常有足够大的存储空间，可以将中文字库放在系统的存储空间中，因此常选用不带中文字库的 LCD 模块；对于较小的应用系统，系统控制器本身的存储空间有限，且运行速度较低，因此通常选用带中文字库的 LCD 模块。

在一般应用中，人机交互是通过显示屏和键盘来完成的。但在小型便携式应用中，不仅系统控制器的存储空间较小，无法将中文字库放在其存储空间内，而且控制器的 I/O 口数量也非常有限，不能满足实际应用对按键及 I/O 口数量的要求。瑞佑科技最新推出的 RA8816，不仅内置了中文字库，而且还提供了 4×5 的键盘扫描接口及多种控制器接口，特别适用于各种小型屏幕应用产品，如收银机、公用及家用电话、传真机、手持式电子装置及各式测量设备等。

## 1 RA8816 介绍

RA8816 是一个点阵 LCD 驱动控制器，能够支持中英文及图形显示模式，其内部结构框图如图 1 所示。RA8816 内部主要是由显示内存、256K 字节的字模 ROM、命令寄存器、LCD 控制器、LCD 驱动器、升压电路、分压电路、微控制器接口及键盘扫描电路等组成。其中内置 256K 字节的 ROM 字库，包含中文字库、数字和英日欧文等字母字库。1170 字节的显示内存，最大可支持 144×65 的 LCD 面板。此外，RA8816 还提供 450 字节的屏幕滚动缓存，使其具有滚动屏幕时显示内容不断更新的效果。在文字模式中，RA8816 可接收标准中文编码 (GB 码或 BIG5)，直接显示中文，而不需要进入图形模式以绘图的方式显示中文，提高了液晶显示效率。与 ST7920 不同的是，RA8816 可以任意指定中文字符的显示位置，并且无须偶地址对齐。RA8816 还提供了 256 字节的 CGRAM 用于自行造字，以便在文字模式下显示字库中没有的文字符号。

在控制器接口方面，RA8816 提供了多种接口方式，可直接支持

8080 / 6800 系列控制器的并行数据总线接口，并提供 4 位或 8 位的数据位宽。此外 RA8816 还支持串行总线控制接口，包括 3 线、4 线 (A—Type 和 B—Type 两种) 及 I2C 共四种的串行接口。

RA8816 整合了多项的实用接口，包含内置 DC / DC 升压电路及液晶亮度调整电路，4 x 5 的键盘扫描接口并支持长按及短按时间判断，8 个通用 IO 口可直接驱动 LED，这些接口极大地简化了外围电路设计。因此在系统开发时，可选用成本较低微控制器与 RA8816 搭配，不仅使系统设计快捷、方便，而且能够降低开发成本。

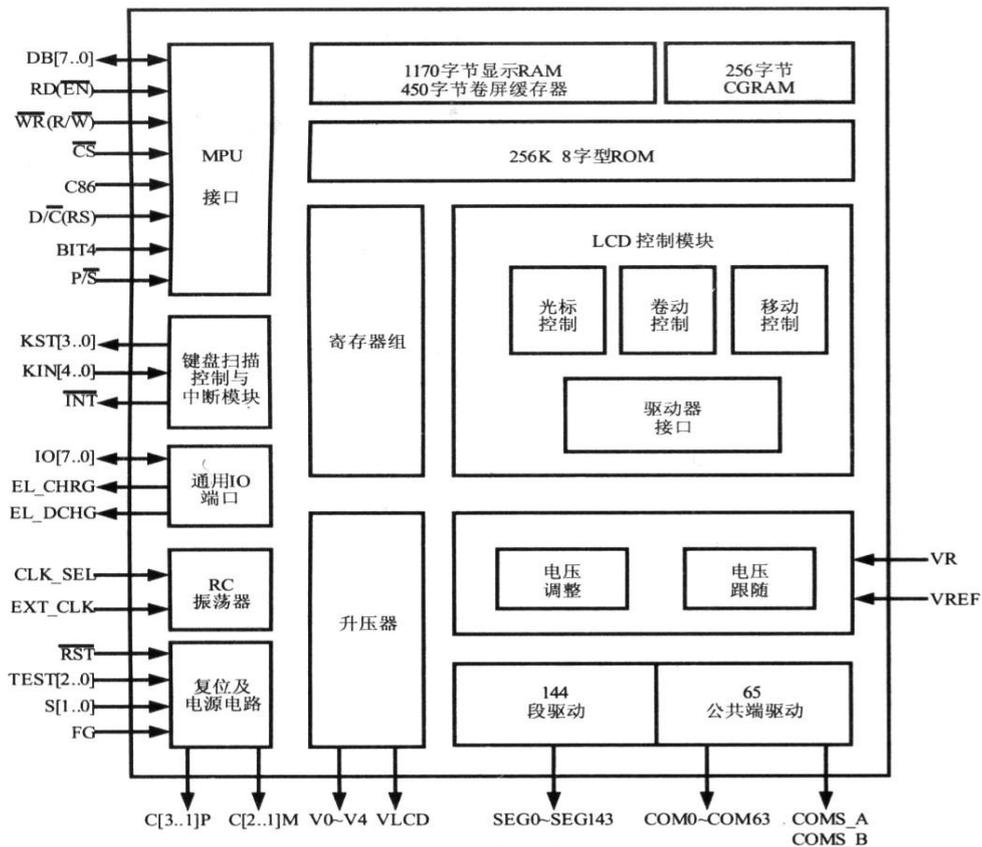


图1. RA8816内部结构框图

RA8816 的主要特性有：

1. 支持 1 / 65 占空比，1 / 9~1 / 5 偏压的 LCD 面板；
2. 内置 2~4 倍升压电路、电压调整电路及电压跟随电路；

3. 内置 RC 振荡器；
4. 支持文字与图形两种混和显示模式；
5. 支持 BIG5 或 GB0 码
6. 支持 ASCII 8×8 英文字型，8×16 半角和 16×16 全角简体中文或繁体中文；
7. 支持粗体、反白、滚动等多种显示效果。

## 2 RA8816 硬件配置

RA8816 可工作在并行模式或串行模式，工作模式的选择由 P/S 引脚状态来决定，P/S 为高电平时，选择并行模式，否则为串行模式。引脚 C86 用来选择并行接口的类型，当 C86 为高电平时选择 6800 系列控制器接口，为低电平时选择 8080 系列控制器接口。引脚 BIT4 用来确定并行模式中数据总线的宽度，BIT4 为高电平时总线宽度为 4 位，否则总线宽度为 8 位。当工作在串行控制模式时，其串行接口方式由表 1 确定。

表1 串行接口方式

SMOD (DB[7..6])	串行接口方式	脚位分配
00	I <sup>2</sup> C 方式(2-Wire)	SCK(DB0); SDA(DB1)
01	3 线方式(3-Wire)	SCK(DB0); SDA(DB1); $\overline{CS}$ (DB3)
10	4 线方式(A-Type)	SCK(DB0); SDA(DB1); RS(DB2); $\overline{CS}$ (DB3)
11	4 线方式(B-Type)	SCK(DB0); SDO(DB1); SDI(DB2); $\overline{CS}$ (DB3)

LCD 驱动方式及面板材质的不同，所需要的驱动电压是不同的；在实际应用系统中，系统能提供的电压也是有差异的。这要求 LCD 驱动芯片有较宽的电压工作范围，RA8816 内置有步进升压电路，可以产生 4 倍、3 倍或 2 倍于 (VDD—VSS) 电压的 LCD 驱动电压—VLCD，该

电压供给电压调整器及内部 LCD 驱动电路使用。若在 C1P 与 C1M 跨接一个  $1\mu\text{F}$  左右的电容则能产生 2 倍于  $(V_{DD}-V_{SS})$  电压；若 C2P 与 C2M 也跨接一个  $1\mu\text{F}$  左右的电容则产生 3 倍于  $(V_{DD}-V_{SS})$  电压；若要产生 4 倍于  $(V_{DD}-V_{SS})$  电压，则还要在 C3P 与 C1M 间跨接一个  $1\mu\text{F}$  左右的电容，如图 2 所示。RA8816 灵活的 VLCD 产生方式，使得其能够驱动不同材质的 LCD 面板。

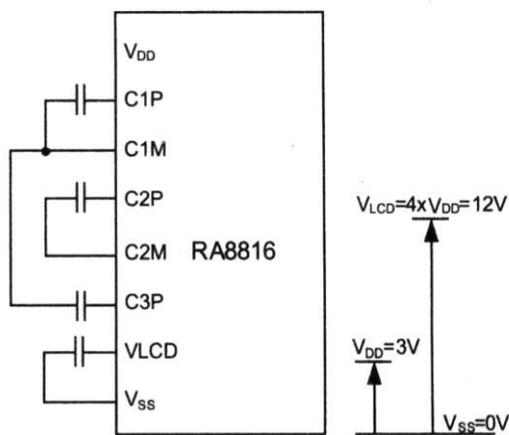


图2. LCD驱动电压升压配置

RA8816 的系统时钟由引脚 CLK\_SEL 的状态来确定，CLK\_SEL 为高电平时，系统时钟由内部 RC 振荡电路产生；CLK\_SEL 为低电平时，系统时钟需外电路提供并从 EXT\_CLK 引脚输入。RA8816 的矩阵键盘扫描接口应用起来非常方便，只要将  $KsT[3..0]$  行扫描输出和  $KIN[4..0]$  列输入信号接上相应的按键，如图 3 所示，并通过相应命令寄存器的设置就能正常工作。当有按键按下时，引脚会产生中断信号，系统控制器就可以通过控制总线接口读入相应的键码。

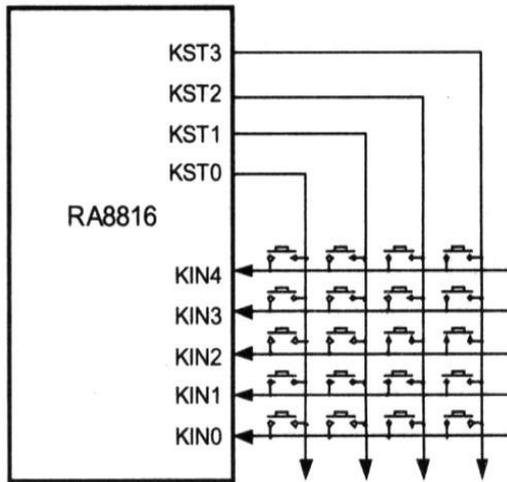


图3. 键盘扫描电路

### 3 RA8816 软件编程

系统控制器对 RA8816 有两种控制时序，一种是对命令寄存器的读写 ( $R_s=0$ )，另一种是对数据寄存器的读写 ( $R_s=1$ )。RA8816 的命令寄存器共有 25 个，在对命令寄存器读写时，控制器必须先告诉 RA8816 要对哪一个命令寄存器进行读或写，因此控制器传递给 RA8816 的第一个数据是 RA8816 命令寄存器的索引号，第二个数据才是真正写入 (或读出) 该命令寄存器的有效数据。命令寄存器控制着 RA8816 的显示属性，数据寄存器反映 RA8816 的显示内容。RA8816 的控制接口有多种形式，下面以 I2C 接口为例，说明如何对 RA8816 命令寄存器进行读写，其他接口的读写过程与此相似。RA8816 命令寄存器的写时序如图 4 所示。

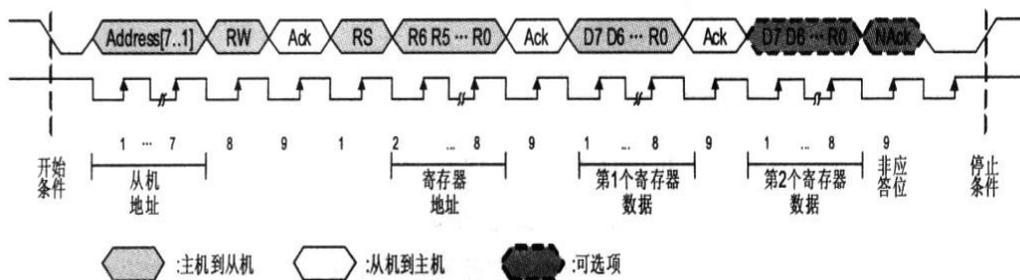


图4. I<sup>2</sup>C接口对命令寄存器写时序

根据图 4 所示时序，写命令寄存器的子程序如下所示：

```
uchar LCD_CmdWrite(uchar Reg_Addr, uchar
Data) {
    uchar check_flag;
    I2C_Start(); //I2C开始条件
    I2C_SendAddress(RA8816_Addr, I2C_WRITE);
//RA8816器件地址(写)
    check_flag = I2C_ReadAck(); //读取应答位
    if (check_flag) goto lable_i2c_err;
//RA8816没有应答
    I2C_SendData (Reg_Addr|RS_0); //发送RA8816命
令寄存器索引号
    check_flag = I2C_ReadAck();
    if (check_flag) goto lable_i2c_err;
    I2C_SendData(Data); //发送有效数据
    check_flag = I2C_ReadAck();
    if (check_flag) goto lable_i2c_err;
    return I2C_OK;
lable_i2c_err:
    I2C_Stop(); //I2C停止条件
    return I2C_ERROR;
}
```

当 RS=I，表示控制器对 RA8816 进行数据寄存器的读写，如果写入 (RW=0) 数据到数据寄存器，写入的内容根据命令寄存器设置而显示在 LCD 屏上。如果是读取 (RW=1) 操作，则读取的是不同模式下的字模数据。

有了对寄存器的读写函数后，就可以对 RA8816 的显示进行控制，下面以文字滚动功能为例，说明 RA8816 的控制方法。RA8816 提供 LCD 的显示画面，可根据命令寄存器任意指定一块区域，由硬件自动做屏幕画面的滚动，其中命令寄存器可以设置屏幕的滚动方向和速度，每次滚动画面的像素与滚动的范围等。与滚动相关的主要命令寄存器有：SWSXR[0x08]和 SWSYR[0x09]两个寄存器决定滚动区域的左上角坐标 x 和 Y；SWR\_xR[0x0A]和 SWRYR[0x0B]两个寄存器决定滚动区域的大

小  $A_x$  和  $\Delta Y$ ; SCOR[0x0C]表示滚动的位移量; ASCR[0x0D]的 DB7' DB4 决定滚动的速度, DB3~DB0 决定每次位移的像素点数; SCCR[0x0E] 的 DB7 和 DB6 决定滚动多少个像素点后产生中断, DB3 和 DB2 决定屏幕滚动的方向。图 5 给出了屏幕滚动流程图。

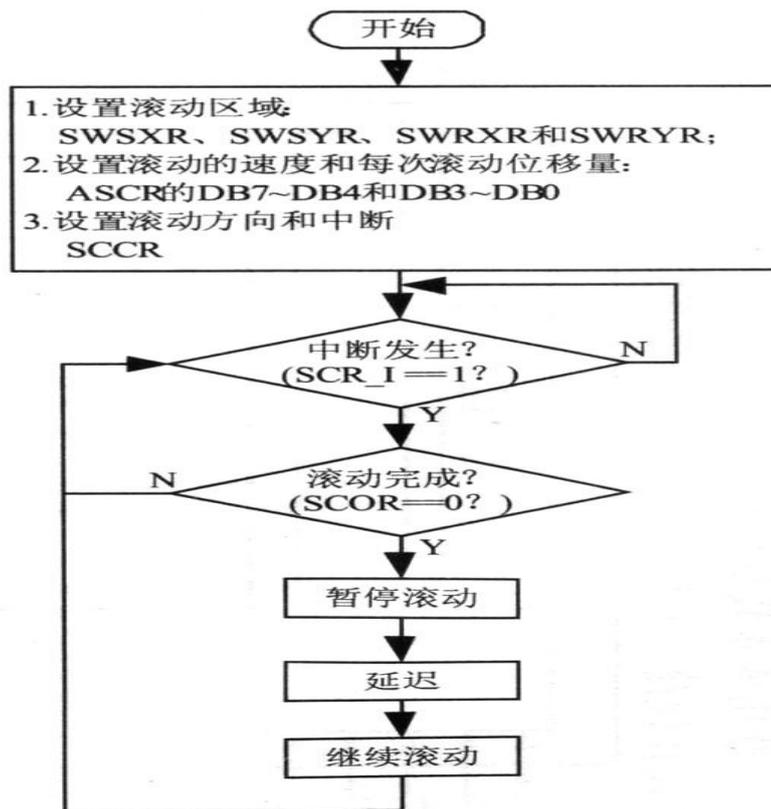


图5. 屏幕滚动流程图

图 5. 屏幕滚动流程图

屏幕滚动的主要代码如下:

```

void Roll_LCD(uchar R_times) {
LCD_CmdWrite(MWMR, 0x43); //正常文字显示模式
GotoXY(0, 0); //设定光标地址
Disp_Str("液晶显示滚动测试文字由上往下滚动");//显示中文字符串
LCD_CmdWrite(SWSXR, 2); //设定滚动范围水平起始地址
LCD_CmdWrite(SWSYR, 16); //设定滚动范围垂直起始地址
LCD_CmdWrite(SWRXR, 10); //设定滚动范围水平范围
LCD_CmdWrite(SWRYR, 16); //设定滚动范围垂直范围
LCD_CmdWrite(ASCR, 0x30); //滚动速度设定
  
```

```
    LCD_CmdWrite(SCCR, 0xAB); //由上往下滚动, 每移动8个像素  
点中断一次  
    while(scr_times--){  
        while((LCD_CmdRead(ISR) & 0x04) == 0x04) { //中断产生  
            if(LCD_CmdRead_SPI3(SCOR) == 0x00) { //判断是否已经滚动  
完成  
                LCD_CmdWrite(SCCR, LCD_CmdRead(SCCR)&0xFE); //暂停滚动  
                delay_5ms(100); //延迟  
                LCD_CmdWrite(SCCR, LCD_CmdRead(SCCR)|0x01); //继续滚动  
            }  
        }  
    }  
}
```

#### 4 小结

内置中文字库的液晶显示驱动器 RA8816, 不仅具有多种显示模式及显示效果, 而且支持常见的 8080 / 6800 系列控制器并行接口和多种串行接口, 这极大地扩展了 RA8816 的应用范围。RA8816 内置的步进升压电路及电压调整电路, 可以匹配较宽范围的系统电压。此外, RA8816 还提供了键盘扫描和通用 IO 接口, 大大简化了系统硬件电路设计和软件设计, 使得用低成本的控制器的就能实现完善的人机交互界面。

北京中显电子有限公司祝你研发愉快!