

单片机系统液晶菜单设计

Design of LCD Display Menus in MCU System

文 浩 (武汉理工大学自动化学院,湖北 武汉 430070)

摘 要

以 MGLS-19264 液晶显示器为例,介绍了在 LCD 液晶上实现多级菜单和多层屏幕的建立。提出了一种利用数据结构,设计 LCD 液晶菜单通用方法。该方法通用性较强。

关键词:液晶显示,菜单,单片机,数据结构

Abstract

As an example of LCD module MGLS-19264, the paper introduces how to establish multilevel menus and multilayer screens. According to data construction, the method used to design LCD menus is presented. The design thoughts of software is effective and universal.

Keywords: LCD, menu, single-chip microprocessor, data construction

在单片机系统设计中, LCD 人机交互界面的设计往往占据着很大一部分工作。本文介绍一种用 C 语言实现的 LCD 菜单的软件设计方法,具有很好的可移植性,可大大提高开发效率。本文使用北京精电蓬远公司 MGLS-19264 液晶模块为例介绍这种固定的 LCD 菜单编程模式。

1 液晶显示模块 MGLS-19264 简介及其基本驱动函数

1.1 MGLS-19264 简介

MGLS-19264 点阵式图形液晶模块点阵数为 192 列 \times 64 行,由 1 块行驱动器 HD61203 和 3 块列驱动器 HD61202 及外围电路组成,HD61202 片内有显示 RAM,用户无需外接显示 RAM。各引脚功能如下:

1、2- CSA、CSB 两片选端,供 CPU 接口选用。CSA、CSB 为 00 时,选通 HD61202(1),为 01 时,选通 HD61202(2),为 10 时,选通 HD61202(3); 3- GND 系统接地; 4- VCC+5V 电源; 5- V0 液晶驱动控制电压,用于 LCD 对比度调整; 6- D/I 数据/指令选择, D/I=1, 数据操作, D/I=0, 写指令或读状态; 7- R/W 读写选择信号, R/W=1, 读选通, R/W=0, 写选通; 8- R 读写使能信号,在下降沿,数据被锁存(写)入 HD61202,在高电平期间,数据被读出; 9- 16 为 DB0-DB7 数据总线。

1.2 基本驱动函数编写

首先使用 C 语言在头文件中对 MGLS-19264 控制线进行定义,然后编写 LCD 驱动函数,这些函数包括模块初始化、写指令代码、写显示数据、读显示数据、写菜单项、清屏、清页等函数。

2 LCD 显示菜单设计

2.1 菜单界面设计

本系统应用于无功补偿控制器,要求显示电网三相电压、电流值,无功功率、有功功率、无功因数,20 次的电压谐波含量,无功补偿控制方式自动/手动选择,参数设置等。针对显示要求把整个系统设计成多级屏幕(窗口),又设计成多级主菜单,类似的功能放在一个主菜单菜单项下构成子菜单目录。如果菜单目录项过多,多屏显示此菜单目录。通过分级,使操作人员能很直观快速找到自己当前需要的菜单项,液晶屏菜单界面设置如图 1 所示。

2.2 建表

字符或数字在点阵液晶中的显示是通过写相应的内码(1 点亮,0 灭)实现的,于是先做好码表,把用到的汉字对应的内码保存在一个名为 HZ_TAB 的表中,每个 16×16 点阵的字符需要

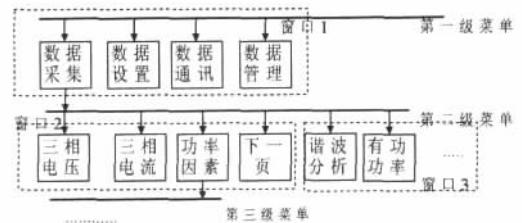


图 1 电网无功补偿控制器的分级菜单

32 个 byte 表示; 把要用到的数字 0~9 和字母对应的 16×8 点阵内码保存在一个名 NUM_TAB 的表中,每个需要 16 个 byte 表示; 把图片保存在 PIC_TAB 表中,需要 byte 数根据大小而定。 16×16 点阵汉字和 16×8 点阵数字、字母和任意大小黑白图片字模数据库可由 pyhzbj.EXE 软件提取字模建立。

2.3 数据结构

本系统采用 6 个按钮来接受用户输入,6 个按钮为上、下、左、右、回车、退回。为方便数据的显示和处理,需要首先建立一个菜单窗口结构体,并定义一个结构变量 WinStruct,该结构中共有 9 个结构元素,分别是 8 个字符型和 1 个指针变量,指针变量指向需执行函数。

```

typedef struct{
    uchar MenuIndex; //当前菜单索引号
    uchar BackWinIndex; //当前窗口的上一窗口索引号
    uchar NextDnIndex; //下一个菜单索引号,按下“上,下”键时当前菜单项上下移动
    uchar NextUpIndex; //或可修改参数数值向上下循环变化
    uchar NextLIndex; //下一个菜单索引号,按下“左,右”键时当前菜单项左右移动
    uchar NextRtIndex; //或在可修改参数位置上左右移动
    uchar KeyCrState; //按下“回车”键时用户选择确认或参数输入确定
    uchar KeyBackState; //按下“退回”键时用户退出当前窗口,返回上一窗口
    void (*CurrentOperate)(); //当前状态应该执行的功能操作
} WinStruct;
  
```

菜单项在 LCD 上显示,通过建立菜单项结构体,定义一个结构变量 MenuStruct,如下:

```

typedef struct{
    uchar MenuIndex; //当前菜单索引号
    uchar NextWinIndex; //下一窗口索引号
    uchar cursor_page; //菜单项起始页位置 (下转第 77 页)
  
```

2 由波特率误差引起的错码现象分析

如果 PC 机与单片机的通信速率约定为 9600b/s, 系统时钟为 6MHz 时, 为了按约定的速率通信, PC 机在 8250 异步通信接口中的 3F8H 和 3F9H 寄存器中设定波特率因子分别为 0CH 和 00H; 而单片机中定时器 1 初值 TH1=TL1=FDH。此时, PC 机发送数据与单片机接收数据的情况如下:

PC 机发送数据	单片机接收数据
10H ~1FH	30H ~3FH
20H ~2FH	40H ~4FH
30H ~3FH	70H ~7FH
40H ~4FH	80H ~8FH
50H ~5FH	130H ~BFH
...	...

以上数据表明了接收数据中出现的错码情况。通过对上述错码进行剖析可以看到: 当因波特率误差引起接收端采样偏移时, 如果这个偏移使得接收数据位的采样在该位中点的半位间隔时, 将会对该位采样两次, 所以形成了上面的错码情况。下面公式表明错码或漏码发生的位数 N:

波特率相对误差 \times 第 N 位 > 0.5 , 即 $N > 0.5 / \text{波特率相对误差}$ 。

当串行异步通信的帧格式为 11 位时:

N = 11, 表示一帧数据中有某位被采样错, 且采样出错在第 N 位;

N > 11, 表示一帧数据中没有数据位发生采样错。

本例中, 波特率误差为 8.5% 由上面公式得出 $8.5\% \times 11 = 0.935 > 0.5$, 说明在这个波特率误差下将出现采样错, 且出错位为: $N = 0.5 / 8.5\% = 5.9 \approx 6$ (含起始位)

即在数据帧包括起始位的第 6 位发生错码, 在第 6 位采样了两次, 因而得到上面错码情况。

0 0 0 1	0 0 0 0	0 起始位
↙		
错位: 0 0 1 1	0 0 0 0	↙ 则 10H 30H
0 1 0 1	0 0 0 0	0
↙		
1 0 1 1	0 0 0 0	则 50H B0H

由此可以类推出错码的接收数据。

相反, 当 PC 机的传输波特率大于单片机的传输波特率时, 则会在第 N 位出现漏码错误。这里不再赘述。

由上面的公式还可以推断出, 当波特率相对误差小于 4.5% 时, 这种采样的偏移将不影响正常数据的接收。但为了保证传的可靠性, 通常波特率的误差不大于 2.5%, 而尽量选择最小的传输误差率。

3 SMOD 位对波特率的影响

在波特率的设置中, 还有一个问题值得注意, 那就是 SMOD 位的选择也影响着波特率的准确率, 如下例:

设波特率为 2400b/s, $f_{osc} = 6\text{MHz}$ 时, SMOD 可以任意选择为 0 或者 1。但是由于对 SMOD 不同的选择, 则产生了不同的波特率误差:

$$2400 = 2^{\text{SMOD}} \times 6 \times 10^6 / 32 \times 12 (256 - x)$$

当 SMOD=0 时, $x = 249 = \text{F9H}$;

当 SMOD=1 时, $x = 243 = \text{F3H}$ 。

实际的波特率误差及误差如下:

SMOD=0 时, 波特率 2232.14b/s, 误差=7%

SMOD=1 时, 波特率 2403.85b/s, 误差=0.16%

上面的分析说明了 SMOD 的值虽然是可以任意选择的, 但是在某些情况下, 它直接影响着波特率误差范围, 因而在波特率设置时, 对 SMOD 的选取也要做适当的考虑。

4 结束语

以上对串行通信所必须涉及的波特率设置问题作了一些分析。在异种机的串行通信中, 波特率的设置是非常重要的问题, 它直接关系到串行通信的成功与失败, 因此在串行通信中要对波特率的设置给予足够的认识。

参考文献

- [1] 胡汉才. 单片机原理及其接口技术[M]. 北京: 清华大学出版社, 1996
- [2] 何立民. 单片机应用系统设计[M]. 北京: 北京航空航天大学出版社, 1999
- [3] 求实科技. 单片机通信技术与工程实践[M]. 北京: 人民邮电出版社, 2005

[收稿日期: 2006.9.27]

(上接第 75 页)

```
uchar cursor_col ; //菜单项起始行位置
uchar type ; //菜单项显示的数据类型, 16 x16 或
16 x8 点阵等
uchar * data; //指向显示数据区
uchar dataNum; //需要显示的数据个数
uchar black; //是否反显菜单项
uchar modify ; //数据是否需要修改
uchar which_bit ; //需要修改的数据是第几位数据
} MenuStruct;
```

2.4 软件设计

系统上电, 初始化点阵液晶显示模块及其他后进入主窗口显示, 当前菜单项反色显示(反色是在一块区域中, 线条上是空白, 该区域中其余部位都被填充黑色)。反色显示是通过对所显示内容的字符串数据取反, 然后传给液晶显示。

在液晶显示程序当中, “上、下、左、右” 按键中断后, 下一个菜单索引号变为当前状态索引号, 对应的结构元素(是否反显)发生改变, 刷新 LCD 液晶显示屏程序执行, 使当前菜单项反色显示。“回车” 按键后执行当前菜单索引号所对应的执行功能操作, 通过建立菜单索引号的条件分支语句可实现。如菜单索引号对应数据远程通信程序, 则控制器执行远程通信传输程序; 如对

应是进入下一子菜单目录程序, 则刷新屏幕, 显示下一窗口索引号对应的窗口, 当前菜单索引号也变为子菜单目录的首菜单项索引号。“退回” 按键后则刷新屏幕, 显示结构元素(BackWinIndex)对应的上一窗口。

窗口显示通过不同窗口索引号包含不同的菜单项实现。建立二维数组 Win[WinIndex][], 以窗口索引号寻找到对应的包含菜单项, 然后刷新屏幕显示对应的菜单项。建立图片数据库, 增加菜单项显示的点阵数据类型, 就可图片菜单窗口显示。

3 结束语

本文介绍的方法可以很方便修改菜单和增加子菜单, 在此菜单编程方法基础上进行改进, 还可增加滚屏, 自动翻页等功能。此方法也同样应用于 ARM7 与 DSP 其他微控制器 LCD 液晶人机界面上。

参考文献

- [1] 北京精电蓬远显示技术有限公司. 内藏 HD61202 控制器图形液晶显示模块的应用[M]. 北京: 北京清华蓬远科贸公司, 2000
- [2] 林志坚. 液晶显示模块在流量计中的实现[J]. 电子器件, 2006(6)

[收稿日期: 2006.9.28]