

基于DSP的摊铺机人机接口设计

Design of Paver Onboard HMI Based on DSP

翁寅生¹, 焦生杰², 王海英²

WENG Yin-sheng¹, JIAO Sheng-jie², WANG Hai-ying²

1.中国科学院 西安光学精密机械研究所, 陕西 西安 710119

2.长安大学 道路施工技术与装备教育部重点实验室, 陕西 西安 710064

1. Xi'an Institute of Optics and Precision Mechanics of Chinese Academy of Sciences, Xi'an 710119, Shaanxi, China

2. Key Laboratory for Highway Construction Technology and Equipment of Ministry of Education, Chang'an University, Xi'an 710064, Shaanxi, China

【摘要】 针对摊铺机人机界面需要显示大量图文信息的问题, 提出了一种基于DSP TMS320LF2407A的按键输入和显示输出的软硬件实现方案。设计了DSP与液晶显示控制器SED1330及EEPROM 24LC256的硬件接口电路, 重点介绍了利用DSP的数字I/O口来模拟SED1330时序的方法, 给出了关键的软件程序, 在实际系统应用中取得了成功。

【Abstract】 According to plentiful information of graphics and characters needed by the onboard HMI of paver, a realization means of software and hardware is brought forward based on key-press as input and LCD as output controlled by DSP TMS320LF2407A. The hardware interface circuit among DSP, LCD controller SED1330 and EEPROM 24LC256 is designed, and using digital I/O ports to simulate time sequence of SED1330 is introduced in detail, and key program is given. Finally, it is accomplished in the practical system application.

【关键词】 摊铺机; 液晶; 数字信号处理器; SED1330

【Key words】 Paver; LCD; DSP; SED1330

中图分类号: U415.52

文献标识码: B

文章编号: 1000-033X(2008)01-0047-03

0 引言

摊铺机人机接口中要求显示欢迎画面、可以实时更改并存储机器的设定参数的主界面、故障代码及其排除方法的画面, 甚至需要显示控制系统与外界的通信情况, 所以摊铺机人机接口需要显示的图片 and 文字比较多。大量图片和文字所产生的点阵数据会占用大量的程序存储空间, 程序过长时点阵数据易溢出, 从而导致溢出部分的图形点阵无法显示, 所以常采取的将点阵数据存放于程序区中的方法并不可取。如何保存大量图片和文字的点阵, 并将其分屏显示出来是本文所要解决的重点与难点。为此, 下面以摊铺机人机界面中的状态主界面为例, 选用320×240点阵式LCD模块CA320240B作为显示器, 电可擦写可编程只读存储器(EEPROM)24LC256作为图片点阵数据的预存储器以及参数设定的存储器, 2个按键用于控制上下翻页; 液晶显示控制器SED1330采用图形方式显示, 利用Windows自带的画图工具画一个不超过320×240像素点的位图图片, 然后利用字模提取软件取出图片的点阵数据, 通过I²C总线可以将点阵数

据分批写入EEPROM, 待显示时再将欲显示的图片点阵读出来进行显示^[1-2]。

1 硬件设计

图1所示为液晶显示控制器SED1330、电可擦写可编程只读存储器(EEPROM)与DSP TMS320LF2407A的硬件接口电路。DSP对SED1330接口控制板采用间接访问的方式, 即用DSP的I/O口来控制SED1330。DSP的工作电压是+3.3 V, 而SED1330的驱动电压为+4.5~+5.5 V, 因此在设计DSP与液晶的

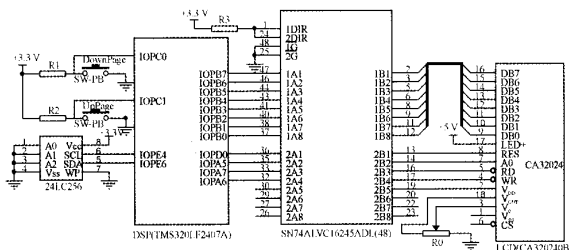


图1 SED1330、EEPROM与DSP的硬件接口电路

接口时,必须进行电平转换,这里使用专门的电平转换器件SN74LVC16245A。该芯片采用+3.3 V和+5 V电压双供电,能很好地解决电平转换问题。SN74LVC16245A的1G与2G接地,使其一直工作;1DIR与2DIR置高,数据从DSP输出到液晶。DSP的IOPB0-IOPB7与液晶显示模块的数据线DB0~DB7相连,完成与SED1330间的数据传送;IOPA5与A0相连,决定是数据(A0=0)还是命令(A0=1);IOPA6与WR相连,写SED1330时置低;IOPA7与RD相连,读SED1330时置低;IOPD0与RES相连,用于复位。CS接地,始终使能SED1330。背光电压由逆变器提供,LED+接+5 V以提供背光,调节电位器R0即可调节液晶显示屏的对比度。

EEPROM 24LC256的工作电压为+2.5~+5.5 V,可以与DSP直接相连。Vcc与Vss是电源正、负极;WP是硬件写保护管脚,接Vss可以对整个存储空间进行读写操作,否则只能读不能写;A2、A1、A0用于多片24LC256同时使用中,接相应的高低电平作为片选;DSP的IOPE4与SCL相连,作为串行时钟输入,用于控制数据的输入与输出;IOPE6接SDA,用于控制输入或输出串行数据。由于DSP内部对I/O口进行了上拉,所以外部可以不再接上拉电阻。这里A2、A1、A0均接地,故EEPROM的写控制字为0xA0,读控制字为0xA1。

为简单起见,这里只用2个按键来说明键盘的输入,分别用IOPC1和IOPC0控制液晶屏显示信息的上下翻页—每按下IOPC1所接的键UpPage则向上翻一页,每按下IOPC0所接的键DownPage则向下翻一页。

2 软件设计

系统硬件设计部分比较简单,重点应是软件设计。软件设计主要包括SED1330模块(初始化、读写时序、清屏及显示程序)、EEPROM模块(读写程序)、按键扫描模块。其中核心程序是SED1330模块中的通用显示程序及EEPROM模块的读写程序,所以下面重点介绍这些模块。

2.1 SED1330的初始化、读写时序及通用显示程序

由于本设计中液晶显示用DSP的数字I/O口来模拟SED1330的接口时序,液晶显示模块接口控制时序采用8080时序,M8080系列接口时序如图2所示。SED1330初始化要注意设置4个显示区的起始地址分别为0、0x2580、0x4B00、0x7080,每行需要的字节数为40,点行数为240,LCD

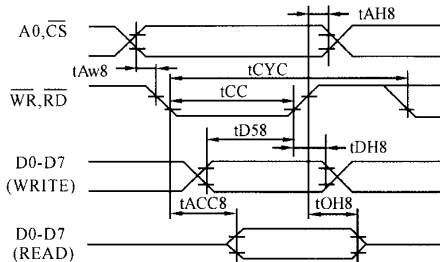


图2 SED1330-M0808系列接口时序

工作频率为70 Hz,显示内存分配给一个显示行的单元数为40字节。1显示区与3显示区采用文本方式,2显示区与4显示区采用图形方式,二重或合成显示。显示地址LcdAddr用行列交叉计算法得到,分别用csrh和csrl来控制行值和列值,对应显存地址为csrl+csrh×N,N为每行的字节数,根据指令参数并结合图2编写以下各主要子程序。其中最主要的程序是DisplayEeprom(),它实现了从EEPROM中读出点阵数据并显示出来的功能。

```
void CmdWrite(unsigned char lcdcmd)
{ *PADATDIR=*PADATDIR|0x2020;
  *PADATDIR>(*PADATDIR|0x4000)&0x0FFBF;
  *PBDATDIR>(*PBDATDIR&0x0FF00)|lcdcmd;Delay(100);
  *PADATDIR=*PADATDIR|0x4040; }
void DataWrite(unsigned char lcddata)
{ *PADATDIR=*PADATDIR|0x2000)&0x0FFDF;
  *PADATDIR>(*PADATDIR|0x4000)&0x0FFBF;
  *PBDATDIR>(*PBDATDIR&0x0FF00)|lcddata;Delay(100);
  *PADATDIR=*PADATDIR|0x4040; }
```

通用字符汉字及图形显示程序实现从EEPROM处读出点阵并显示出来的功能。EepromBlockAddr是点阵存储的起始地址,LcdAddr是要显示区域的左上角地址,x,y是显示区域的宽与高,0≤x≤40,0≤y≤240,横向取模。不过需要先调用WriteAllPages()将图片点阵写入EEPROM,才能调用此函数读点阵显示,用字符汉字及图形显示程序如下。

```
void DisplayEeprom(unsigned int EepromBlockAddr, unsigned int LcdAddr, unsigned int x, unsigned int y)
{ unsigned int i, j, addr, p, csrh, csrl;
  for(i=0; i<x; i++)
  {p=EepromBlockAddr+i;
   addr=LcdAddr+i; csrl=addr%0x100; csrh=addr/0x100;
   CmdWrite(0x46); DataWrite(csrl); DataWrite(csrh);
   CmdWrite(0x4F); CmdWrite(0x42);
   for(j=0; j<y; j++){ DataWrite(ReadOneData(p));
   p=p+x; }}}
```

2.2 EEPROM读写程序

显示图片以及系统的参数主要就是从EEPROM中先存入点阵再读出、显示,所以这部分程序也非常重要。由于DSP没有专门的PC总线,故只能用I/O口来模拟PC总线时序,所以程序开始部分可以利用宏定义,再结合时序图编写程序。下面给出本文中要用到的主要程序。

ReadOneData()函数从EEPROM地址DataAddr16中读出1个字节数据,并返回读出的数据。显示时就是调用这个函数将点阵数据逐个读出来。

```
unsigned char ReadOneData(unsigned int DataAddr16)
{unsigned int k, ReadData, DataAddr8H, DataAddr8L;
  DataAddr8H=DataAddr16/0x100;
```

```
DataAddr8L=DataAddr16%0x100;
do { Start;Write8Bit (0x0A0);k =TestAck ();}while (k ==1);
do { Write8Bit(DataAddr8H);k=TestAck();}while(k==1);
do { Write8Bit(DataAddr8L);k=TestAck();}while(k==1);
do { Start;Write8Bit(0x0A1);k=TestAck();}while(k==1);
ReadData=Read8Bit();
NoAck;Stop;return(ReadData);
```

WriteAllPages()函数实现写整个EEPROM的功能,利用这个函数可以将每个图片的点阵SourceArray[n1][n2]按顺序存至EEPROM的首址为RamAddr16的单元中。需要注意的是EEPROM每页包含64字节数据,当内部地址指针达到本页的最高地址后就自动返回到本页的最低地址,并将已写过的数据覆盖,所以不满1页的应该先写。

```
void WriteAllPages(WriteCmdAddr,RamAddr16,Source Array,
n1,n2)
```

```
unsigned char WriteCmdAddr,RamAddr16,*SourceArray,n1,
n2;
```

```
{ unsigned int i,n0,n,p,k;n=n1*n2;
for(i=0;i<512;i++){if(RamAddr16<=64*i)break;}
n0=64*i-RamAddr16;
```

```
if (n0!=0)WriteOnePage (WriteCmdAddr;RamAddr16,Source
Array,n0);
```

```
RamAddr16+=n0;SourceArray+=n0;n-=n0;p=n/64;
```

```
if(p==0)WriteOnePage(WriteCmdAddr,RamAddr16,Sou-
rce Array,n);
```

```
else { for(k=0;k<p;k++)
```

```
{WriteOnePage(WriteCmdAddr,RamAddr16,SourceArray,64);
```

```
RamAddr16+=64;SourceArray+=64;n-=64;}
```

```
WriteOnePage(WriteCmdAddr,RamAddr16,SourceArray,n1,
n2);}}
```

3 试验

下面在图形方式下以显示设定参数主界面的图片为例说明如下:首先,将图片点阵数据写入EEPROM,如果要显示一个最大像素为320×240的图片,计算该BMP图片总字节数为:320/8×240=9000(0x2580),假设写入EEPROM的首址为0x0000,则EEPROM存储该图片的地址为0x0000~0x257F,以后的图片要存储在0x2580以后,否则会将以前的点阵数据覆盖;用画图工具画一个像素点为320×240的BMP图片,再用字模软件取出其点阵数据(注意所编程序要求是横向取模),将上半屏点阵数据保存在数组BMP[40][120]中,并放入头文件中;再将此头文件包括进来,主程序调用WriteAllPages(0xA0,0x0000,&BMP,40,120),下半屏点阵数据保存在数组BMP[40][120]中,包括进来后调用WriteAllPages(0xA0,0x12C0,&BMP,40,120),这样就完成了将320×

240的BMP图片点阵分2次存入EEPROM中的0x0000~0x257F单元内;然后,再将EEPROM中已存储的点阵数据显示出来。要将保存在EEPROM首址为0x0000的图片显示在SED1330的两区液晶屏上,则液晶屏的左上角首址为0x2580(此地址应与LCD初始化时的Scroll参数一致),主程序调用DisplayEeprom(0x0000,0x2580,40,240)即可。需注意的是:写入EEPROM的图片点阵一定要正确,若程序太长,为防点阵数据丢失,可将其分2次或多次写入,地址应连续;液晶接口地址及显示RAM的分区和逻辑合成方式应正确,WriteAllPages(0xA0,0x0000,&BMP,40,240)与DisplayEeprom(0x0000,0x2580,40,240)参数要对应;DSP的处理速度远远快于SED1330的显示速度,故每次向SED1330送出待显数据后,DSP应产生一定的延时来等待响应。

图3所示为欲显示的主界面的图片以及相应的显示结果。实际显示结果的方框中为可以实时更改的参数,可以为文本或存储在EEPROM中的图片点阵数据。由于文本方式下的字不能改变大小,只能用8×8点阵,所以还是采用图形方式。一个24LC256可以存储32 000字节数据,而一个像素为320×240的图片占320×240/8=9 600字节,所以可以显示的屏数为(32×1 024)/9 600=3.4,即可以显示3个像素为320×240的完整图片,图片太多时可将虚拟I²C总线继续扩展。

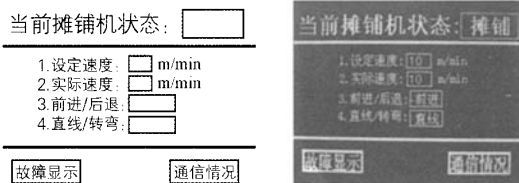


图3 欲显示的主界面图片及相应的显示结果

4 结语

本设计已调试成功,运行使用时可以稳定地显示图片及任何自定义字符,证明该系统是可行的。通过改变指令参数可实现图像闪烁、翻转、移动等复杂、灵活的显示;改变DisplayEeprom()的参数可以显示任意不超过320×240像素的位图图片。本系统的优点在于只用虚拟I²C总线就可以存入图片的点阵数据,从而降低成本,而且结构简单,系统稳定可靠,显示功能强大。软件采用C语言编写,可读性好,较容易移植到需进行复杂内容显示的系统中。

参考文献:

- [1] 翁寅生,焦生杰.LCD控制器SED1330与TMS320LF240xA的接口[J].现代显示,2006,6(5):49-52.
- [2] 刘建华,亢海伟.基于DSP的虚拟I²C总线[J].仪表技术与传感器,2005,42(5):47-48.

收稿日期:2007-06-13

[责任编辑:杜卫华]