

中文资料文库地址：

http://wenku.baidu.com/link?url=2RskpTcACLnDJp-GdMjaTVbrq7HNjjeLZ\_5Kvkz3Xld-PkmTcTwzUFbEfC\_daKr0KcE57b\_JYbeD2eX3VRI2qSAV7exIPvhsnWfnPw8ATSu

#include<reg52.h>

#include<intrins.h>

#define uchar unsigned char

sbit CS1=P2^4;

sbit CS2=P2^3;

sbit RS=P2^2;

sbit RW=P2^1;

sbit E=P2^0;

void check\_busy(); //忙检测

void write\_cmd(uchar inst); //写命令

void write\_data(uchar dat); //写数据

void set\_xy(uchar x,uchar y); //设置显示坐标

void LCD\_Init(); //lcd初始化

// 字摸软件提取出的数据，提取字模时要注意参数设置，纵向取模，字节倒序。

uchar code HZ0[6][32]={//;-- 文字: 安 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x80,0x90,0x8C,0x84,0x84,0x84,0xF5,0x86,0x84,0x84,0x84,0x84,0x94,0x8C,0x80,0x00,0x00,0x80,0x80,0x84,0x46,0x49,0x28,0x10,0x10,0x2C,0x23,0x40,0x80,0x00,0x00,0x00},

//;-- 文字: 徽 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x10,0x88,0xC4,0x23,0xAE,0x68,0x2F,0x28,0xAE,0x20,0xD8,0x17,0x10,0xF0,0x10,0x00,0x01,0x00,0xFF,0x48,0x29,0x8D,0xFB,0x09,0xAC,0x48,0x37,0x08,0x36,0x41,0x80,0x00},

//;-- 文字: 工 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x00,0x04,0x04,0x04,0x04,0x04,0x04,0xFC,0x04,0x04,0x04,0x04,0x04,0x04,0x00,0x00,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x20,0x20,0x00},

//;-- 文字: 业 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x00,0x10,0x60,0x80,0x00,0xFF,0x00,0x00,0x00,0xFF,0x00,0x00,0xC0,0x30,0x00,0x00,0x40,0x40,0x40,0x43,0x40,0x7F,0x40,0x40,0x40,0x7F,0x42,0x41,0x40,0x40,0x40,0x00},

//;-- 文字: 大 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x20,0x20,0x20,0x20,0x20,0x20,0x20,0xFF,0x20,0x20,0x20,0x20,0x20,0x20,0x20,0x00,0x80,0x80,0x40,0x20,0x10,0x0C,0x03,0x00,0x03,0x0C,0x10,0x20,0x40,0x80,0x80,0x00},

//;-- 文字: 学 --

//;-- 宋体12; 此字体下对应的点阵为：宽x高=16x16 --

{0x40,0x30,0x11,0x96,0x90,0x90,0x91,0x96,0x90,0x90,0x98,0x14,0x13,0x50,0x30,0x00,0x04,0x04,0x04,0x04,0x04,0x44,0x84,0x7E,0x06,0x05,0x04,0x04,0x04,0x04,0x04,0x00}

};

uchar code FZ0[10][32]={

/\*-- 文字: I --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x00,0x08,0x08,0xF8,0x08,0x08,0x00,0x00,0x00,0x20,0x20,0x3F,0x20,0x20,0x00,0x00},

/\*-- 文字: --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

/\*-- 文字: L --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x08,0xF8,0x08,0x00,0x00,0x00,0x00,0x00,0x20,0x3F,0x20,0x20,0x20,0x20,0x30,0x00},

/\*-- 文字: O --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,0x0F,0x10,0x20,0x20,0x20,0x10,0x0F,0x00},

/\*-- 文字: V --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x08,0x78,0x88,0x00,0x00,0xC8,0x38,0x08,0x00,0x00,0x07,0x38,0x0E,0x01,0x00,0x00},

/\*-- 文字: E --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x08,0xF8,0x88,0x88,0xE8,0x08,0x10,0x00,0x20,0x3F,0x20,0x20,0x23,0x20,0x18,0x00},

/\*-- 文字: --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00},

/\*-- 文字: Y --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x08,0x38,0xC8,0x00,0xC8,0x38,0x08,0x00,0x00,0x00,0x20,0x3F,0x20,0x00,0x00,0x00},

/\*-- 文字: O --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0xE0,0x10,0x08,0x08,0x08,0x10,0xE0,0x00,0x0F,0x10,0x20,0x20,0x20,0x10,0x0F,0x00},

/\*-- 文字: U --\*/

/\*-- 宋体12; 此字体下对应的点阵为：宽x高=8x16 --\*/

{0x08,0xF8,0x08,0x00,0x00,0x08,0xF8,0x08,0x00,0x1F,0x20,0x20,0x20,0x20,0x1F,0x00}

};

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*延时函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void delayus(uchar delay)

{

uchar i;

for(i=0;i<=delay;i++)

\_nop\_();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*忙检查\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void check\_busy()

{

uchar a=0;

for(a=0;a<200;a++); //此处为延时，一段时间后确保处于空闲状态

/\* //此处为状态查询，查询忙状态，没有使用成功，以后再调试

RW=1;

RS=0;

E=1;

while(1)

{

E=1;

if(!(P0&0x80))

break;

a++;

if(a>10)

break;

}

E=1;

\*/

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*写命令函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void write\_cmd(uchar inst)

{

check\_busy();

RS=0;

RW=0;

E=1;

P3=inst;

E=0;

\_nop\_();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*写数据函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void write\_data(uchar dat)

{

check\_busy();

RS=1;

RW=0;

E=1;

P3=dat;

E=0;

\_nop\_();

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*初始化函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void LCD\_Init()

{

write\_cmd(0x30);//30H--基本指令操作?

delayus(4);

write\_cmd(0x0c);//开显示，关光标，不闪烁。

delayus(10);

write\_cmd(0x01);//清除显示

delayus(10);

write\_cmd(0x06);//光标的移动方向左，DDRAM的地址计数器（AC）加1。

delayus(10);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*设置显示坐标\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void set\_xy(uchar x,uchar y) //x为行，y为列

{

if(y>=64) //列地址大于等于64时右屏显示?

{

CS2=0; //低电平选择

CS1=1; //写右屏是左屏关闭，不然两边会同时显示

y=y-64;

}

else //列地址小于64时左屏显示

{

CS1=0; //同上

CS2=1;

}

write\_cmd(0x40|y); //写列地址

\_nop\_();

write\_cmd(0xb8|x); //写行地址?

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*显示中文字体\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void disp\_hz(uchar R,uchar L,uchar \*p)

{

uchar i;

for(i=0;i<16;i++)

{

set\_xy(2\*R,16\*L+i); //设定显示位置，每个汉字16\*16个点，显示时每行8\*16个点，所以需要两行8\*32个点，此处为第一行

write\_data(p[i]); //写入DDRAM

set\_xy(2\*R+1,16\*L+i); //第二行

write\_data(p[i+16]);

}

}

void disp\_zf(uchar R,uchar L,uchar \*p)

{

uchar i;

for(i=0;i<8;i++)

{

set\_xy(2\*R,8\*L+i); //设定显示位置，每个字母16\*8个点，显示时每行8\*8个点，所以需要两行8\*16个点，此处为第一行

write\_data(p[i]);

set\_xy(2\*R+1,8\*L+i); //第二行

write\_data(p[i+8]);

}

}

void main()

{

uchar k=0,n=0;

LCD\_Init(); //初始化

for(k=0;k<6;k++)

disp\_hz(1,1+k,HZ0[k]); //显示安徽工业大学

for(n=0;n<10;n++)

disp\_zf(2,3+n,FZ0[n]); //显示 I LOVE YOU

while(1){;};

}