#include <reg52.h>

#include <intrins.h>

#include <stdio.h>

#include <string.h>

sfr AUXR = 0x8E; //AUXR手动添加地址，具体地址请查看手册

#define uchar unsigned char

#define uint unsigned int

#define ON 0

#define OFF 1

int z=0,count=0;

uchar buff;

uchar a,on;

uchar dat1[11]; //指令数组

uchar Random1[10]; //随机八位数数组

uchar code curtain\_order[][9]={ //串口接收到的指令在这个数组中进行对比

"01\_10\_01", //左右打开

"00\_10\_01", //左右闭合 "01\_11\_01", //左右打开 "00\_11\_01", //左右闭合 "11\_01\_01", //上升 "11\_00\_01", //下降 "10\_01\_01", //上升 "10\_00\_01", //下降 "00\_00\_00" //停止所有动作

};

unsigned char flag\_time=0,t0\_num=0,t1\_num=0,t2\_num=0,t3\_num=0;

unsigned int flag\_uart=0,Flag\_receive = 0,R\_Flag = 1;

void delayms(uint xms) //延时函数

{

uint i,j;

for(i=xms;i>0;i--)

for(j=110;j>0;j--);

}

void Heart\_Beat\_init () //上电后发送的心跳包（发5次）

{

uint x;

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

{

ES = 0;

TI = 1;

puts(

"{\"first\_type\":\"AABC\",\"second\_type\":\"AASX\",\"device\_number\":\"1\ ",\"current\_status\":\"00\_00\_00\",\"request\":0}"

);

while (!TI);

TI = 0;

ES = 1;

delayms(10000);

}

}

void init(void) //频率为115200 ，串口初始化

{

PCON &= 0x7F; //

SCON = 0x50; //

AUXR |= 0x40; //

AUXR &= 0xFE; //

TMOD &= 0x0F; //

TMOD |= 0x20; //

TH0 = (65536-50000)/256;

TL0 = (65536-50000)%256;

TL1 = 0xFD; //

TH1 = 0xFD; //

ET1 = 0; //

ET0 = 1;

TR1 = 1; //

EA=1; //

ES=1; //

}

void main() //main函数

{

init(); //串口初始化

delayms(50);

Heart\_Beat\_init(); //上电时候，发送上电心跳包5次

while(1)

{

if (flag\_uart == 1)

{

flag\_uart = 0;

R\_Flag = 0; //防止接受到错误指令标志位 for(a=0;a<9;a++)

{

if (memcmp(dat1,curtain\_order[a],8)==0)

{

if(a==0){on=0;L\_MOTOR\_Rotation= ON;delayms(1500);L\_MOTOR\_Rotation = OFF;} if(a==1){on=1;L\_MOTOR\_Reverse = ON;delayms(1500);L\_MOTOR\_Reverse = OFF;}

if(a==2){on=2;L\_MOTOR\_Rotation = ON;delayms(1500);L\_MOTOR\_Rotation = OFF;}

if(a==3){on=3;L\_MOTOR\_Reverse = ON;delayms(1500);L\_MOTOR\_Reverse = OFF;}

if(a==4){on=4;window\_up = ON;delayms(50);window\_up = OFF;}

if(a==5){on=5;window\_down = ON;delayms(50);window\_down = OFF;}

if(a==6){on=6;window\_up = ON;delayms(50);window\_up = OFF;}

if(a==7){on=7;window\_down = ON;delayms(50);window\_down = OFF;}

if(a==8){on=8;window\_up=OFF;window\_down=OFF;L\_MOTOR\_Stop=ON;delayms(1500);L\_MOTOR\_Stop = OFF;}

R\_Flag = 1;

ES = 0;

TI = 1;

printf("{\"first\_type\":\"AABC\",\"second\_type\":\"AASX\",\"device\_number\":\"1\",\"current\_status\":\"%s\",\"request\":\"%s\"}\n",dat1,Random1);//接受到指令后，立即发出的心跳包

while(!TI);

TI=0;

ES=1;

TH0 = (65536-50000)/256;

TL0 = (65536-50000)%256;

t0\_num = 0;

t1\_num = 0;

t2\_num = 0;

t3\_num = 0;

TR0 = 1;

}

}

}

if ((flag\_time == 1) && (flag\_uart == 0))

{

if(R\_Flag == 1)

{

flag\_time = 0;

ES = 0;

TI = 1;

printf("{\"first\_type\":\"AABC\",\"second\_type\":\"AASX\",\"device\_number\":\"1\",\"current\_status\":\"%s\",\"request\":\"%s\"}\n",dat1,Random);//从上一次接受到正确指令后，每隔一段时间（可调，现在大约为一分钟）发出的心跳包

while(!TI);

TI = 0;

ES = 1;

}

if(R\_Flag ==0)

{

flag\_time = 0;

dat1[11] = {0};

Random1[10] = {0};

ES = 0;

TI = 1;

printf("{\"first\_type\":\"AABC\",\"second\_type\":\"AASX\",\"device\_number\":\"1\",\"current\_status\":\"%s\",\"request\":\"%s\"}\n",dat1,Random);/从上一次接受到错误指令后，每隔一段时间（可调，现在大约为一分钟）发出的心跳包，直到接受到正确指令后停止

while(!TI);

TI = 0;

ES = 1;

}

}

}

}

void timer0() interrupt 1 //定时器中断

{

TH0 = (65536-50000)/256;

TL0 = (65536-50000)%256;

t0\_num++;

if (t0\_num == 255)

{

t1\_num++;

if(t1\_num==1)

{

t0\_num=0;

t1\_num=0;

flag\_time=1;

}

}

t2\_num++; //

if(t2\_num == 255)

{

t3\_num++;

if(t3\_num==20)

{

if(flag\_time==0)

{

window\_up = OFF;

window\_down = OFF;

t2\_num = 0;

t3\_num = 0;

}

}

}

void ser() interrupt 4 //串口中断

{

RI=0;

if( SBUF == '#' )

{

Flag\_receive = 1;

buff = SBUF;

}

if( SBUF == '$' )

{

buff = SBUF;

}

if( SBUF == '@' )

{

buff = SBUF;

}

if( buff == '#')

{

if(SBUF != '#')

{

dat1[z] = SBUF;

z++;

if(z > 7)

{

z=0;

dat1[8] = '\0';

}

}

}

if( buff == '$')

{

if(SBUF != '$')

{

Random1[count] = SBUF;

count++;

if(count > 7)

{

count=0;

Random1[8] = '\0';

}

}

}

if( buff == '@' && flag\_uart == 0 && Flag\_receive == 1)

{

flag\_uart = 1;

Flag\_receive = 0;

}

}