**PWM驱动风扇并在LCM12864显示状态**

**第一步**：硬件电路连接：LCM12864\_date->GPIOA[7:0] LCD1602\_contr->GPIOB[4:0]

风扇驱动控制线->PC4 外部中断按键->PC13

**第二步**：创建工程并注释掉原有驱动程序的入口

**第三步**：添加变量与函数的声明定义等

void delay(void);

void lcdwritecmd(unsigned char cmd);

void lcdwritedata(unsigned char dat);

void lcdinit(void);

void lcdsetcursor(unsigned char x, unsigned char y);

void lcdshowstr(unsigned char x, unsigned char y, unsigned char \*str);

int speed=0;

unsigned char tstri[2];

int location=4;

int counter=0;

unsigned char tstr[5];

**第四步**：自定义函数获取对应管脚中断状态以及释放其中断标志位

ITStatus EXTI\_GetITStatus(uint32\_t EXTI\_Line)

{

ITStatus bitstatus = RESET; //chushi wei zhuangtai 0

uint32\_t enablestatus = 0; //chushi shineng zhuangtai 0

/\* Check the parameters \*/

assert\_param(IS\_GET\_EXTI\_LINE(EXTI\_Line));

enablestatus = EXTI->IMR1 & EXTI\_Line;

if (((EXTI->FPR1 & EXTI\_Line) != (uint32\_t)RESET) && (enablestatus != (uint32\_t)RESET))

bitstatus = SET;

else

bitstatus = RESET;

return bitstatus;

}

void EXTI\_ClearITPendingBit(uint32\_t EXTI\_Line)

{

/\* Check the parameters \*/

assert\_param(IS\_EXTI\_LINE(EXTI\_Line));

EXTI->FPR1 = EXTI\_Line;

}

**第五步**：在主函数中添加运行逻辑

①在while(1)之前的初始状态

lcdinit();

delay();

lcdshowstr(0,0,"Level");

lcdshowstr(3,0,"0");

lcdshowstr(5,1,"times");

sprintf(tstr,"%d",counter);

lcdshowstr(location,1,tstr);

②在while(1)之间的循环运行

if(speed==0)

HAL\_GPIO\_WritePin(GPIOC,GPIO\_PIN\_4,RESET);

else if((speed>0)&&(speed<5))

{

HAL\_GPIO\_WritePin(GPIOC,GPIO\_PIN\_4,SET);

HAL\_Delay(speed);

HAL\_GPIO\_WritePin(GPIOC,GPIO\_PIN\_4,RESET);

HAL\_Delay(5-speed);

}

else if(speed==5)

HAL\_GPIO\_WritePin(GPIOC,GPIO\_PIN\_4,SET);

else

speed=0;

**第六步**：补充子函数

void delay ()

{

for(int i=0;i<99;i++)

for(int j=0;j<99;j++)

{}

}

void lcdwritecmd(unsigned char cmd)

{

delay(); // 使能 写 指令

GPIOB->ODR=0x18; //RST=1\PSB=1\E=0\RW=0\RS=0

GPIOA->ODR=cmd;

GPIOB->ODR=0x1C; //RST=1\PSB=1\E=1\RW=0\RS=0

delay();

GPIOB->ODR=0x18; //RST=1\PSB=1\E=0\RW=0\RS=0

}

void lcdwritedata(unsigned char dat)

{

delay(); // 使能 写 数据

GPIOB->ODR=0x19; //RST=1\PSB=1\E=0\RW=0\RS=1

GPIOA->ODR=dat;

GPIOB->ODR=0x1D; //RST=1\PSB=1\E=1\RW=0\RS=1

delay();

GPIOB->ODR=0x19; //RST=1\PSB=1\E=0\RW=0\RS=1

}

void lcdinit()

{

lcdwritecmd(0x38); //E 使能 RS1数据0指令 RW1读0写

lcdwritecmd(0x06);

lcdwritecmd(0x01);

lcdwritecmd(0x0C);

}

void lcdsetcursor(unsigned char x, unsigned char y) //字符定位

{

unsigned char address;

if(y==0)

address=0x80+x;

else if(y==1)

address=0x90+x;

else if(y==2)

address=0x88+x;

else

address=0x98+x;

lcdwritecmd(address|0x80);

}

void lcdshowstr(unsigned char x, unsigned char y, unsigned char \*str) //字符输出

{

lcdsetcursor(x,y);

while((\*str)!='\0')

{

lcdwritedata(\*str);

str++;

}

}

void EXTI4\_15\_IRQHandler(void)

{

if (EXTI\_GetITStatus(0x2000) != 0x00) // 0x2000==EXTI\_PIN\_13

{

EXTI\_ClearITPendingBit(0x2000); //Clear interrupt flag bit

if(speed==5)

{

speed=0;

sprintf(tstri,"%d",speed);

lcdshowstr(3,0,tstri);

counter++; //interrupt users code

if(counter>99) lcdshowstr(0,1," !!!ERROR!!! ");

else if(counter>9) location=3;

sprintf(tstr,"%d",counter);

lcdshowstr(location,1,tstr);

}

else

{

speed++;

sprintf(tstri,"%d",speed);

lcdshowstr(3,0,tstri);

}

}

}

**第七步**：效果说明

风扇档变速：0、20%、40%、60%、80%、100%

每当按下外部中断按键风扇转速变档。

每次启动，风扇启动次数加一。

LCM12864上显示风扇的裆位以及启动次数。

**第八步**：效果图展示

