呼吸灯

PWM

一种通过改变信号脉冲的宽度来控制输出功率的技术

1. 通过TIMx外设实现

   配置TIM2的PWM输出

```
void TIM_PWM_Init(uint16_t arr,uint16_t psc)
{
 GPIO_InitTypeDef GPIO_InitStructure;
 TIM_TimeBaseInitTypeDef  TIM_TimeBaseStructure;
 TIM_OCInitTypeDef  TIM_OCInitStructure;

 RCC_APB1PeriphClockCmd(RCC_APB1Periph_TIM2,ENABLE);
 RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA,ENABLE);
 //使能时钟
 GPIO_InitStructure.GPIO_Mode=GPIO_Mode_AF_PP;
 GPIO_InitStructure.GPIO_Pin=GPIO_Pin_0;
 GPIO_InitStructure.GPIO_Speed=GPIO_Speed_50MHz;
 GPIO_Init(GPIOA,&GPIO_InitStructure);
 //配置GPIO口
 TIM_TimeBaseStructure.TIM_ClockDivision=0;
 TIM_TimeBaseStructure.TIM_CounterMode=TIM_CounterMode_Up;
 TIM_TimeBaseStructure.TIM_Period=arr;
 TIM_TimeBaseStructure.TIM_Prescaler=psc;  //psc为预分频器，一般为72MHz，此处
TIM2不起定时作用，只生成PWM信号
 TIM_TimeBaseInit(TIM2,&TIM_TimeBaseStructure);
 //配置TIM2

 TIM_OCInitStructure.TIM_OCMode=TIM_OCMode_PWM1;
 TIM_OCInitStructure.TIM_OutputState=TIM_OutputState_Enable;
 TIM_OCInitStructure.TIM_Pulse=0;
 TIM_OCInitStructure.TIM_OCPolarity=TIM_OCPolarity_High;
    TIM_OC1Init(TIM2, &TIM_OCInitStructure);
 //output compare,配置为PWM模式，设置比较值

 TIM_Cmd(TIM2,ENABLE);
    //使能TIM2
 }
```

2. 呼吸灯函数

   - 设置比较值：即TIM_Pulse

```
TIM_SetCompare1(TIM_TypeDef* TIMx, uint16_t Compare1);
```

通道：1 =>Pin_0; 2 =>Pin_1; 3 =>Pin_2; 4 =>Pin_3

- 预重装值：即TIM_period
- 通过direction变量控制brightness变化的方向，brightness即为CCR的值。**brightness/（arr+1）即为占空比**

代码 (改进后)

```
static int direction=1;
void BREATHING_LIGHT(void){
 if(direction){
    brightness++;
    if(brightness>=1000){
        direction=0;
    }
 }

 if(!direction){
    if(!brightness){
        direction=1;
    }
    brightness--;
 }
 TIM_SetCompare1(TIM2,brightness);
 TIM_SetCompare2(TIM2,brightness);
}
```