#include"stm32f10x.h"

#include<stdio.h>

#define ITM\_Port8(n)

#define ITM\_Port16(n)

define ITM\_Port32(n)

#define DEMCR

#define TRCENA

struct\_ \_FILE{int handle;}

FILE\_ \_stdout;

FILE\_ \_stdin;

int fputc(int ch, FILE \*f)

{

if(DEMCR & TRCENA)

{

while(ITM\_Port32(0)==0);

ITM\_Port8(0)=ch;

}

return(ch);

}

volatile unsigned vhar flag=0;

unsigned int CountOfToggle=0;

void LED0\_Config(void);

void LED0\_On(void);

void LED0\_Off(void);

unsigned char LED0\_IsOn(void);

void NVIC\_Config(void);

void TIM2\_Config(void);

int main(void)

{

LED0\_Config();

NVIC\_Config();

TIM2\_Config();

while(1)

{

if(flag)

{

if(LED0\_IsOn())

{

LED0\_Off();

}

else

{

CountOfToggle++;

printf("CountOfToggle is %d\n",CountOfToggle);

LED0\_On();

}

flag=0;

}

}

}

void NVIC\_Config()

{

NVIC\_InitTypeDef NVIC\_InitStructure;

NVIC\_PriorityGroupConfig(NVIC\_PriorityGroup\_1);

NVIC\_InitStructure.NVIC\_IRQChannel=TIM2\_IRQn;

NVIC\_InitStructure.NVIC\_IRQChannelPreemptionPriority=0;

NVIC\_InitStructure.NVIC\_IRQChannelSubPriority=1;

NVIC\_InitStructure.NVIC\_IRQChannelCmd=ENABLE;

NVIC\_Init(&NVIC\_InitStructure);

}

void TIM2\_Config()

{

TIM\_TimeBseInitTypeDef TIM\_TimeBseStructure;

RCC\_APB1PeriphColckCme(RCC\_APB1Periph\_TIM2,ENABLE);

TIM\_TimeBaseStructure.TIM\_Prescaler=36000-1;

TIM\_TimeBaseStructure.TIM\_Period=9000-1;

TIM\_TimeBaseStructure.TIM\_ClockDivision=0;

TIM\_TimeBaseStructure.TIM\_CounterMode=TIM\_CounterMode\_Up;

TIM\_TimeBaseInit(TIM2,&TIM\_TimeBseStructure)

TIM\_ClearFlag(TIM2,TIM\_FLAG\_Update);

TIM\_ITConfig(TIM2,TIM\_IT\_Update,ENABLE);

TIN\_Cmd(TIM2,ENABLE);

}

void LED0\_Config(void)

{

GPIO\_InitTypeDef GPIO\_InitSturcture;

RCC\_APB2PeriphClockCmd(RCC\_APB2Periph\_GPIOA,ENABLE);

GPIO\_InitSturcture.GPIO\_Pin\_8;

GPIO\_InitStructure.GPIO\_Mode=GPIO\_Mode\_Out\_PP;

GPIO\_InitStructure.GPIO\_Speed=GPIO\_Speed\_2MHz;

GPIO\_Init(GPIOA,&GPIO\_InitStructure);

}

void LED0\_On(void)

{

GPIO\_ResetBits(GPIOA,GPIO\_Pin\_8);

}

void LED0\_Off(void)

{

GPIO\_SetBits(GPIOA,GPIO\_Pin\_8;

}

unsigned char LED0\_IsOn(void)

{

return !GPIO\_ReadOutputDataBit(GPIOA.GPIO\_Pin\_8);

}

#include "stm32f10x\_it.h"

extern volatile unsigned char flag;

void TIM2\_IRQHandler(void)

{

if(TIM\_GetITStatus(TIM2,TIM\_IT\_Update)!=RESET)

{

flag=1;

TIM\_ClearITPendingBit(TIM2,TIM\_IT\_Update);

}

}

 