```
#define RCC_BASE (0x40021000)
#define GPIOC_BASE                                                  (0x40011000)
#define TIM2_BASE                                                            (0x40000000)

#define RCC_APB2ENR                                         (*(volatile unsigned int *)(RCC_BASE + 0x18))
#define RCC_APB1ENR                                         (*(volatile unsigned int *)(RCC_BASE + 0x1C))
#define RCC_CFGR                                                     (*(volatile unsigned int *)(RCC_BASE + 0x04))

#define TIM2_ARR      (*(volatile unsigned int *)(TIM2_BASE + 0x2C))
#define TIM2_PSC      (*(volatile unsigned int *)(TIM2_BASE + 0x28))
#define TIM2_SR      (*(volatile unsigned int *)(TIM2_BASE + 0x10))
#define TIM2_CR1                                            (*(volatile unsigned int *)(TIM2_BASE + 0x00))

#define GPIOC_CRH                                                            (*(volatile unsigned int *)(GPIOC_BASE + 0x04))
#define GPIOC_ODR                                                            (*(volatile unsigned int *)(GPIOC_BASE + 0x0C))

void delay(int s);

int main(void) {

        RCC_CFGR &= ~(3 << 0);
        while((RCC_CFGR & (3 << 2)) != (0 << 2));

        RCC_APB2ENR |= (1 << 4);
        RCC_APB1ENR |= (1 << 0);

        GPIOC_CRH &= ~(3 << 20);
        GPIOC_CRH |= (1 << 21);
        GPIOC_CRH &= ~(3 << 22);

        while(1) {
                GPIOC_ODR |= (1 << 13);
                delay(1000); //ms


                GPIOC_ODR &= ~(1 << 13);
                delay(1000); //ms
        }
}

void delay(int s) {
        TIM2_PSC = 8000 - 1; // 1000 Hz
        TIM2_ARR = s - 1;

        // Clear update flag
 TIM2_SR &= ~(0x01);

        // Enable
        TIM2_CR1 |= (1 << 0);

        while(!(TIM2_SR & 0x01));

        // Disable
        TIM2_CR1 &= ~(1 << 0);

}
```