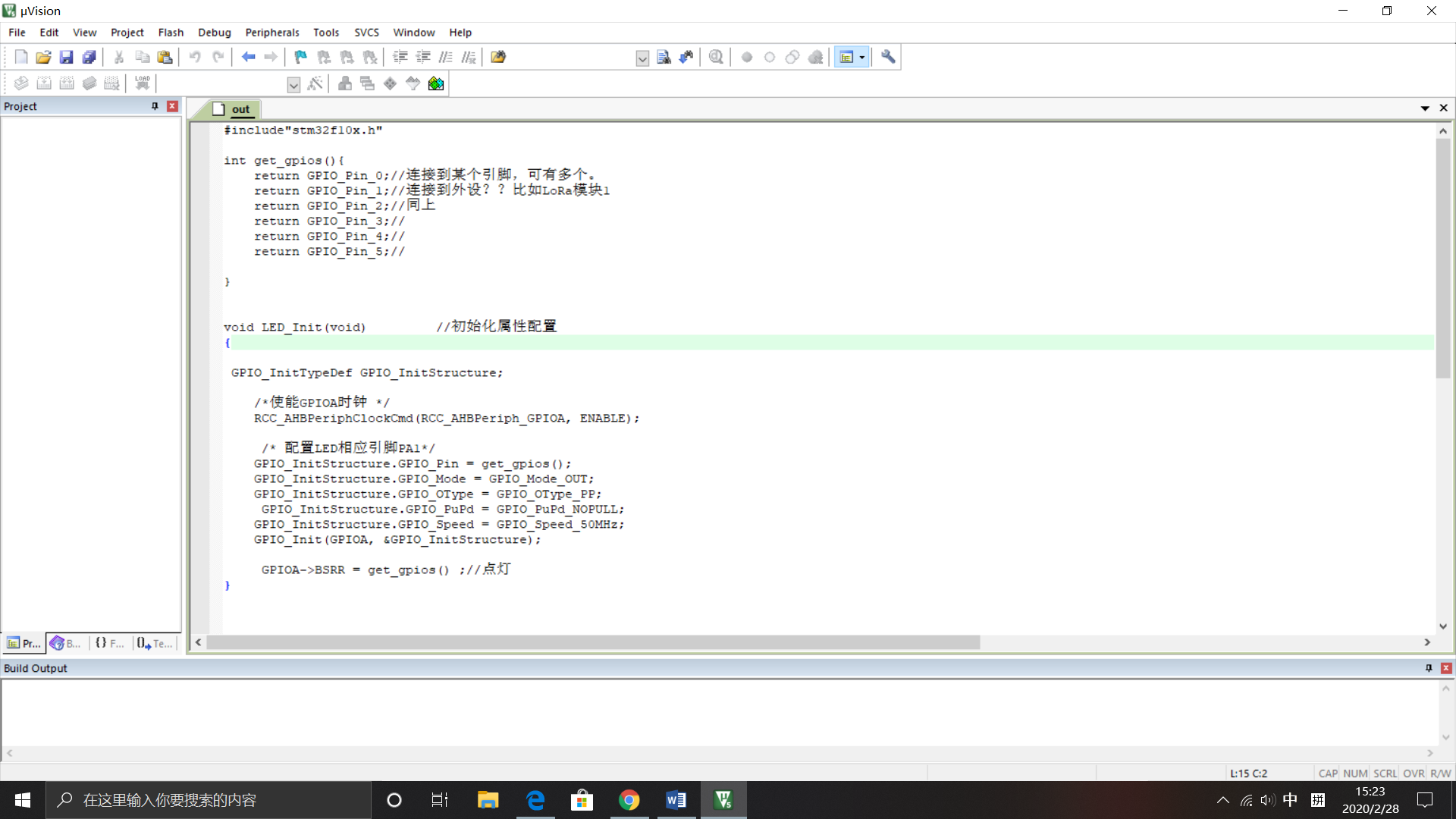
进度报告

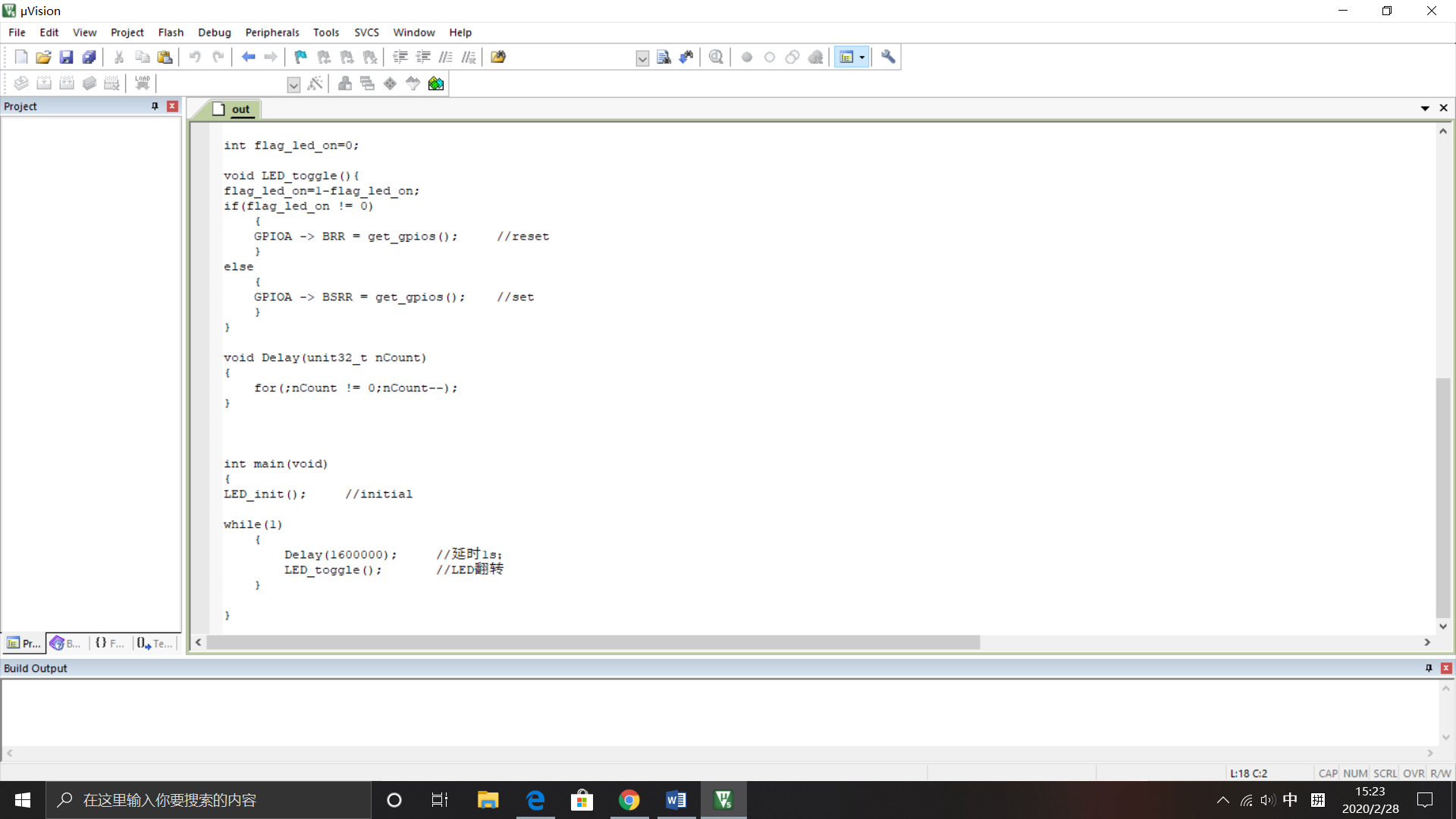
一、学习stm32使用方法，进行功能学习和简单程序练习

Stm32和之前使用过的freescale单片机有若干相似之处，都需要对口的寄存器进行操作从而实现功能的选择和数据的流入流出。虽然在使用的便利程度上不如Arduino单片机，但其性能和性价比是超越Arduino单片机的。

下面是对stm’32初步学习的简单程序，点亮LED灯实验，熟悉操作。为下一步使用单片机对即将连接的外设模块如振动传感器和LoRa模块做准备，学习时钟的控制。

此处创建工程使用的是stm32f103RC这一款芯片，仅作练习用，以后可能并不使用，不代表未来芯片挑选。





二、查询LoRa模块WH-L101-L-C、AS30M-TTL-100等的性能参数。

查询技术手册，确定其与stm32的相关连接方式。

1. 串口传输  
   struct \_\_FILE  
   {  
           int handle;  
     
   };  
   FILE \_\_stdout;      //重定义fputc函数

int fputc(int ch, FILE \*f)  
{     
        //用串口实现一个字符的传输，基于 STM32HAL库。  
        HAL\_UART\_Transmit(&huart1,(uint8\_t \*)&ch,1,50);    
        return ch;  
}

1. 连续输出

if(USART\_RX\_STA&0x8000)//接受到中断，就执行保存

{

USART\_RX\_STA=0;//对数组清零，方便下一次接收

len=USART\_RX\_STA&0x3fff;//取出u16中的低16 位，得到此次接收到的数据长度

for(t=0;t<len;t++)

{

USART1->DR=USART\_RX\_BUF[t];//把每一个数据都在串口打印出来

while((USART1->SR&0X40)==0);//等待发送结束

}

USART\_RX\_STA=0;//对数组清零，方便下一次接收

if(keys==1)

{

printf("%c",j);

delay\_ms(200);

printf("%c",j);

delay\_ms(100);

printf("%c",j);

delay\_ms(100);

}

delay\_ms(400);

printf("%c",M);

delay\_ms(500);

printf("AT+AID=00000001\r\n");

delay\_ms(400);

printf("AT+NID=00013c2d\r\n");

delay\_ms(400);

printf("AT+SPD=1\r\n");

delay\_ms(400);

printf("AT+CH=72\r\n");

delay\_ms(400);

printf("AT+ENTM\r\n");

delay\_ms(400);

printf("AT+ENTM\r\n");

LED=!LED;

delay\_ms(1000);

delay\_ms(1000);

delay\_ms(1000);

LED=!LED;

// printf("AT+AID\r\n");//可以查询设置的数据

}