# 嵌入式系统实验报告



|  |  |
| --- | --- |
| 实验名称： | Lab1 实验环境安装与验证 |
| 姓 名： | 王小龙 |
| 学 号： | 2020211502 |
| 学 院(系)： | 计算机学院 |
| 专 业： | 网络工程 |
| 指导教师： | 戴志涛、刘健培 |

2022年 11 月 11 日

# 实验目的

完成MDK开发环境的安装，并创建示例项目验证安装过程。

进行相关的测试。

# 实验环境

1、Microsoft Windows 64位操作系统的 PC

2、MDK软件

3、STM32CubeMx软件

# 实验要求

1、按照以下实验过程的说明阅读相关材料，完成 MDK 开发环境的安装，并创建示例项目验证安装过程。

2、完成要求的实验过程，将过程与分析结果截图贴在作业答卷里。

# 实验原理

LED 对应的芯片引脚是 PORTA.5（PORT A 的第 5 引脚）。

按键对应的芯片引脚是PORTC.13（PORT C的第13引脚）。

Blinky.c 在 main 的 for 死循环中，每隔 0.5 秒，翻转一次 LED 灯的状态

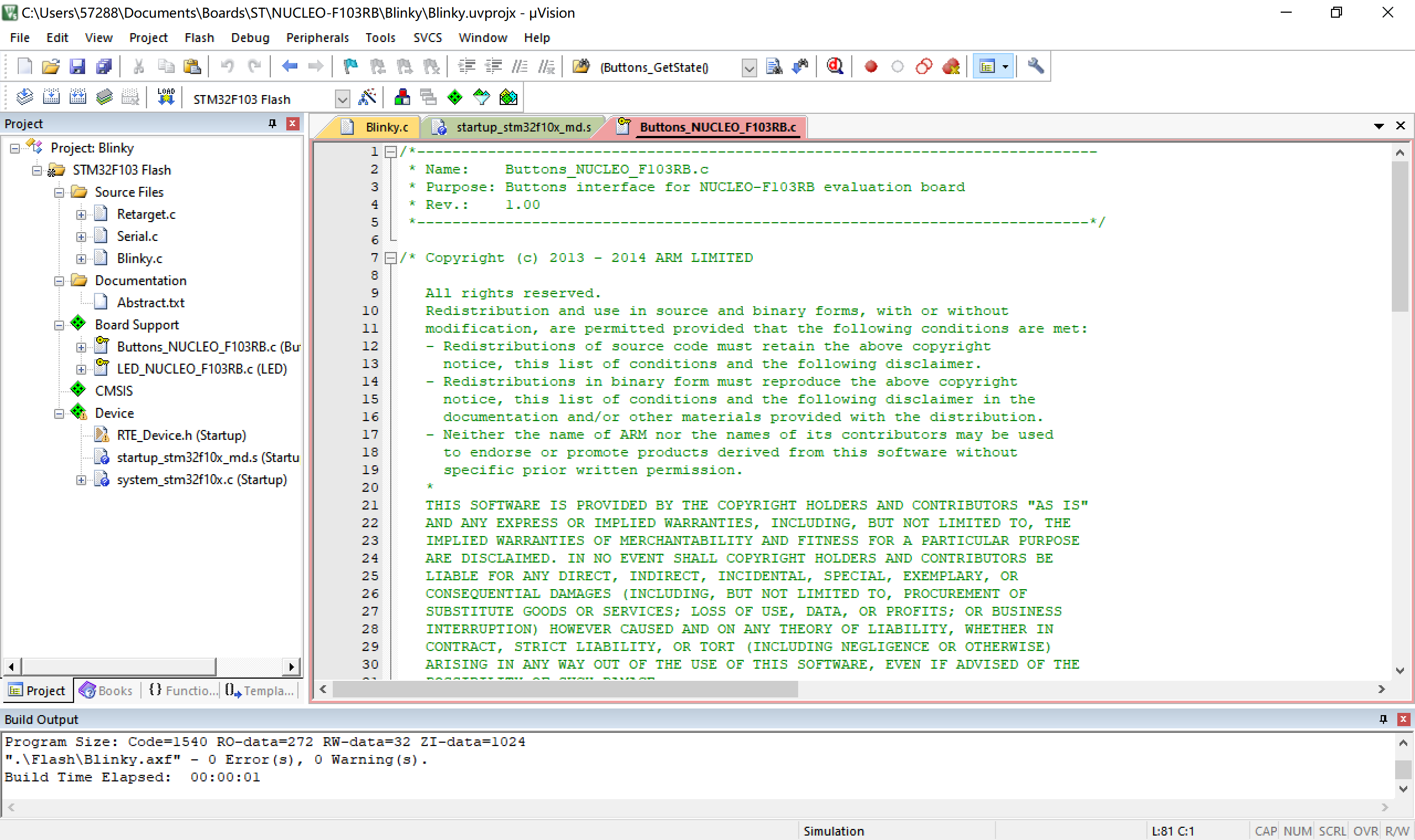
（LED\_On(num)- Delay(500)- LED\_Off(num)- Delay(500)）。

本实验通过模拟一个按键被按下再抬起的过程来观察LED灯的变化。

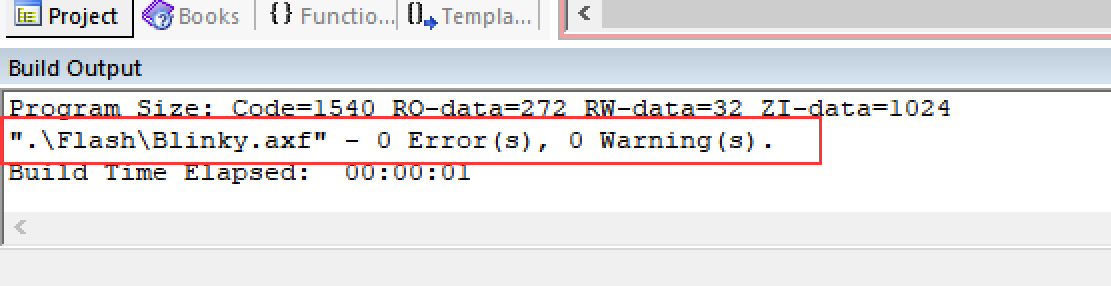
# 实验步骤

1.安装好MDK等软件及其相关需要的组件；

2.导入blinky（NUCLEO-F103RB）示例项目，如下图：

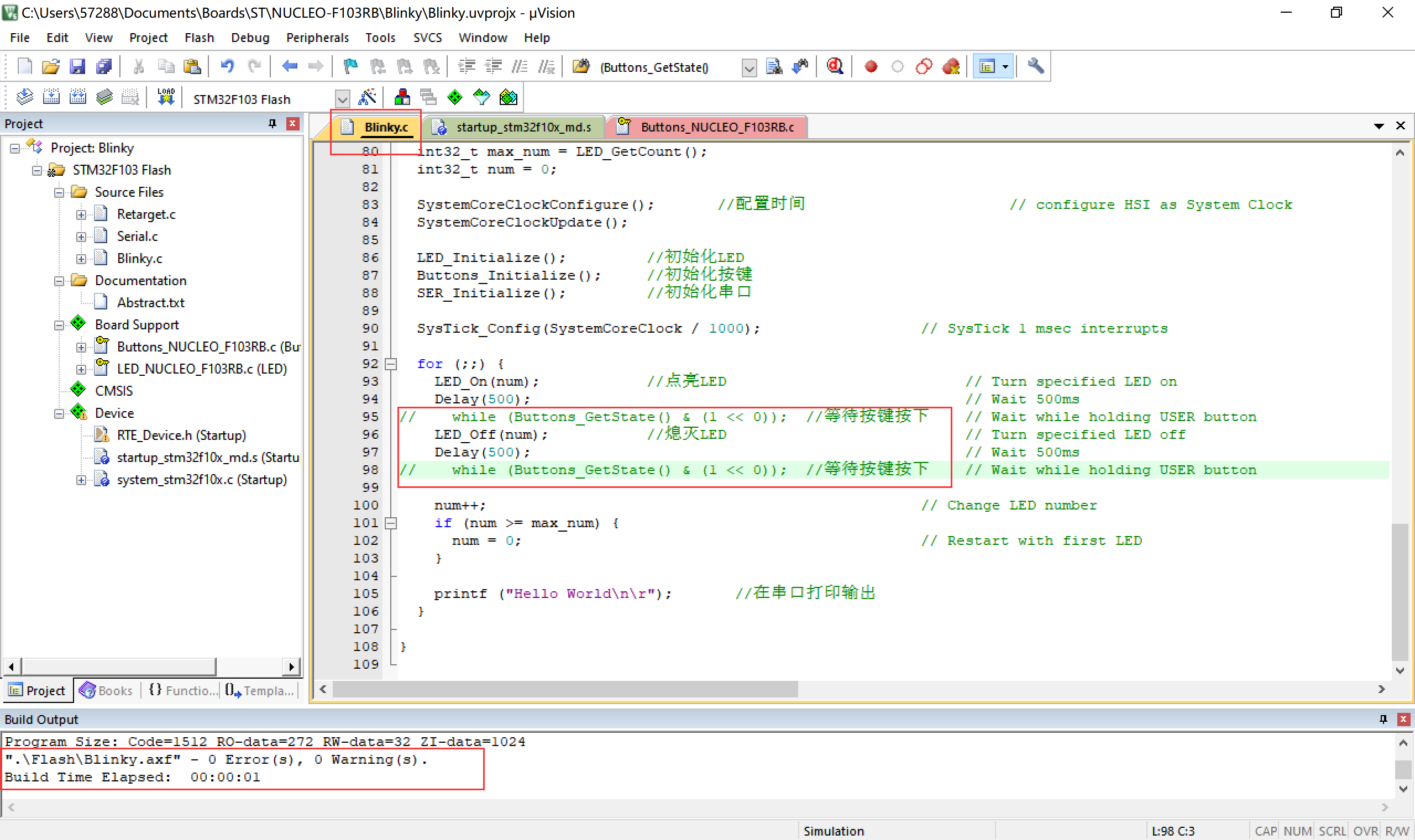


3.测试编译成功后，进行各项配置；

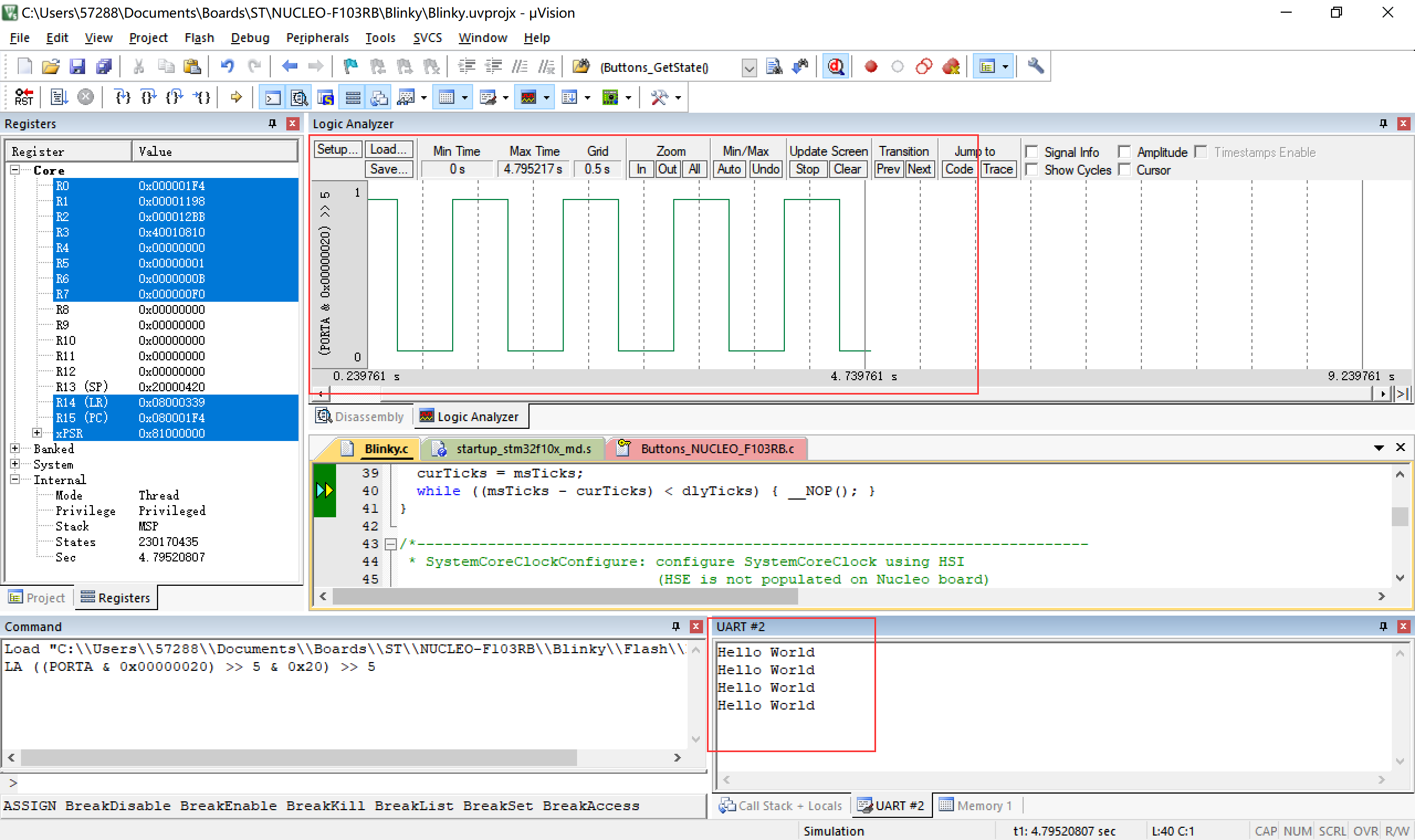


4.打开文件 Blinky.c，注释掉 while (Buttons\_GetState() & (1 << 0));（这两行是轮询读取按键状态，直到按键被按下才返回，模拟器模拟输入稍复杂，先跳过）。

重新编译。

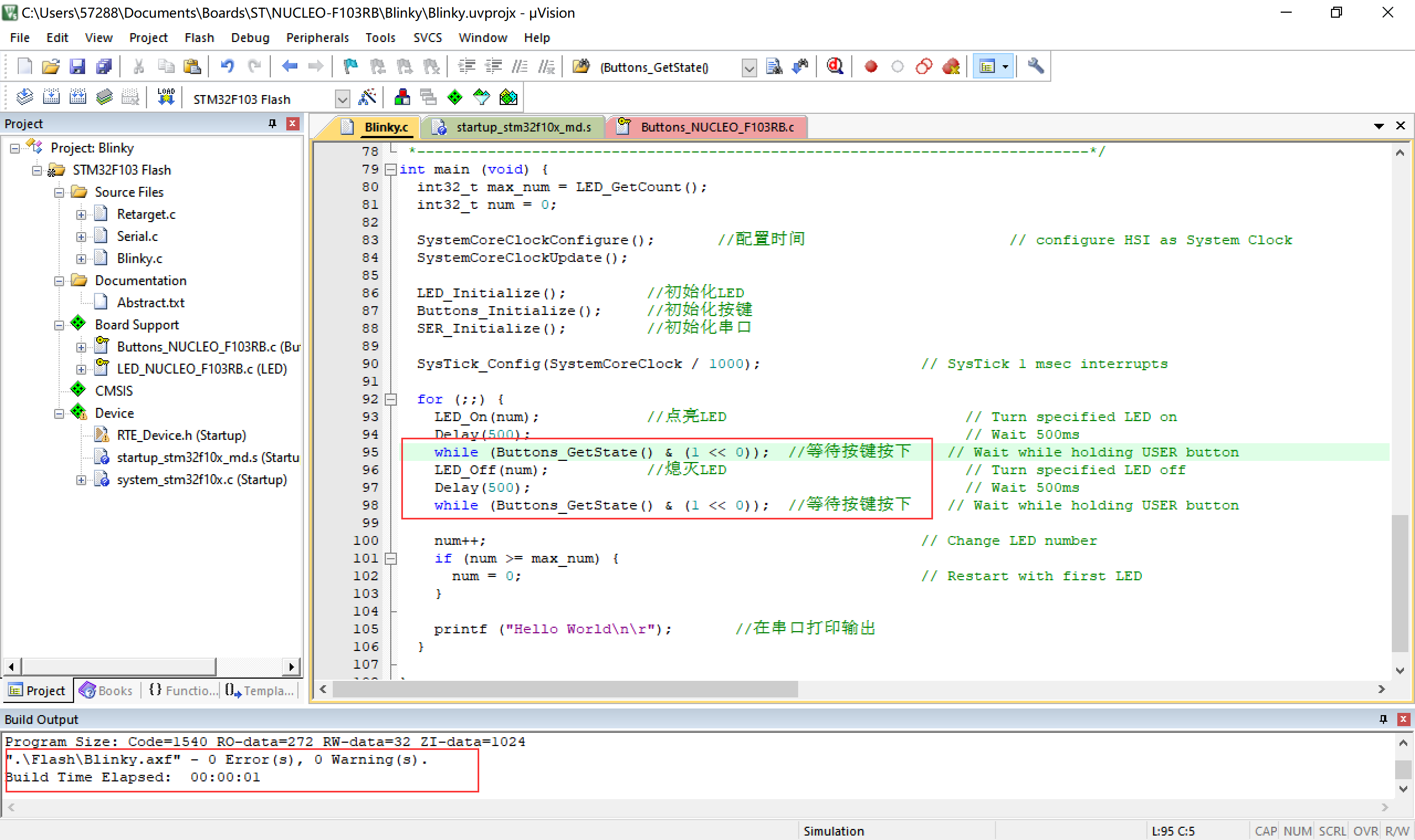


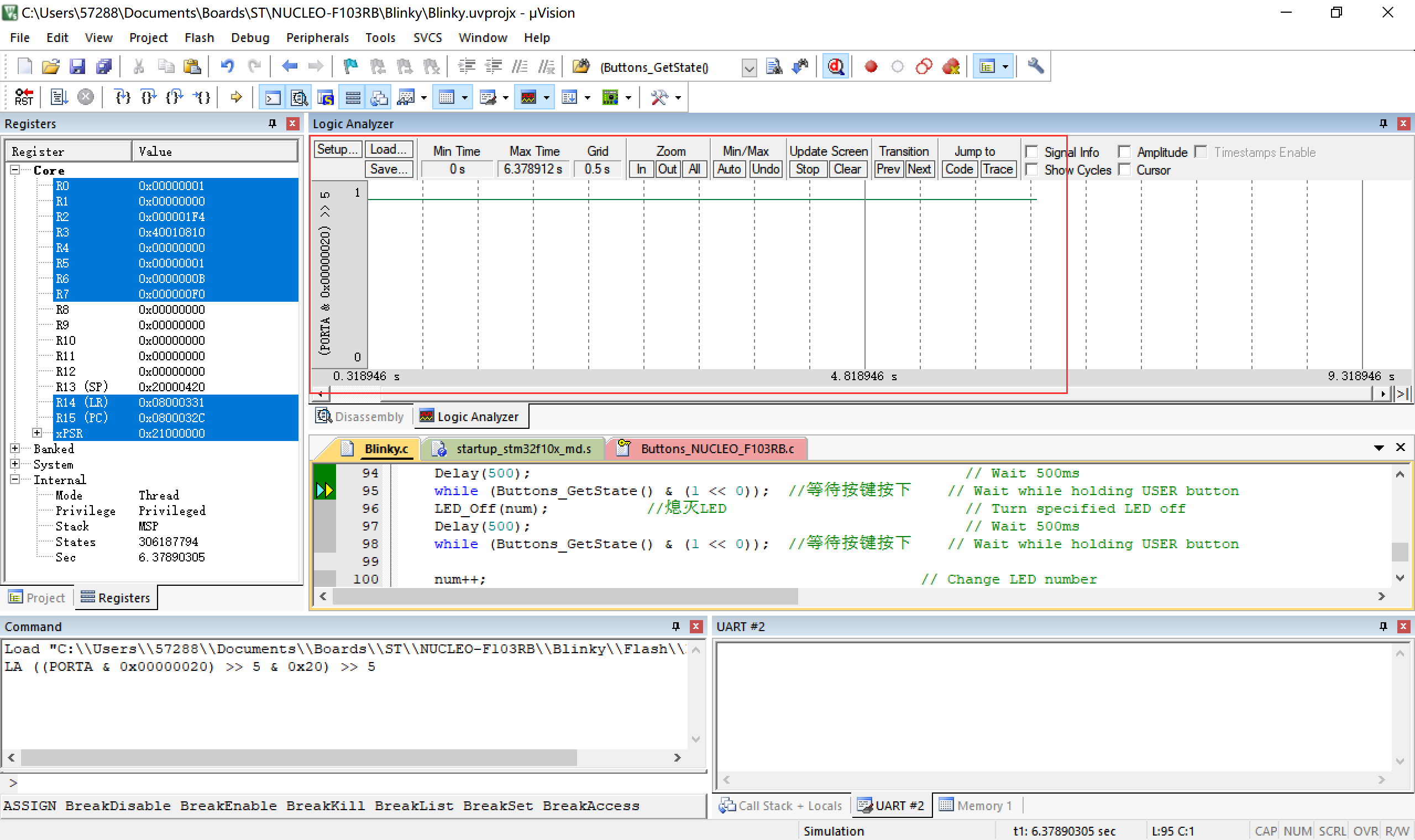
5.启动调试，添加观测信号PORTA.5并运行，可得如下运行结果：



从图中，可以观察到程序软件仿真运行期间的 PA5 信号和 UART2 输出符合预期，仿真结果正确。

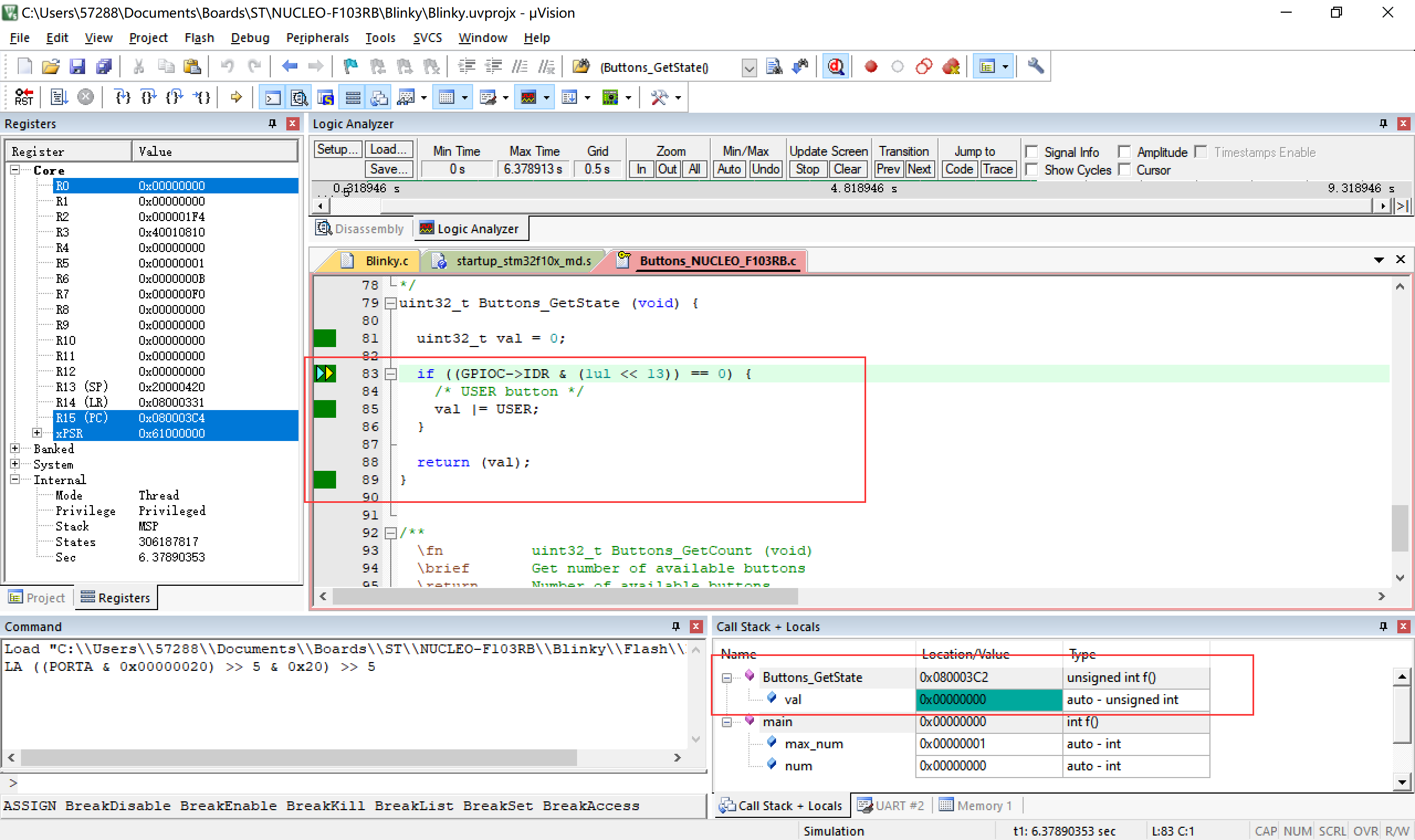
6.取消注释，还原代码，重新编译，启动调试，运行可得下图：



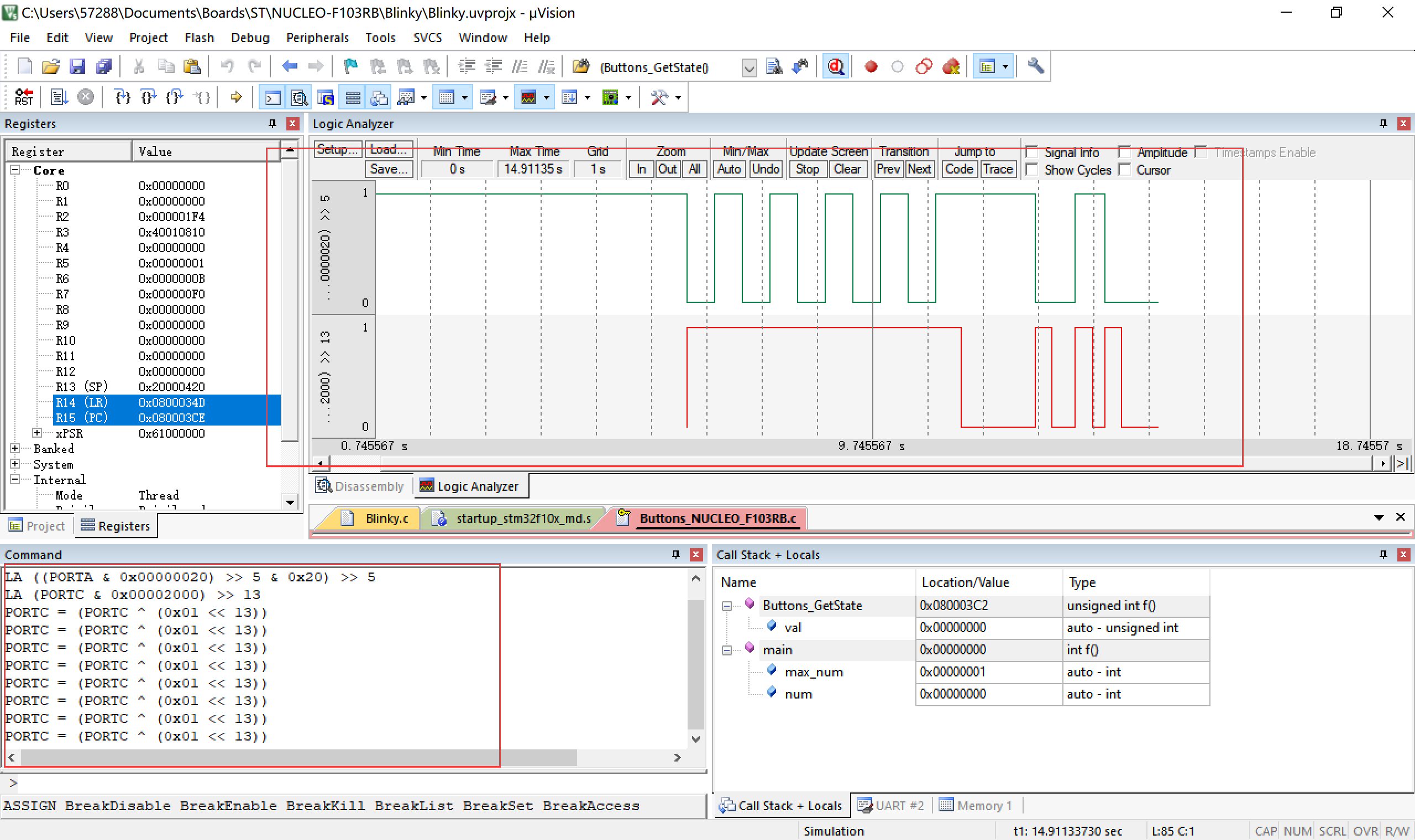


观察 logic analyzer 窗口，显示 LED 的 PORTA.5 一直处于高电平。

单步跟踪代码，会发现程序一直在 main 中调用在 Buttons\_GetState 函数，原因是 button 对应 PORTC.13 的值没有改变过，如下图：

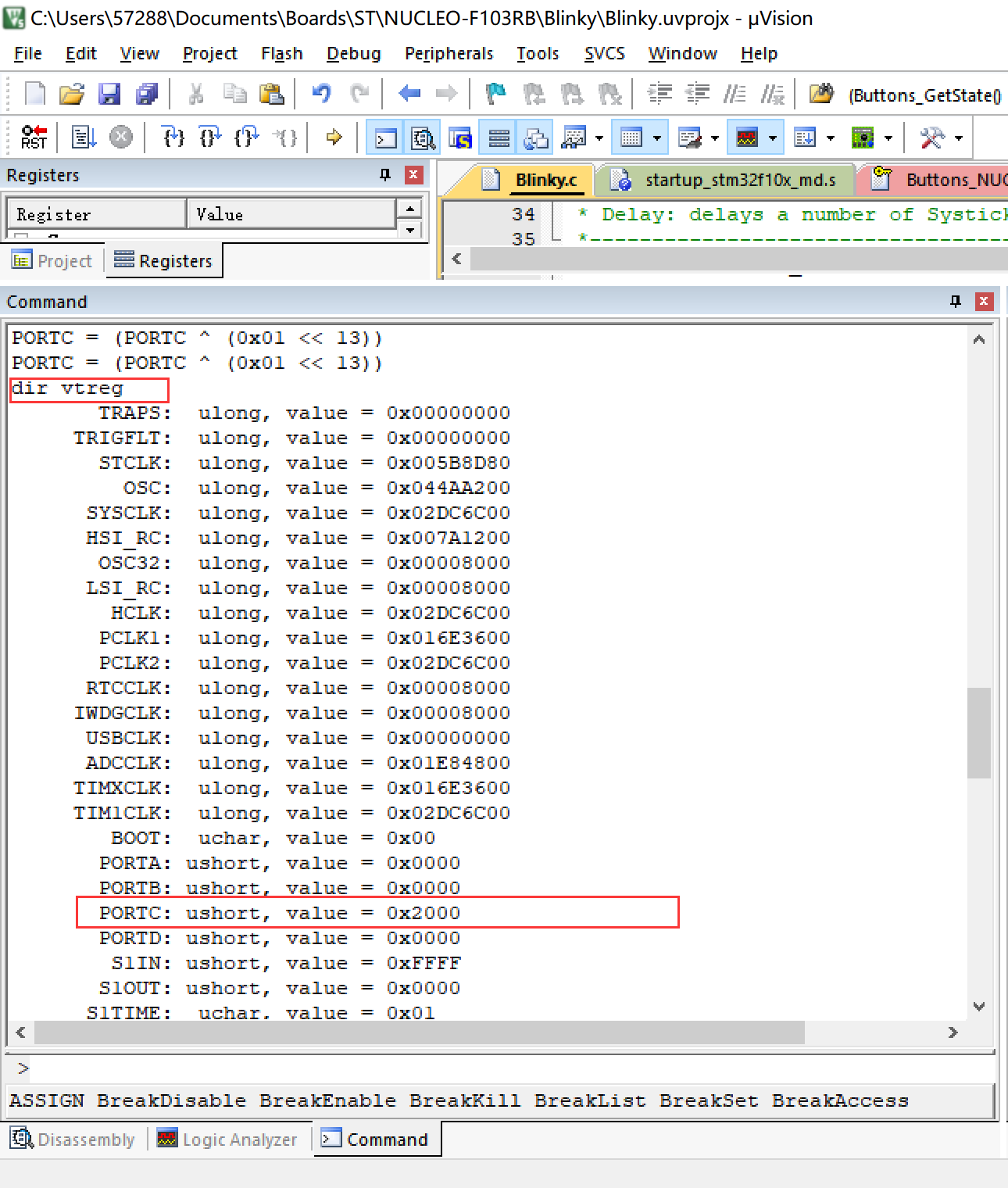


7.将 PORTC.13 加入到 logic analyzer 的观察窗口，在 command 窗口的命令提示符下输入命令：PORTC = (PORTC ^ (0x01 << 13))，结果如下图：

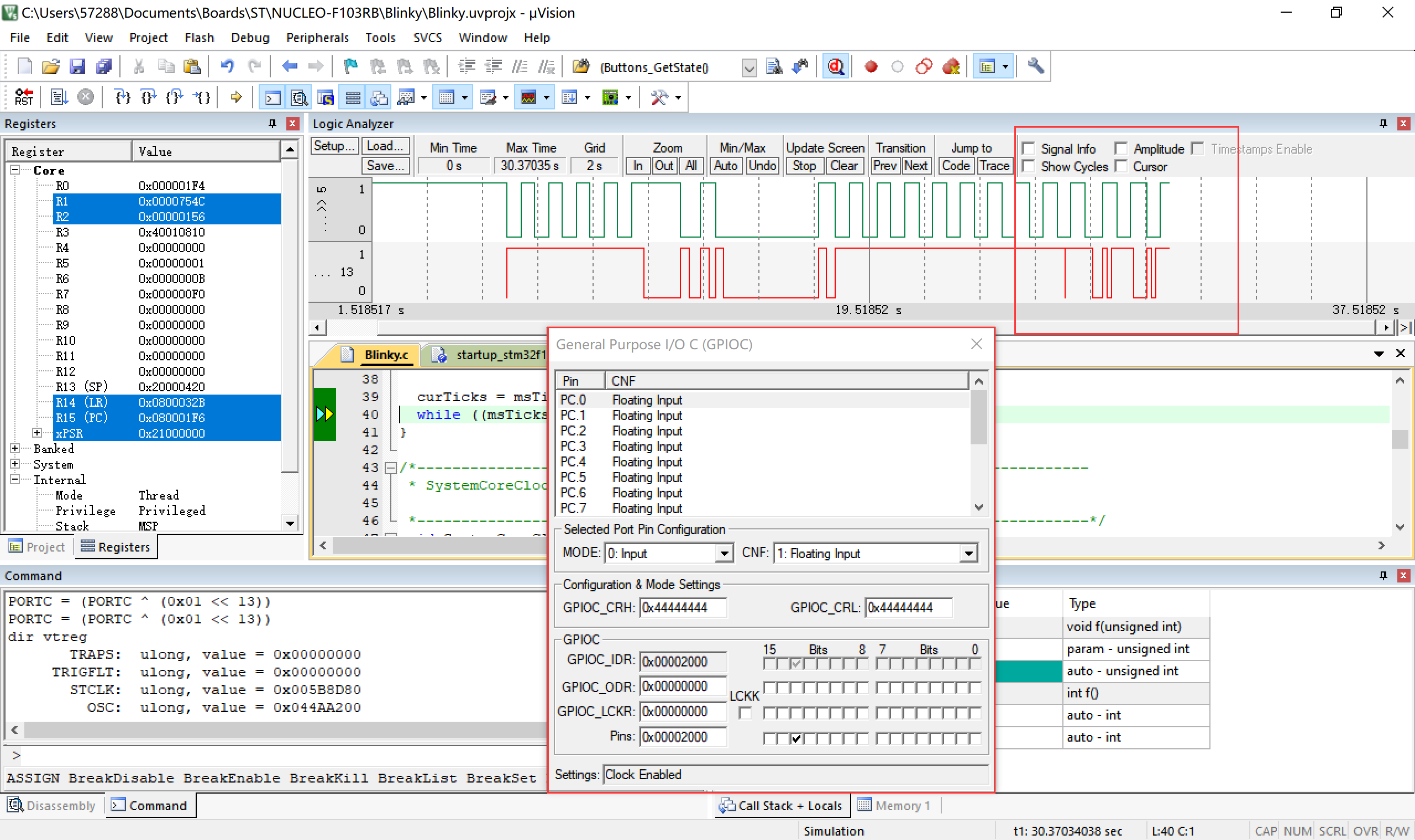


从上图可知，PORTC.13 的改变导致了PORTA.5 的变化。

下面输入命令 dir vtreg可以看到PORTC的值发生了改变，如下图：



8.还可以通过，在 GPIOC 的 pins 项下改变 bit 13 的值（通过是否勾选来实现），也能实现上述效果，如下图：



**可以发现，当bit13被勾选时，LED会没5秒跳转一次，若取消勾选，则不再发生跳转。**

# 实验方案与实现

## 软件结构

Binky示例代码的结构

## 源代码

1. int main (void) {
2. int32\_t max\_num = LED\_GetCount();
3. int32\_t num = 0;
4. SystemCoreClockConfigure();                             *// configure HSI as System Clock*
5. SystemCoreClockUpdate();
6. LED\_Initialize();
7. Buttons\_Initialize();
8. SER\_Initialize();
9. SysTick\_Config(SystemCoreClock / 1000);                  *// SysTick 1 msec interrupts*
10. for (;;) {
11. LED\_On(num);                                                           *// Turn specified LED on*
12. Delay(500);                                               *// Wait 500ms*
13. while (Buttons\_GetState() & (1 << 0));                  *// Wait while holding USER button*
14. LED\_Off(num);                                                       *// Turn specified LED off*
15. Delay(500);                                               *// Wait 500ms*
16. while (Buttons\_GetState() & (1 << 0));                 *// Wait while holding USER button*
17. num++;                                                 *// Change LED number*
18. if (num >= max\_num) {
19. num = 0;                                             *// Restart with first LED*
20. }
21. printf ("Hello World\n\r");
22. }
23. }
24. uint32\_t Buttons\_GetState (void) {
25. uint32\_t val = 0;
26. if ((GPIOC->IDR & (1ul << 13)) == 0) {
27. */\* USER button \*/*
28. val |= USER;
29. }
30. return (val);
31. }

# 实验结果与分析

截图如实验步骤所示；

结果：**可以发现，当bit13被勾选时，LED会没5秒跳转一次，若取消勾选，则不再发生跳转。同时熟悉了软件的使用，为后续学习打好了基础。**

# 实验总结

本次实验中，较难的地方是安装过程中出现的各种问题以及软件的使用方法。

但只要认真对待，搜集资料，就能顺利解决。