**实验报告**

PB21061326 陈昕颜

**实验一**

**1.观察变量存放格式**

|  |  |  |  |
| --- | --- | --- | --- |
| ui\_tmp | 255 | R5 | 0xFF |
| ui\_a | 1 | R2 | 0x01 |
| ui\_b | 2 | R3 | 0x02 |
| ui\_c | 0xFF | R4 | 0xFF |

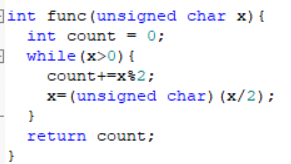
PC单步执行加2，执行前:0x08000432，执行后:0x08000438

执行后PSR寄存器的标志位 Negtive变化:从0变成1

i\_tmp的-1的补码表示：0xFFFFFFFF；-2的补码表示：0xFFFFFFFE；

s16\_tmp的-1的补码表示：0xFFFF,-2的补码表示：0xFFFE。

**2.统计unsigned char类型数据中二进制1的个数**

PD6I3U@0)JW{`OYA[)9R085

1. **使用CMSIS-CORE函数实现底层操作并记录读取内容**

|  |  |  |
| --- | --- | --- |
| steps | ui\_tmp | Register |
| ui\_tmp = \_\_get\_FAULTMASK(); | 0x00000000 | FAULTMASK 0 |
| ui\_tmp = \_\_get\_BASEPRI(); | 0x00000000 | BASEPRI 0x00 |
| ui\_tmp = \_\_get\_PRIMASK(); | 0x00000000 | PRIMASK 0 |
| ui\_tmp = \_\_get\_CONTROL(); | 0x00000004 | CONTROL 0x04 |
| ui\_tmp = \_\_get\_xPSR(); | 0x60000000 | xPSR 0x61000000 |
| ui\_tmp = \_\_get\_MSP(); | 0x20000670 | MSP 0x20000670 |

**实验二**

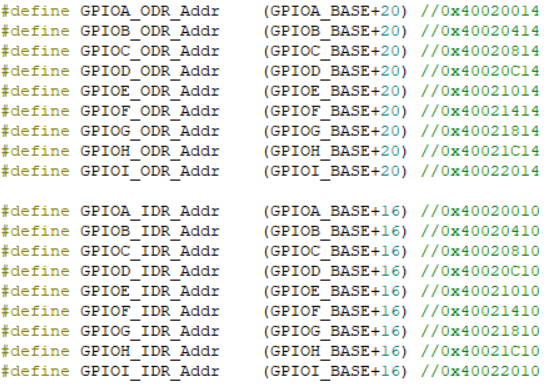
**1.位带地址映射的原理：**

利用位段/位带操作，一次存储器操作可以只访问一个位。M3/M4中，有两个预定义的存储器区域支持这种操作：一个位于SRAM区域的最低1MB(0x2000 0000 ~ 0x200F FFFF)， 另一个位于外设区域的最低1MB(0x4000 0000 ~ 0x400F FFFF) ，这两个区域被称作位段区域，里面的每一个字被映射为一 个位段别名区域，即一个字的32个位被映射到32个位段别名地址（32个字）。使用位段别名地址访问存储器时, 所得到字数据的最低位LSB，对应位段区域中某个字的特定的位。位段特性在M3/M4上是可选的。

**2.实验代码实现：**

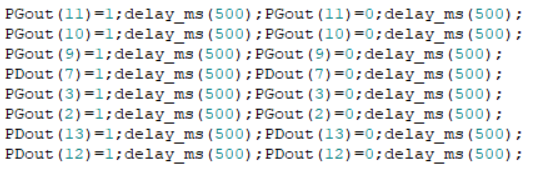


其中GPIOX\_ODR和GPIOX\_IDR地址为：



实现位带操作的函数为：AVY`{0ARL}ZMF}%AJ8I[[BK

**3.使用位带操作控制LED的操作：**

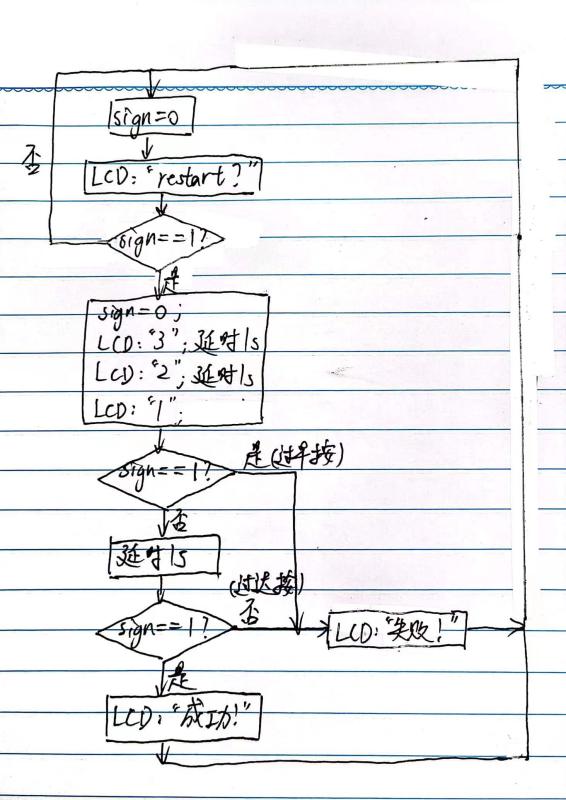


**综合实验**

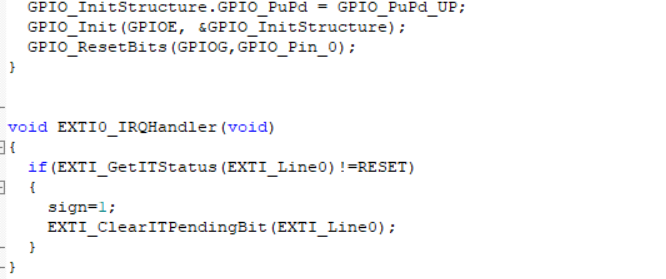
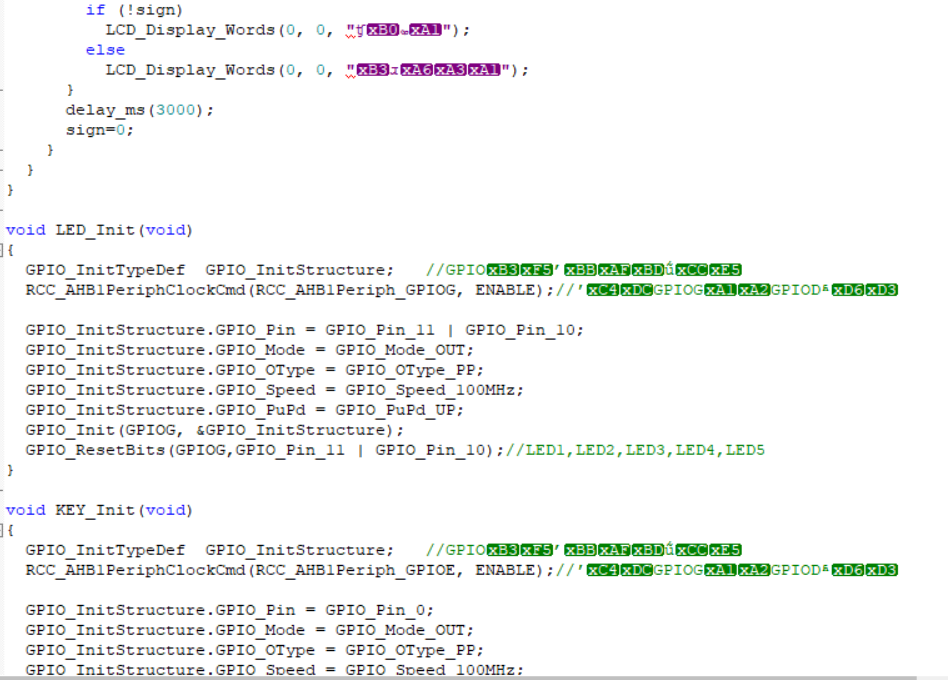
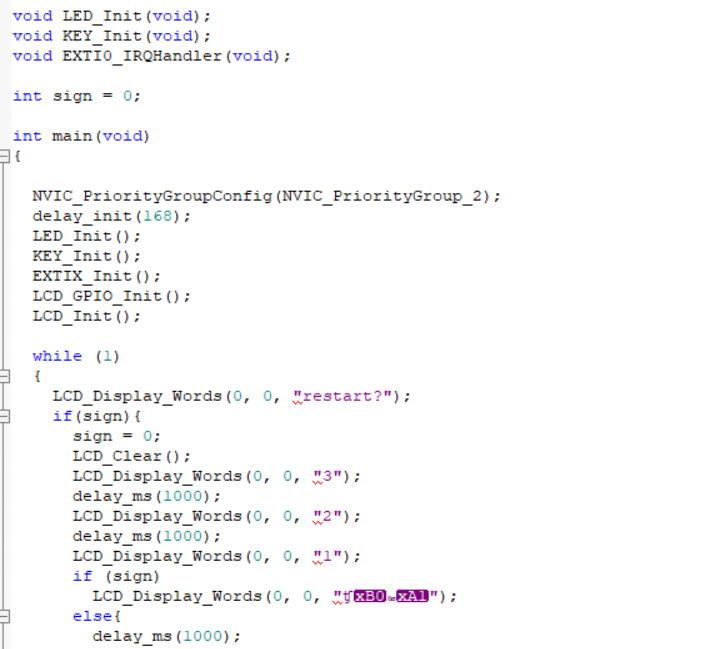
1. **功能描述**

定时抢答器：开始后，LCD上依次显示“3”、“2”、“1”，用户需要刚好在“1”后按下key0，若提早按下，则在LCD上显示“失败！”；若在超时按下，则也在LCD上显示“失败！”；若刚好在显示“1”后按下，则在LCD上显示“成功！”。抢答完成后，等待3s，LCD上将显示“restart?”,此时再次按下key0，则可以重新进行上述定时抢答流程。

1. **功能模块划分**
2. 使用EXTI功能，设置按键控制的外部中断
3. 通过后台控制lcd的显示，实现相关功能
4. 时间由delay\_ms()函数粗略控制
5. **系统结构图**

****

1. **实现功能的核心代码**

****

**5.总结**

本实验利用了EXTI和GPIO功能，基本实现了预期的定时抢答功能。不足之处是，用delay\_ms()函数会导致计时不够准确，可以将其改进为使用定时器实现1s的计时，并且将外部中断的优先级设置为高于定时器中断的优先级，这样将得到更准确的结果。