

## 嵌入式实验

姓名	学号	班级	电话	邮箱
曹广杰	15352015	1501	13727022190	<a href="mailto:1553118845@qq.com">1553118845@qq.com</a>

### 第17周

#### Question1

What is a thread?

线程是一个正在运行的程序，或者是正在运行的程序的一部分。

#### Question2

What is the main thread? What are interrupt threads?

主线程：是正在运行的主程序；

中断线程：是ISR正在运行的程序；

通常在嵌入式系统中，主线程会在一开始被初始化一次，此后就在无限循环中执行，而对于中断，每一次中断被触发，就会有一个新的线程被创建、运行和销毁。

中断服务程序（Interrupt Service Routines）又称ISR：

中断服务程序是嵌入式系统的重要组成部分，所以在开发板的使用中，开发商需要为之提供拓展，例如关键词 `__interrupt`，就是使用中断程序的标志。

中断程序：

- 不能返回一个值

对于裸奔的系统：裸奔即没有操作系统的控制调度，硬件中断响应程序的插入和运行是随机的，此时有返回值也没有接收方，不知道应该返回给谁。

- 不能传递参数

对于裸奔的系统：由于响应程序的插入和运行是随机的，所以此时的调用者是未知的，因此使用参数传递也是难以实现的。

- 尽量不要使用浮点运算

ISR应该是短而且有效的，使用浮点运算则延长了运算的时间。

- 不应该使用 `printf` 函数

不应该有重入和性能上的问题（？）

而对于具有操作系统的情况下，操作系统由于具有调度安排机制，因而具有中断的出入口，至此，中断线程就可以有返回值并且可以传递参数。

#### Question3

What are the five steps that occur automatically (in hardware) as the context switches from the main thread to an interrupt thread?

从当前的主线程到中断线程的跳转有5步：

1. 完成当前的指令；

2. 寄存器入栈

将寄存器（PSW, PC, LR, R12, R3, R2, R1, R0）压入栈；

3. 设置LR寄存器为 0xFFFFFFFF；

4. 设置寄存器 IPSR 为中断数；

5. 设置寄存器 PC 向量

LR 寄存器有两个作用：

1. 储存子程序的返回地址。在返回的时候程序会通过将该寄存器中的地址复制到PC中以实现子程序的跳转；
2. 保存异常发生的地址。异常发生的时候，可以通过地址来处理嵌套中断。

[http://blog.csdn.net/kai\\_wei\\_zhang/article/details/8197006](http://blog.csdn.net/kai_wei_zhang/article/details/8197006)

## Question4

Define the following terms as they relate to interrupts.

- Hardware trigger
- Interrupt enable bit I in the PRIMASK register
- Interrupt enable bit in the NVIC\_EN0\_R register
- Interrupt priority in the NVIC\_SYS\_PRI3\_R or NVIC\_PRI1\_R register
- Interrupt arm bit like bit1 (INTEN) in the NVIC\_ST\_CTRL\_R register
- Interrupt vector

触发器是一个被硬件事件设置的寄存器元件，可以触发中断。计数器就可以被设置。

在寄存器 PRIMASK 中设置为1，可以延迟所有的中断。该寄存器是特殊的寄存器，称为中断屏蔽寄存器，该寄存器只有一位，置为1后可以关闭所有可屏蔽中断的异常，只剩NMI和硬fault。

将寄存器 NVIC\_EN0\_R 中的位设置为1，允许其他针对中断的特殊标志。

中断优先级表明了中断的重要性，0是最高的优先级；7是最低的优先级。

ARM的位允许触发器中断，在 TM4C123 中有上百个触发器单元，但是常用的只有一些，多数的触发器有两个arm比特位：

- 一个在设备的寄存器中
- 一个在 NVIC\_ENX\_R 中

系统时钟只有一个arm比特位，在 NVIC\_ST\_CTRL 寄存器中。

中断向量是指ISR的地址序列，栈中的寄存器（PSW, PC, LR, R12, R3, R2, R1, R0）。

## Question5

What is an interrupt acknowledge? How does the SysTick interrupt get acknowledged and how is SysTick acknowledge different from the other interrupts?

批准中断：通常情况下，是通过ISR中的软件清除触发器位的数据来批准中断；

系统时钟的中断是通过硬件自动获知的，对于系统时钟来说没有清除触发的软件程序，除了它之外，其他的ISR都有用于清理触发位的程序。