# Laboration 3 Task 1 Questions

1. What is the maximum value that the SysTick counter can start counting down from?   
Since the SysTick is a 24-bit counter, its highest starting value is 224 – 1, which is about 16.7 million.  
  
2. What are the addresses of the CTRL, LOAD and VAL registers?

|  |  |
| --- | --- |
| **Registers** | **Addresses** |
| CTRL | 0xE000E010 |
| LOAD | 0xE000E014 |
| VAL | 0xE000E018 |

3. To use register names (CTRL, LOAD and VAL) in my assembly code, how to define them? (Hint: EQU)   
EQU - directive gives a symbolic name to a numeric constant, a register-relative value or a PC-relative value.  
Syntax (name EQU expr{, type})  
- Name: is the the symbolic name to assign the value  
- Expr: is a register-relative address, a PC-relative address, an absolute address, or a 32-bit integer constant  
- Type: is optional, (type) can be any of (arm, thumb, code32, code16, data)  
EQU basically gives names to different expressions(registers) which means u can give them the name of CTRL, LOAD, VAL etc.  
  
4. What is the value to be written in the LOAD register to activate the SysTick interrupt every 2 ms, in case the Clock Source is 12MHz?   
LOAD Value = (Clock Source(12MHz) \* Period(2ms)) – 1  
Which means the LOAD Value is going to be 23999.  
  
5. What are the clock sources that you choose from to configure the SysTick?   
Clock sources  
MCK or MCK8  
  
6. Write down an assembly program that enables SysTick interrupt with a period of 5 ms.  
See program

# Laboration 3 Task 2 Questions

1. What happens to the stack in the event of SysTick interrupt?  
  
2. What is saved on the stack in the event of SysTick interrupt?   
3. What is in the LR register when the SysTick interrupt handler executes, and what does it mean?   
4. What instruction causes the SysTick interrupt to end, i.e. how do you return to the main program? 5. Where is the return address stored?   
6. What are the shared resources between the main program and the SysTick Handler?   
7. What is the maximum value that the Binary counter of the main program reaches? Why?   
8. Do you recognise any sensitive code section in this program? Where is it?   
9. If you recognise a sensitive code section in your program, How can you protect it? (Hint: Do not use PUSH and POP)