**EENG 260 (Microcontrollers)**

**Home Work 2**

**Problem 1:** Logical Operations

In class we said a divide by 2 operation can be achieved by a logical shift left instruction and also said a multiply by 2 operation can be achieved by an arithmetic shift right. Without worrying about the contents of the registers write out the mathematical expressions of R0 after each of the two instructions:

a) ADD R0, R1, LSL #4

b) ADD R0, R1, R2, ASR #4

Answer:

1. R0 = R1/24
2. R0 = (R1+R2)\*24

**Problem 2:** GPIO Memory Map

From the class notes titled “Lect\_On\_Ports\_Rev1” or chapter 10 of Tiva C manual it is said that **Port A** has the address range: 0x4000.4000 – 0x4000.4FFF.

1. What components are addressable in this address range?
   1. 36 GPIO registers are addressable in the address range of each port
2. Provide addresses, names and functions for some of these components
   1. GPIODATA – 0x4000.4000-0x4000.43FF – address range for data to be stored for ports marked as input or output for use by other hardware and software functions
   2. GPIODIR – 0x4000.4400-0x4000.4403 – register for configuring the data direction of the port to be either input or output for that port’s various pins
   3. GPIODEN – 0x4000.451C-0x4000.451F – register for enabling digital I/O for the various pins on a port. That is, to enable executing GPIO functions on the pins of that port

**Problem 3:** The stack (frame and pointer)

1. Provide a stack frame for a program that saves its contents from memory location 0x2000.7FFC, assume 5 words are saved.

|  |
| --- |
| 0x2000.807C – 0x2000.809B |
| 0x2000.805C – 0x2000.807B |
| 0x2000.803C – 0x2000.805B |
| 0x2000.801C – 0x2000.803B |
| 0x2000.7FFC – 0x2000.801B |

1. Where does the return address of a subroutine get stored i.e., when a subroutine completes how does the program know where to start?
   1. The current address before the subroutine starts gets pushed onto the stack and popped once the subroutine finishes.
2. What is a stack overflow?
   1. A stack overflow is when the stack runs out of memory when trying to make a stack call
3. What is a stack underflow?
   1. A stack underflow happens when data is attempted to be taken from the stack when it is empty

**Problem 4:** Exceptions and Interrupts

1. Please state the difference between an exception and an interrupt
   1. Exceptions typically track software related issues, such as overflows and out of bounds calls, while interrupts are usually hardware related and outside the cpu, and often aren’t caused by the program itself
2. What is the purpose of a vector table?
   1. Vector tables are set by the manufacturer to align certain priority levels with certain peripherals to be able to properly handle various hardware
3. What is the purpose of an interrupt service routine?
   1. ISRs are used to allow some events and processes to take priority over others, putting lower priority processes on hold until said ISRs are complete.
4. Interrupts IRQ0 and IRQ1 have the same priority and both happen to occur at the same time. Which of the two interrupts will be serviced first?
   1. The interrupt at the earlier memory location will be serviced first
5. Provide the steps taken by a processor to enter into an exception and how it returns from exception

**Problem 5:** GPIODATA Register

You have the following assembly instructions in your assembly code:

MOVW R1, #0x4000 ;load lower half of reg R1 with 0x4000

MOVT R1, #0x4000 ;load upper half of reg R1 with 0x4000

LDRB R0, #0x05 ;load value 5 into R0

STRB R0, [R1] ;store content of R) at memory location

pointed to by R1

After these four instructions execute:

1. What port is this program manipulating?
   1. Port A
2. What register of this specific port is being manipulated?
   1. GPIODATA
3. How are this port’s pins configured?
   1. GPIODEN is set to 1 for pin 0, GPIODIR is untouched as a default of 0 allows input
4. Now that the port’s pins are set up what register would be used to Read date from or write data to the port?
   1. GPIODIR would need to be set to 1 to allow read, but the register GPIODATA would be used for reading and writing, along with the correct offset for reading/writing the pins’ data.
5. Which pins of the port will be written to and which pins will be read from?
   1. That depends on what values are set in GPIODIR for allowing input/output, but up to 8 pins can be used for a combination of the two