In Class Assignment 2

Name: ____

UMBC ID: ____

Academic Integrity Statement:

"Integrity of scholarship is essential for an academic community. The University expects that students will honor this. By signing this, you state that all work for this exam is performed by the individual to whom it is assigned, without unauthorized aid of any kind."

Signature: ____

Directions: Each section will have one theoretical question and one application question over a topic covered in class. Each question will be worth 10% of the total points allocated for the assignment. Answer all questions thoroughly and thoughtfully, showing all work and explanations so that partial grades may be assigned where a complete answer is lacking.

Section 1: Interrupts

Theoretical Question:

Define the following terms

Interrupt

Signal telling the processor to stop its currently running code and execute a different section of code corresponding to the signal

Critical Section

The section of code that must complete without being interrupted

ISR

Interrupt Service Routine – The code that is run upon an interrupt

Application Question:

Assume you are developing a navigational system which has 2 external interrupt sources: a magnetometer which produces readings at 4 millisecond intervals and a GPS sensor which produces readings at 50 millisecond intervals. Further, assume there is an ISR that reads the magnetometer results over SPI and consumes 1 millisecond and another ISR which reads the GPS results over SPI consuming 2 milliseconds and processes the results consuming another 5 milliseconds. Lastly, assume that you must read the result of a sensor before the result is overwritten by the next packet. Explain a strategy for deterministically reading all values from both sensors. You may change the ISR in some way, or discuss other options for creating this operation.

You should use nested interrupts giving higher priority to the magnetometer so that it may interrupt the GPS ISR, read a packet, and give control back to the GPS so that you never miss a magnetometer reading. You could optionally move the processing of the GPS results to the main loop.

Section 2: Timers and Counters

Theoretical Question:

Explain the difference between a timer and a counter:

Timer counts clock transitions, counters count arbitrary signal transition

Explain one way you can extend a hardware timer without losing precision:

Use the overflow bit as a defacto n+1th bit. Optionally, use the overflow flag to toggle a software counter.

Application Question:

Assume you have an 8-bit hardware timer based on a system clock of 1MHz. You have available the following prescalers: 1, 8, 64, 256. Lastly, assume that your timer is set up to toggle an output pin when the counter reaches a value in the output compare register (CTC mode for AVR). Define the prescaler and output compare register in order to achieve a timer that oscillates close to 2,550 hz.

1,000,000/2550 = 392

392 > 255 -- So…

392/8 = 49 – Therefore, 8 – Prescaler, 49- Count register

Section 3: Analog to Digital Converters

Theoretical Question:

Explain the successive approximating algorithm that is used in many Analog to Digital Converters:

Binary search tree. Compare against half of the voltage and keep a 1 if the(measured) voltage is higher, 0 if the voltage is lower. Proceed through the n-bits until the generated voltage is between two values. (Error upperbounded by 1 LSB precision)

Application Question:

How many clock cycles would a 10-bit ADC take to implement the successive approximation algorithm?

Binary Search Tree, therefore N clock cycles. The clock cycle length is dependent upon the resolution(precision) of the LSB.

Given an ADC which has a 0V ground reference and a 3V top reference, what is the lowest number of bits you need to quantize an ADC with precision of 0.02V or better?

3V-0V/(n^2) = 0.02V

3 / 0.02 = n^2

150 = n^2

Log2(150) = n

n ~= 7.22

Because we need a precision of at least 0.02V, we should choose the ceiling of that value to find the number of bits. Therefore we need at least 8 bits.

Section 4: Real Time Operating Systems

Theoretical Question:

Describe the difference between thread safe and reentrant code:

Thread safe code is safe to be run alongside other code. It locks globals and resources it needs so that if its preempted it still maintains access to those components. Reentrant code is safe to run alongside itself(i.e. the same function being called in multiple places). This is done by locking globals, static variables, and other resources, preventing preemption while holding a lock. (the static variables and preventing preemption especially is how code can be thread-safe but not reentrant).

What is a Task Control Block, and what is it useful for?

A structure which minimally contains the information about a task, including a pointer to the function the task should run. Useful for RTOS implementation so that queues can be implemented properly.

Application Question:

Discuss a strategy for making code thread safe:

Mutexes

Discuss a strategy for making code reentrant:

Local variables, mutexes, avoiding preemption

Section 5: Tasks

Theoretical Question:

What is “Priority inversion?” Give an example of a situation where this could happen:

When a task is allowed to run to completion before a task of higher priority because a lower priority task has blocked the higher priority task and been preempted by the priority inverting task.

C runs and locks R1.

A preempts C and requests R1, gives control to C to release R1.

B Preempts C and runs to completion.

What is a Ring Buffer? What pieces are needed to build the structure?

A circular structure used to store data continuously. You need an array (or any block of memory) and two variables which point to the head and the tail of the data respectively. You should also take care of state variables for when the buffer is empty or full.(prevent overrun or underrun)

Application Question:

Choose one of the time-shared scheduling algorithms talked about during lecture (First-Come First-Serve, Shortest Job First, and Round Robin). Given a system which is based off of synchronous tasks, explain an advantage and a disadvantage of using the chosen scheduling algorithm.

Round Robin:

Advantage : No starvation

Disadvantage: High priority tasks may not finish in a guaranteed amount of time