Embedded Real-Time Systems (AME 3623)
Homework 3

April 24, 2011

This homework assignment is due on Thursday, April 28th at 5:00pm. Your work may be handed in electronically (use the **Homework 3** digital dropbox on D2L) or in hardcopy form (in person or to my office).

This assignment must be done individually: do not share/discuss your answers with others or look at the answers of others.

Question 1

Assume that *student_ID* is the number that corresponds to your student ID number.

1. (2pts) What is *student_ID % 4*? Call this *key1*

   Possible answers are: 0, 1, 2, 3

2. (2pts) What is *student_ID % 5*? Call this *key2*

   Possible answers are: 0, 1, 2, 3, 4

Question 2

Assume the timer/counter equal to your key1.

Assume a prescaler of 1 (if key2 == 0), 8 (key2 == 1), 64 (key2 == 2), 256 (key2 == 3) or 1024 (key2 == 4).

1. (5 pts) What is the frequency of counting of the timer/counter?

2. (5 pts) Assume that we have the overflow interrupt enabled. What is the period between overflow interrupts?

Question 3

Suppose that we want to produce an overflow interrupt frequency of $488 Hz$. Assume that we are using a 16 MHz crystal for our clock.

1. (5 pts) Which timer should we use?

2. (5 pts) Which prescaler should we use?

Question 4

1. (15pts) Suppose that we want a function – called $control()$ – to be executed approximately once every second, and another function – called $sense()$ – to be executed approximately once every 5 minutes. We will use the timer1 overflow interrupt to call both of these. Assume a system clock of $16MHz$. What is the timer1 prescaler configuration and the code for the interrupt routine (the code does not need to be syntactically correct)? Also - show the code in your main function that configures the timer.

Question 5

Consider the following code:

```
ISR(TIMER1_OVF_vect) {
    static uint8_t counter = 0;
    static uint8_t phase = 0;

    if(counter == 0) {
        switch(phase) {
            case 0:
                PORTC = PORTC & 0xFC | 1;
                counter = 75;
                phase = 1;
                break;
            case 1:
                PORTC = PORTC & 0xFC | 2;
                counter = 100;
                phase = 2;
                break;
            case 2:
                PORTC = PORTC & 0xFC;
                counter = 25;
                phase = 0;
                break;
        }
    }
    --counter;
};

Somewhere in the main program:

// Initialization
timer1_config(TIMER1_PRE_64);
// Enable the timer interrupt
timer1_enable();
// Enable global interrupts
sei();

DDRC = 0x3;
PORTC = 0;

while(1)
{
}
```

1. (15 pts) Explain in detail what the program does. You are welcome to provide a picture.

Question 6

How much time did you spend on this assignment?