

# Embedded Real-Time Systems (AME 3623)

## Homework 3

April 9, 2010

This homework assignment is due on Tuesday, April 20th 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* % 2? Call this *key1*

2. (2pts) What is *student\_ID* % 5? Call this *key2*

## Question 2

Assume timer/counter 0 (if key1 == 0) or 1 (if key1 == 1).

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 interval between counts of the timer/counter (tocks)?
2. (5 pts) Assume that we have the overflow interrupt enabled. What is the frequency of interrupts?

## Question 3

Suppose we want to produce a regular interrupt every  $262ms$ . Assume that we are using a  $16MHz$  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 we want a function – called *myfunc()* – to be executed once every  $20.97s$ . 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(TIMER0_OVF_vect) {
    static uint8_t counter = 0;
    static uint8_t phase = 0;

    if(counter == 0) {
        switch(phase) {
            case 0:
                PORTC |= 3;
                counter = 100;
                phase = 1;
                break;
            case 1:
                PORTC &= ~1;
                counter = 50;
                phase = 2;
                break;
            case 2:
                PORTC &= ~2;
                counter = 75;
                phase = 0;
                break;
        }
    }
    --counter;
};
```

Somewhere in the main program:

```
// Initialization
timer0_config(TIMER0_PRE_8);
// Enable the timer interrupt
timer0_enable();
// Enable global interrupts
sei();

DDRC = 0x3;

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?