

CSCE 236 Embedded Systems, Spring 2015

Exam 2

Monday, May 4th, 2015

Instructions: You will have the full class period to complete this test. **Make sure to show your work to ensure you receive partial credit if your final answer is incorrect.** This is a closed book quiz, no computers, textbooks, notes, etc. are allowed.

Unless otherwise specified, assume that questions are referring to the Arduinos/Atmel processors we have been using in class.

Name (2 pts.):

Problem 1. *Hex and bit operations (all references to bit locations are zero referenced). For each bit operation subproblem write a **single line** of C code to achieve the desired result.*

a) (5 pts.). *Set the upper 5 bits in the 8-bit variable `foo` to the upper 5 bits in the 16-bit variable `bar`.*

b) (5 pts.). *Set the lower 5 bits variable `var` to ones.*

c) (5 pts.). *What is the value (in hex) of the 8-bit variable after this operation `var = ((12<<6) | (3<<6)) + 7`?*

d) (5 pts.). *What is the value of `((0x236>>3) - 3)` in hex?*

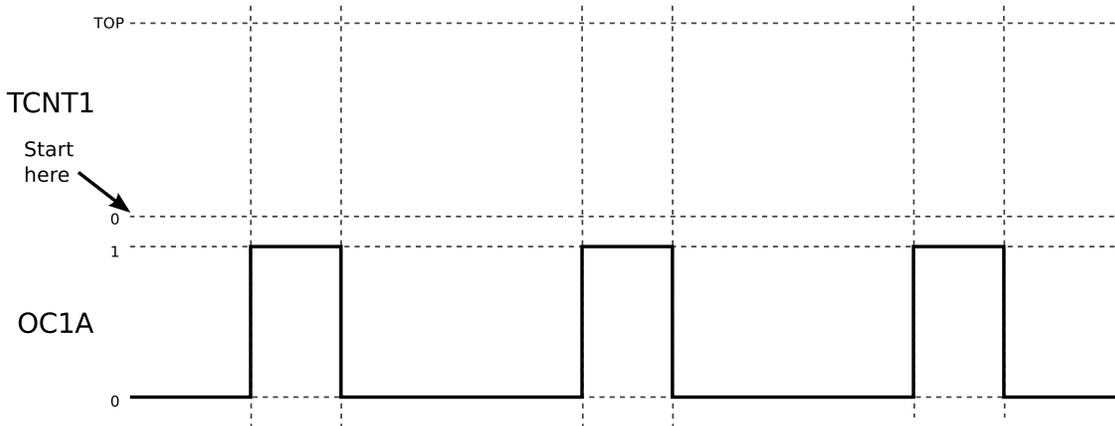
Problem 2. Timer/PWM

a) (5 pts.). Given the following PWM/Timer configuration for a 10MHz ATmega processor, what are the values for ICR1 and OCR1A to generate an interrupt at 10KHz? The comments should give you all the information you need to complete this problem. You can leave the answer as a fraction.

```
void setup(){
  //Set to Fast PWM Mode with ICR1 as top
  TCCR1A = (1<<WGM11);
  TCCR1B = (1<<WGM13) | (1<<WGM12);
  //Set clock to clk/1
  TCCR1B |= (1 << CS10);
  //Enable interrupt on match with OCR1A
  TIMSK1 = (1 << OCIE1A);
  //Set the value of the top
  ICR1 = //FILL IN
  //Set the value of the output compare register
  OCR1A = //FILL IN
}

SIGNAL(TIMER1_COMPA_vect){
  //Interrupt handler code goes here
}
```

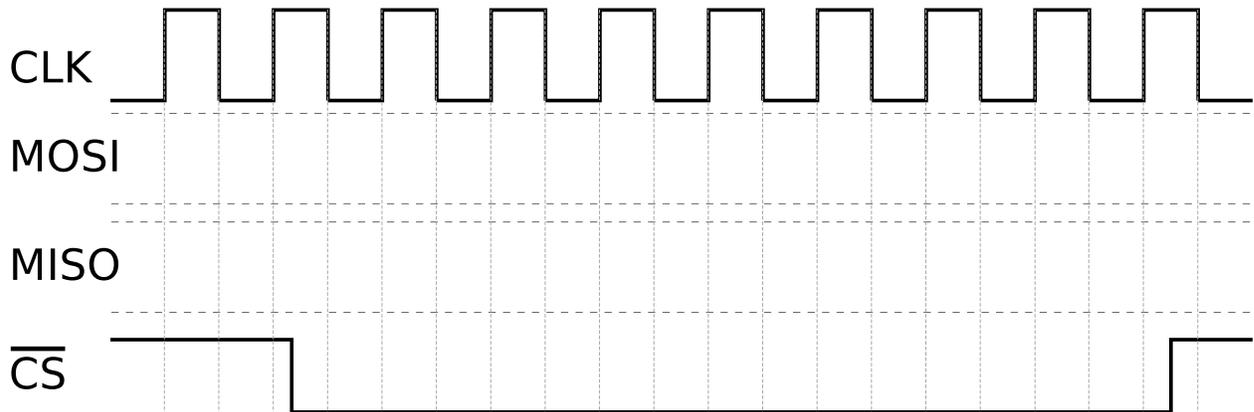
b) (5 pts.). In the following figure, draw the clock, TCNT1, that will generate the lower square wave when it is configured in phase and frequency correct PWM mode. Also draw the correct line in the upper part of the diagram to indicate the location of OCR1A. Assume it is configured as set on up-count, clear on down-count.



c) (5pts). If a PWM signal was 200Hz in Fast PWM Mode, what would the frequency be if you switched to Phase Correct Mmode without changing other settings?

Problem 3. Communication

a) (5 pts.). In the following figure, draw the proper signal for sending the from the master to the slave the value 0xAAC on the rising edge and from the slave to the master the value 0x23 on the falling edge.



b) (5pts). What is one advantage and one disadvantage of differential signaling when communicating (e.g. like RS435)?

c) (5 pts.). In I²C, describe how arbitration works when there are two masters. What is unique at the physical layer that enables multiple masters with I²C?

d) (5 pts.). How many bytes could be sent per second with a baud rate of 57600 when configured as (you can leave the answer as a fraction):

```
//Use double speed here
UCSROA = (1<<U2X0);
//Enable tx and rx
UCSR0B = (1<<RXEN0)|(1<<TXEN0);
//Set frame format to 8 data, 2 stop bits, even parity
UCSR0C = (3<<UCSZ00) | (1<<USBS0) | (1<<UPM01);
```

Problem 4. *Interrupt Example Code.* For this problem, refer to the following code. This code monitors two of the external interrupts connected to two different buttons. The goal of the code is to turn on one of the LEDs if the counter is less than zero, the other when the counter is greater than zero, and to have both off when counter is zero. Note that during the SIGNALs interrupts are disabled.

```
00: int32_t counter = 0;
01:
02: SIGNAL(INT0_vect){
03:   counter++
04: }
05:
06: SIGNAL(INT1_vect){
07:   counter--;
08: }
09:
10: void loop(){
11:   if(counter < 0){
12:     //Red on
13:     digitalWrite(LED_RED,HIGH);
14:   }else if(counter > 0){
15:     //Green on
16:     digitalWrite(LED_GREEN,HIGH);
17:   }else if(counter == 0){
18:     //Both off
19:     digitalWrite(LED_RED,LOW);
20:     digitalWrite(LED_GREEN,LOW);
21:   }
22: }
```

a) (5 pts.). *The above code does not work properly. Describe how this code could end up with both LEDs on (at least until the buttons are pressed again). Refer to line numbers to help in your explanation.*

b) (5 pts.). *How can you fix this code by modifying the main loop?*

c) (5pts.). *Describe one benefit and one drawback of using interrupts in your code.*

d) (5 pts.). *What are the steps that occur to switch from executing the main code to executing interrupt handler code when an interrupt occurs and then back to program execution?*

Problem 5. *Analog to Digital Converters*

a) (10 pts.). *Recall that the Atmel ADC requires an input frequency between 50kHz and 200kHz for maximum reading. Given a 12MHz clock speed (otherwise assume the processor is configured as on our Arduino), write the code to configure the registers properly for maximum resolution and speed and read the ADC value on pin 2 (by reading registers).*

b) (5 pts.). *If you get a reading of 0x1F from a 12-bit ADC and a 5.0V reference voltage, what will the corresponding voltage on the pin be? You can leave your answer as a fraction.*

Problem 6. *Embedded Operating Systems*

a) (5 pts.). *What is the difference between a fully preemptive scheduler and a cooperative multi-tasking scheduler?*