CSCE 236 Embedded Systems, Spring 2019
Test 1

February 21st, 2019

Instructions: Write your name on <u>every</u> sheet. You will have 75 minutes to complete this test. **Make sure to show your work to ensure you receive partial credit if your final answer is incorrect.** This is an open book test, textbooks, notes, etc. are allowed, except digital media such as phones, tablets, computers, etc. This test is out of 100 points.

Unless otherwise specified, assume that questions refer to Arduino/Atmel processor we have been using in class. **IMPORTANT:** Write to the point and be brief.

Name:

1 Multiple Choice (Circle <u>all</u> answers that apply).

(i) (2 pts) How many 8-bit hardware timers does atmega328P provide?

(a) 0

(b) 2

(c) 3

(d) 1

(ii) (2 pts) Which of the following is non-volatile memory?

(a) SRAM

(b) DRAM

(c) Flash memory

(d) EEPROM

(iii) (3 pts) How would you debounce a button that "bounces" so that it does not bounce?

(a) Add a pullup resistor

(b) Read the pin in analog mode

(c) Add a delay after your digital read

(iv) (3 pts) An Instruction Set Architecture defines which of the following for a processor:

(a) The memory architecture

(b) The number of available instructions

(c) The number of clock cycles an instruction will take

(d) The amount of memory available for instructions

2   Bit Operations

All bit locations are zero referenced.

(i) (4 pts) What is the value of $((4 << 2)|(3 << 2))$ in hex?

(i) (5 pts) What is the value of $((4 << 2)|(3 << 2))\&(7 << 1)$ in hex?

(i) (5 pts) Set bit 4 in variable **var**. Give your answer using C in one line.

(i) (5 pts) Clear bit 6 in variable **var**. Give your answer using C in one line.

(i) (5 pts) Set bits 1-3 (inclusive) while also clearing bits 5-7 (inclusive) in variable **var**. Give your answer using C in one line.

(i) (5 pts) Set bits 1-3 (inclusive) in variable **var** to the lowest three bits in variable **data**. Give your answer using C in one line.

3 Memory

(i) (5 pts) What is the best practice in sharing a variable between different sections of the code like functions? What should be avoided while sharing the variables between sections of the code?

(iii) (3 pts) What *memory architecture* does Arduino atmega328 (the one used in our course) use? Think of how instructions and data are organized in the memory.

(iii) (5 pts) Consider the following code segment. What is the value of globalVar and var at the end?

```
uint8_t data[] = {0,1,2,3,4,5,6,7};
uint16_t globalVar = 0x236;
uint8_t *globalPtr;

int8_t main(void){
uint8_t var = 0x9;
uint8_t *p1 = data + 2;

p1[1] = 0xc;
*(p1+3) = data[0];
var += globalVar;
globalVar = (p1++)[1];
data[3] = p1[2];
return 0;
}
```

4   Timer/Counter.

(ii) (3 pts) What is the value of MAX (*i.e.* maximum value that can be counted) for Timer0 and Timer2 in Atmel, respectively?

(iv) (5 pts) Assume a normal mode of operation where the timers count to MAX and then roll over (reset to zero). How many seconds will it take for Timer0 and Timer1 before they will roll over.

(ii) (5 pts) Assume a clock speed of 16MHz, what is the prescaler and OCR1A register value that needs to be set to generate interrupt at a period of 0.25 seconds **using Timer1**. Show your work.

(iv) (5 pts) When does the Input Capture Register value `ICR1` get incremented by 1? What pin on the arduino board is the input capture pin?

(v) (5 pts) Is it possible to configure **Timer0** to overflow every 1 second? If so, which prescaler would you chose? (Note: There is no need to consider OCR1A/B registers for this question. Only a simple division may be required.)

(vi) (5 pts)Before you implemented the debounce logic, what mode did you configure your Timer1 in lab 2 to count the number of button presses?

5   PinMapping schematic

(ii) (5 pts) For this problem refer to the attached PinMapping schematic for the Arduino. What port and pin number does the arduino pin labeled 9, 10, 5 and 4 on the Arduino correspond to on the Atmel processor?

6   Digital I/O

(ii) (5 pts) Write C code to configure the arduino pin **digital pin 5** as an output and then set it to high (recall the I/O registers have general names **DDRx**, **PORTx,PINx**).

(ii) (5 pts) Write C code to test if **digital pin 7** on the Arduino is high by reading the Atmel registers directly.

(ii) (5 pts) Write C code to set the arduino pin **digital pin 8** to input with the internal pullup resistor enabled.