(2 pts) Name:

AME 3623: Embedded Real-Time Systems: Final Exam

May 10, 2006

- This examination booklet has 15 pages.
- Do not forget to write your name at the top of the page and to sign your name below.
- The exam is closed book, closed notes, and closed electronic device. The exception is that you may have one page of your own notes.
- The exam is worth a total of 200 points (and 20% of your final grade).
- Explain your answers clearly and be concise. Do not write long essays (even if there is a lot of open space on the page). A question worth 5 points is only worth an answer that is at most 1.5 sentences.
- You have 2 hours to complete the exam. Be a smart test taker: if you get stuck on one problem go on to the next. Don't waste your time giving details that the question does not request. Points will be taken off for answers containing extraneous information.
- Show your work. Partial credit is possible, but only if you show intermediate steps.

Problem	Topic	Max	Grade
0	Name	2	
1	Interrupts and I/O	65	
2	Finite State Machines I	20	
3	Finite State Machines II	25	
4	Analog Processing	25	
5	Microprocessor Design	35	
6	Logic	30	
Total		202	

On my honor, I affirm that I have neither given nor received inappropriate aid in the completion of this exam.

Signature: _____

Date: _____

1. **Interrupts and I/O** (65 pts)

(a) (15 pts) Below is an interrupt service routine that is supposed to produce a signal of some frequency on port D, pin 2 (counting from 0). However, there exist several bugs (errors). Make the necessary changes to this code to remove these bugs. The "half period" is specified by the variable **duration** (i.e., the signal changes state every **duration** counts). You may assume that the I/O hardware has been initialized correctly.

```
volatile uint16_t counter;
volatile uint16_t duration;   // Half period

ISR(TIMER0_OVF_vect) {

  ++counter;

  if(counter == duration) {

     PORTD |= 0x10;

  }
}
```

(b) (5 pts) Are there any *shared data problems* with the above code (in association with a main program)? Explain.

(c) (10 pts) Assume that the code above has been corrected. Given a prescalar of 256 and $duration = 20$, what is the resulting signal frequency? (set up the fraction, but do not reduce it)

(d) (5 pts) Briefly define *serial communication.*

(e) (5 pts) What is the function of a *start bit* in serial communication (be specific)?

(f) (5 pts) Briefly define *polling.*

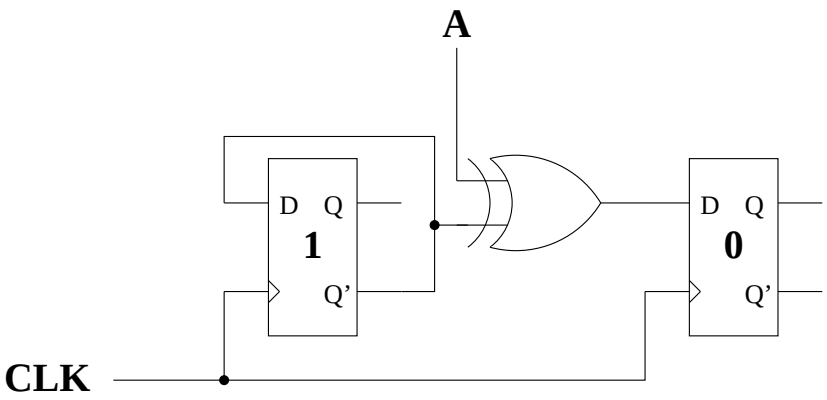
(g) (5 pts) What is one disadvantage of polling?

(h) (10 pts) Define a *buffer*. Briefly explain why it is useful.

(i) (5 pts) Briefly explain why it is a problem to wait inside of an ISR for a digital I/O pin to change state.

2. **Finite State Machines I** (20 pts)

Consider the following sequential logic circuit:



Answer the following questions in the context of describing this circuit in terms of a finite state machine.

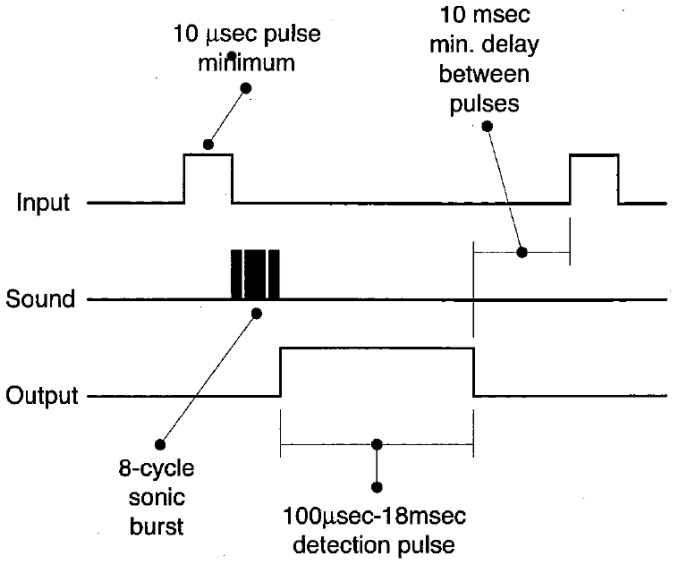
(a) (5 pts) What are the states?

(b) (5 pts) What are the events?

(c) (10 pts) Draw the finite state machine diagram. (hint: for each state and event, you must show what the next state is)

3. **Finite State Machines II** (25 pts)

Consider the specification for the behavior of the sonar unit that we used for projects 3 and 4. Over the next several questions, we are going to develop a finite state machine that is responsible for controlling the sensor and for processing the sensed distance. You may use a counter, but it **should not** be considered part of the state for our purposes (instead, consider it an input). You can assume that the FSM is implemented using an ISR that is called once every $16\mu sec$ and that the counter increments on each execution of the ISR. Also, you can define reasonably named constants without giving explicit values for them (e.g., you don't have to compute the number of counter steps that make up $5ms$).



(a) (5 pts) What are the states? (hint: recall that the FSM is in one state at any given time. You should be able to show on the above timing diagram what state the FSM is in at any given moment)

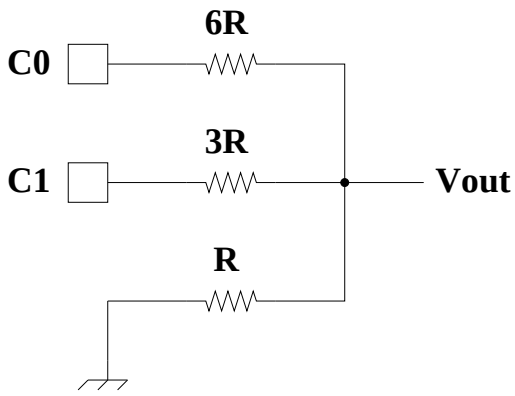
(b) (5 pts) What are the events? (in addition to **NONE**)

(c) (5 pts) What are the actions?

(d) (10 pts) Show the FSM.

4. **Analog Processing** (25 pts)

Consider the following digital-to-analog conversion circuit:



(a) (15 pts) Give the expression for V_{out} in terms of binary digits $C0$ and $C1$. Show your derivation

(b) (10 pts) Given that a program "wants" to generate a voltage $V_{out} = 1V$. What choice of binary values would best approximate this value?

5. **Microprocessor Design** (35 pts)

(a) (10 pts) List two special purpose registers and briefly describe the function of each.

(b) (5 pts) During a read operation from a memory chip, which microprocessor component is responsible for driving the data bus?

(c) (5 pts) (True/False) In the Atmel Mega8, special purpose registers provide the values to the Arithmetic Logical Unit (e.g., the values to be added together). Explain.

(d) (5 pts) Briefly state the function of an *instruction decoder.*

(e) (10 pts) What is the value of variable *baz* at the end of this segment of code (in hexadecimal)?

```
foo = 0x40;

bar = 0xF3;

baz = foo & bar;

baz = baz ^ 0xA5;  // XOR
```

6. **Logic** (30 pts)

Given the following truth table:

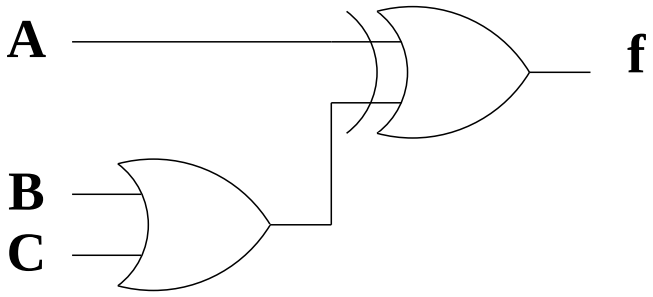
A	B	C	f
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

(a) (5 pts) Give the "minterm" form of the corresponding algebraic expression.

(b) (10 pts) Derive a simplified algebraic description for f. Justify each step (provide the name of the rule that you are using).

(c) (5 pts) Draw the corresponding circuit.

Given the following circuit:



(d) (10 pts) What is the truth table?

A	B	C	f
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	