Embedded Real-Time Systems (AME 3623)
Homework 4 Solutions

April 29, 2007

Question 1

1. (10pts) Briefly explain the need for *buffers* in communication.

   A buffer is used to temporarily store data that is arriving until a program can get a chance to process the data. This allows the program to perform other tasks as the data is arriving. (note that there can also be output buffers: the program places a set of bytes into a buffer to be sent out, but does not have to wait for them to leave).

2. (10pts) Define the "shared data problem."

   The shared data problem can occur when:

   - *two parts of a program can access a common data structure, and*
   - *at least one of these parts (e.g., an ISR) can interrupt the other at any time.*

   The problem occurs because both parts of the program assume that the data structure does not change while it is executing. However, when one can interrupt the other, this assumption no longer holds.

Question 2

1. (15pts) Suppose we want a small segment of code – called *donow*() – to be executed precisely once every $1.664ms$. What is the timer0 prescalar configuration and the (psuedo)code for the interrupt routine (the code does not need to be syntactically correct)?

   We will use a prescalar of 8. This gets us down to an interrupt every 0.128 ms. We then need an interrupt routine with an additional counter that expires at 13. So, we are left with an interrupt interval of: $13 * 256 * 8/16000000 = 1.6640ms$.

```
SIGNAL(SIG_OVERFLOW0) {
    ++counter;
    if(counter == 13) {
     donow();
     counter = 0;
   };
};

Somewhere in the main program:

// Initialize counter
counter = 0;
// Interrupt occurs every (8*256)/16000000 = 128 usec
timer0_config(TIMER0_PRE_8);
// Enable the timer interrupt
timer0_enable();
// Enable global interrupts
sei();
```

2. (15pts) Suppose we want a *donow*() to be called once every 29.3601*sec*. Which timer should we use? What is the prescalar? What is the software divisor? (there are multiple solutions; select the one that minimizes the interrupt frequency)

Use timer 1

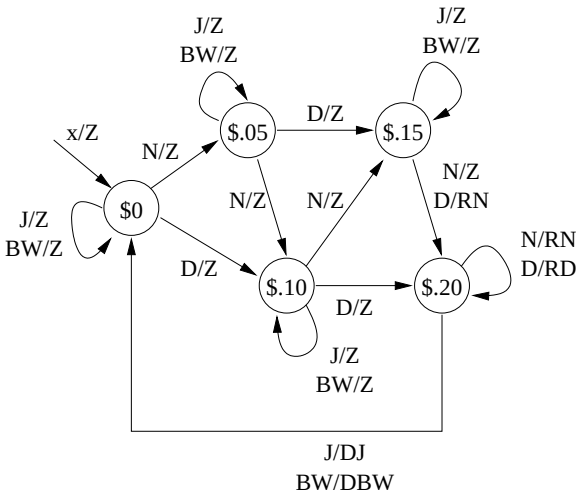
prescalar: 1024

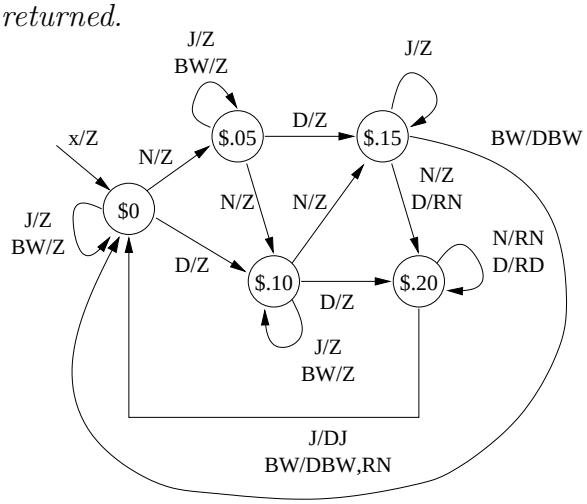
software divisor: 7

```
SIGNAL(SIG_OVERFLOW0) {
    ++counter;
    if(counter == 7) {
      donow();
      counter = 0;
   };
};

Somewhere in the main program:

// Initialize counter
counter = 0;
// Interrupt occurs every
//     (1024*256*256)/16000000 = 4.1943 sec
timer1_config(TIMER1_PRE_1024);
// Enable the timer interrupt
timer1_enable();
// Enable global interrupts
sei();
```

Question 3

(20pts) Below is the FSM for the vending machine that we discussed in class.



Alter this vending machine such that Buzz Water requires only \$.15 in order to buy it. State any additional assumptions that you make.

We will assume that at \$.20*, if Buzz Water is selected, then a nickel will be returned.*