TERRAN BLAKE // ECE 241 // WEDNESDAY 7:30AM

1) Description of the objective of the effort (to creating a software serial port), the design process (state machine and timer) and code in general.

-The objective of the effort to create a software serial port, was to help us better understand the inner workings of serial and bluetooth communications, as well as the complexity behind each of these functions.

2) Testing procedure, such as writing a test program the sends out a fixed string over the BlueTooth Modem.

3) And as always properly documented (Commenting and Variable Naming) code.

-I was unable to finish the rest of the program. It should be commented for the parts that I finished. Also, I was missing my Bluetooth Module until the end of the weekend of April 15th, so I came in during Ethan's office hours to work on it as much as I could.

```
#include <TimerOne.h>

#define BUFFERSIZE 8
char circBuffer[BUFFERSIZE];
int count = 0;

enum isrState {Idle, StartBit, DataBit};
isrState currentState;

int Timer = millis(); // This timer is used for the serial prints

void setup() {
 Serial.begin( 9600 );
 pinMode ( 12, OUTPUT ); //pin modes for the reading

 SW_Serial_Initialize( BaudRate, PinNumber );

}

void SW_Serial_Initialize( int BaudRate, int PinNumber ) {
   Timer1.attachInterrupt( SW_Serial_ISR, ( 1000000/ BaudRate ) );

}

void SW_Serial_ISR() {

  switch (currentState) {
    case Idle:
      if ( Is there data? ) {
```

```
      digitalWrite( 12, LOW );
      currentState = StartBit;

    }
    break;

  case StartBit:
    count = 0;
    bitRead(12, bitRead(Hold, count);
    currentState = DataBit;
    break;

  case DataBit:
    if( count == 8 )
      digitalWrite(12, HIGH);
      currentState = Idle;

    else
      //digitalWrite( 12 ) = bitRead(12, count);
      count++;

    break;
 }

}

void SW_Serial_String ( char circBuffer[] ) {
 circBuffer[] = digitalRead( 12 );

}

void loop() {


}
```