Spring 2021 – ECE 487/587 Lab #4 Grading Sheet

Name: _____ CWID: _____

Functionality/description of hardware and code (70 points):

Good Programming Practices (30 points):

1) Comments

2) Indentation

3) Good/meaningful variable names

4) Minimum/wise usage of global variables; unnecessary use of variables

5) Inefficient code

```
/*****************************************************************************
 * lab_4
 * Created by Matt Mason,
 * CWID 11800439
*****************************************************************************/

#include <avr/wdt.h>

// potentiometer pin
#define POT A5

/*****************************************************************************
 * Function:        doConversions
 * Parameters:      none
 * Return value:    none
 * Purpose:         Performs 30 A->D conversions for the POT analog input.
 *                  Outputs the hex value and the conversion time for each,
 *                  as well as the average conversion time at the end.
*****************************************************************************/
void doConversions()
{
    unsigned long totalTime = 0;
    unsigned long startTime, conversionTime;
    for (int i = 0; i < 30; i++)
    {
        // record start time
        unsigned long startTime = micros();
        // do conversion
        int value = analogRead(POT);
        // calculate elapsed time
        unsigned long conversionTime = micros() - startTime;
        // display #, value, and conversion time nicely formatted
        if (i+1 < 10)
            Serial.print(" ");
        Serial.print("#");
        Serial.print(i + 1);
        Serial.print(": digital value = ");
        if (value < 0x010)
            Serial.print("0");
        if (value < 0x100)
            Serial.print("0");
        Serial.print(value, HEX);
        Serial.print(", conversion time = ");
        Serial.print(conversionTime);
        Serial.println(" us");
        // sum all conversion times
        totalTime += conversionTime;
    }
    // calculate average conversion time
    float averageTime = totalTime / 30.0f;
    // display average conversion time
    Serial.print("average conversion time = ");
    Serial.print(averageTime, 2);
    Serial.println(" us");
    // disregard any received input by clearing serial buffer
    while (Serial.read() != -1);
}

void setup()
{
    // setup input pin
    pinMode(POT, INPUT);

    // open serial connection
    Serial.begin(9600);
    Serial.println("\nlab_4 – Board Reset");

    // enable watchdog timer with a 4-second timeout
    wdt_enable(WDTO_4S);
}
```

```
71
72  void loop()
73  {
74      // display prompt
75      Serial.print("Enter 'c' to start a set of conversions: ");
76      // refresh watchdog
77      wdt_reset();
78
79      // loop condition variable
80      bool waiting = true;
81      // user input string
82      String input = "";
83      while (waiting)
84      {
85          // read serial into input string until buffer is empty or newline received
86          char c = 0;
87          while (Serial.available())
88          {
89              c = (char)Serial.read();
90              if (c == '\n')
91                  break;
92              input += c;
93          }
94          // if last character reveived was a newline, user input is ready
95          if (c == '\n')
96              waiting = false;
97      }
98
99      // refresh watchdog and respond to user input
100     wdt_reset();
101     if (input.equals("c"))
102     {
103         // do A->D conversions
104         Serial.println(input);
105         Serial.println("Starting a set of 30 conversions:");
106         doConversions();
107     }
108     else // invalid input
109     {
110         // display error message
111         Serial.println(input);
112         Serial.println("Invalid input!!");
113     }
114 }
```