Course Name: Principals of Electrical Engineering 1

Course Number and Section: 14:332:223:05, Section #5

Experiment: Lab 5

Lab Instructor: Ghazal Rahmanisane

Date Performed: 12/6/24

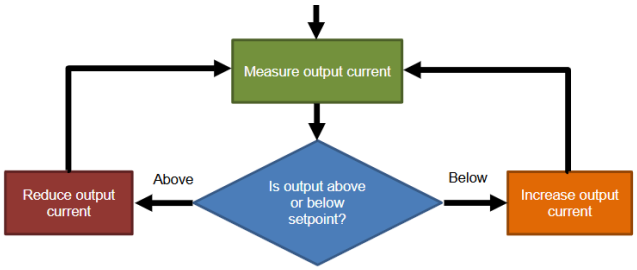
Date Submitted: 12/9/24

Submitted by: Seth Caskey; 224005542

Pre-Lab #5

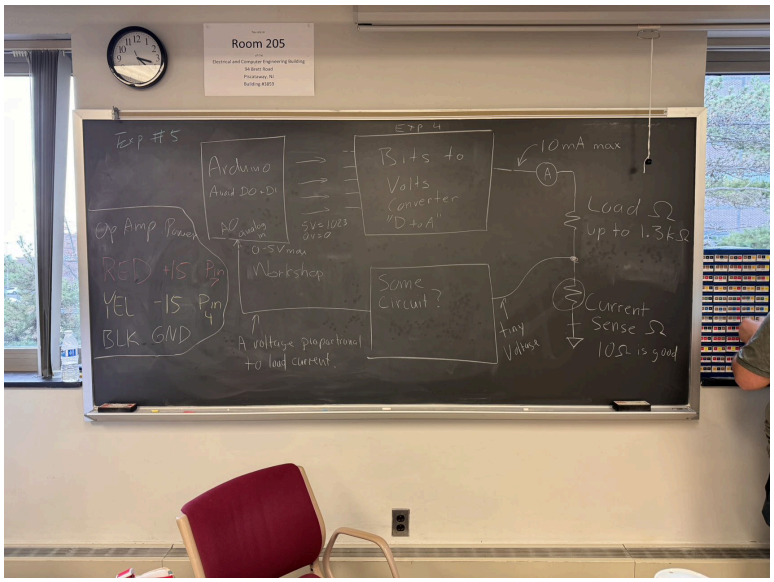
- **Objective**: Create a programmable constant current source using Arduino
  - **Max output current**: 10mA
  - **Setpoint resolution**: 0.5mA
  - **Regulation**: ±0.25mA
  - **Voltage limit**: 15V

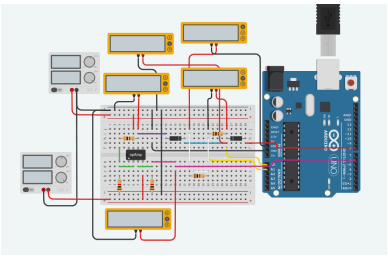
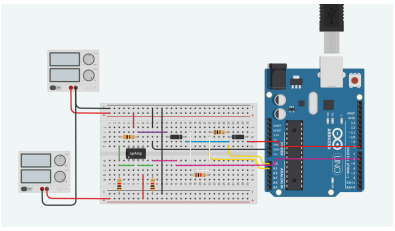
- **How**: Output current will be set using the Arduino IDE serial monitor for the user to input a current setting from 0 to 10mA. The Arduino will then regulate the current level requested by the user.
  - 1) Measure the actual current that is flowing from the output, using the Arduino Analog to Digital (DAC) converters (measure the voltage across a small value resistor of ≤ 1Ω). See (Lab 4)
  - 2) Output a current up to 10mA from the Arduino. Use the digital-to-analog (DAC) converter. See (Lab 4)
  - 3) Regulate the output current using Arduino software

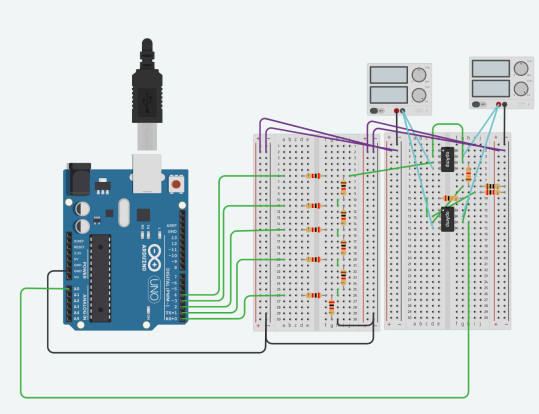


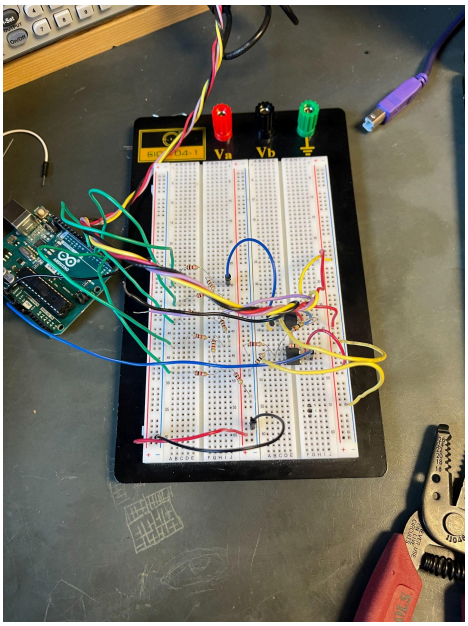
- **Testing**: Provide 2 wires as the output of my circuit (output current). They will be connected to various resistive loads (**Load resistance**: See Post-lab (Lab 4). It is just the resistance values. If you change the load resistance, the current should be the same.) to test the performance of my current source as follows:
  - Setpoint = 0.5mA and Load resistance = 24Ω
  - Setpoint = 0mA and Load resistance = 1.0kΩ
  - Setpoint = 10mA and Load resistance = 1.0kΩ
  - Setpoint = 10mA and Load resistance = 1.3kΩ
  - Setpoint = 10mA and Load resistance = (AN LED)

Post-Lab #5



1. **Final schematic of circuit. (Must be complete enough so that another person could construct the circuit without having to consult external references)**:
    a. Type 1:
        i. 
        ii. 
    b. Type 2:

i. 

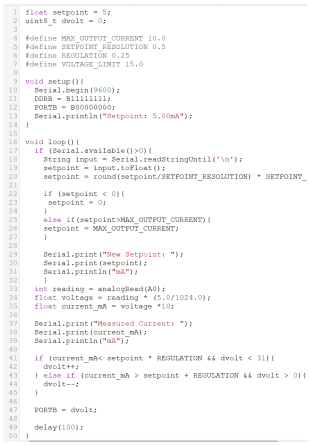
ii. 

2. **Arduino code**:
   a. Type 1:

i.
```
int counter, bitval;
float avgVal, val0, val1, voltagediff, voltage0, voltage1, voltage, current, desv, res;
float desc;
bool calculating = false;
int del = 5; //minimum 3
float bitValueRatio = 0.0535; // adjust based on hardware


void setup() {
  Serial.begin(9600);
  pinMode(6, OUTPUT);
}

void loop() {
  if(calculating == false){
    Serial.println("");
    Serial.print("Current Setpoint? (mA)> ");
    while(Serial.available() == 0) {}
    desc = Serial.parseFloat();
    Serial.println(desc);
    Serial.println("");
    counter = 0;
    calculating = true;
  }
  if(calculating == true){
    if(counter < del){
      digitalWrite(6, HIGH);
      analogWrite(6, 0);
      voltagediff = (voltage0 - voltage1)*1000;
      current = (voltagediff/100);
      res = voltage0/current;
      val0 = analogRead(A0);
      val1 = analogRead(A1);
      voltage0 = val0 * (5.0 / 1023.0);
      voltage1 = val1 * (5.0 / 1023.0);
      voltage = voltage0 - voltage1;
      counter++;
    }
    if(counter >= del){
      calculating = false;
    }

    delay(1000);
  }
}
```

   b. Type 2:

i.



```
float setpoint = 5;
uint8_t dvolt = 0;

#define MAX_OUTPUT_CURRENT 10.0
#define SETPOINT_RESOLUTION 0.5
#define REGULATION 0.25
#define VOLTAGE_LIMIT 15.0

void setup(){
  Serial.begin(9600);
  DDRB = B11111111;
  PORTB = B00000000;
  Serial.println("Setpoint: 5.00mA");
}

void loop(){
  if (Serial.available()>0){
    String input = Serial.readStringUntil('\n');
    setpoint = input.toFloat();
    setpoint = round(setpoint/SETPOINT_RESOLUTION) * SETPOINT_

    if (setpoint < 0){
     setpoint = 0;
    }
    else if(setpoint>MAX_OUTPUT_CURRENT){
    setpoint = MAX_OUTPUT_CURRENT;
    }

    Serial.print("New Setpoint: ");
    Serial.print(setpoint);
    Serial.println("mA");
    }
  int reading = analogRead(A0);
  float voltage = reading * (5.0/1024.0);
  float current_mA = voltage *10;

  Serial.print("Measured Current: ");
  Serial.print(current_mA);
  Serial.println("mA");

  if (current_mA< setpoint * REGULATION && dvolt < 31){
    dvolt++;
  } else if (current_mA > setpoint + REGULATION && dvolt > 0){
    dvolt--;
  }

  PORTB = dvolt;

  delay(100);
}
```

3. **Discuss any challenges you had with your design (Did you have to modify the design, Was there a difficult part, How did you solve the problem(s), Did you simulate anything first, Did the simulation match reality)**:
    a. When trying to make a solution to Lab 5, I found it difficult to understand what was going on and how I would solve it. I wrote out a draft of the problem and ways on how I may solve it. I also looked at previous labs and the images in the Lab 5 directions. I began with the DAC from Lab 4, and used OP-Amps to amplify the voltage difference between the two input signals. Since the voltage is limited by the Arduino, I need to amplify it so I can achieve 10mA of current through a 1.3kΩ. Therefore, I would need a voltage limit of 13V, preferably 15V as in Lab 5 directions. The second OP-Amp is used to amplify the signal so the Arduino can read it.

4. **Discuss any challenges you had with getting your design to work as it should (Did you have wiring problems, Did you have failed components, What was your troubleshooting, What steps did you go through to find any problems)**:
    a. Initially, I had to troubleshoot why the Arduino was not properly regulating the voltage in pin A0. I later discovered I did not have all five bits in my DAC connected properly. I found out about this issue when I simulated my circuit on TinkerCAD and then compared it to the one built on a physical breadboard.

5. **Speculate on what might happen if the load resistor attached to your constant current source varied with time (AKAm What if the load was a potentiometer which was being changed at 100 Hz)**:
    a. If the load resistance varied with time, there is a chance my constant current source would not work. Specifically, if the load resistance varied at 100Hz, the Arduino would not be able to regulate the voltage at pin A0 for my circuit. My program is written so that the Arduino attempts to regulate the voltage entering

pin A0 by increasing or decreasing the voltage on the digital PWM pins to match the setpoint voltage every 100 milliseconds. My program does this with a setpoint resolution of 0.5mV and a regulation of 0.25V with a voltage limit of 15V. Due to the 0.1s delay in my execution loop and the load resistance changing at 100Hz or every 0.01s, my program would not be able to keep up. If I remove the delay in void loop, it will execute 117,000 times per second. I will probably want to update this line of code to delay the loop every 10ms (0.01s) or less.

6. **Please provide any comments or feedback relating to this lab/design project (Did I have adequate knowledge to solve the problem, Was the problem too easy/hard, Would I change or add anything)**:
    a. I think the directions should be more specific regarding how the Arduino is supposed to "be" a constant current source. Such as mentioning how the analog in-pin and digital PWM pins are related.