ABSTRACT

The objective of this project was to design and implement a portable 4–20 mA current source capable of simulating analog signals commonly used in industrial instrumentation and process control systems. The current source provides a selectable current output ranging from 4 mA to 20 mA, allowing for the simulation of various sensor signals such as those from transmitters and actuators. This was achieved using an Arduino Uno microcontroller, which outputs a pulse-width modulated (PWM) signal that is filtered into a DC voltage and subsequently converted into a current via a transistor-based driver circuit. Feedback is provided to the microcontroller through an analog-to-digital converter and both the setpoint and measured current are displayed on a 2x16 LCD. The device is powered by a 9V battery and includes an ICL7660 voltage doubler to supply the higher voltage levels required by the analog circuitry. The system offers coarse and fine current adjustments via push-button input, making it suitable for both quick testing and precise calibration tasks. The resulting unit is compact, cost-effective and suitable for field use where portability and ease of operation are essential.

Table of Contents

INTRODUCTION

Industrial processes require standard current signals (4-20mA) in order to drive their controlled systems; therefore an appropriate current sourcing device is required. This design report details the design of a portable 4-20mA current source that can be used to test industrial elements (such as a controlled valve). The power supply circuits consisted of a battery that was used to energize the microcontroller and a voltage doubler circuit was incorporated to generate the necessary supply voltage to energize the analogue circuits of this design. These circuits consisted of a numbers of passive components (resistors-capacitor circuits) and active components (integrated circuits).

CHAPTER 1

1.1 Introduction

This design report details the design of a 4-20mA portable current source. The first chapter describes the block diagram and general specifications of the overall design of this project. The block diagram is found in **Figure 1.1** below, followed by the general specifications as extracted from the block diagram.

1.2 Block Diagram

The following block diagram outlines the design circuit for the 4-20mA Current Source.

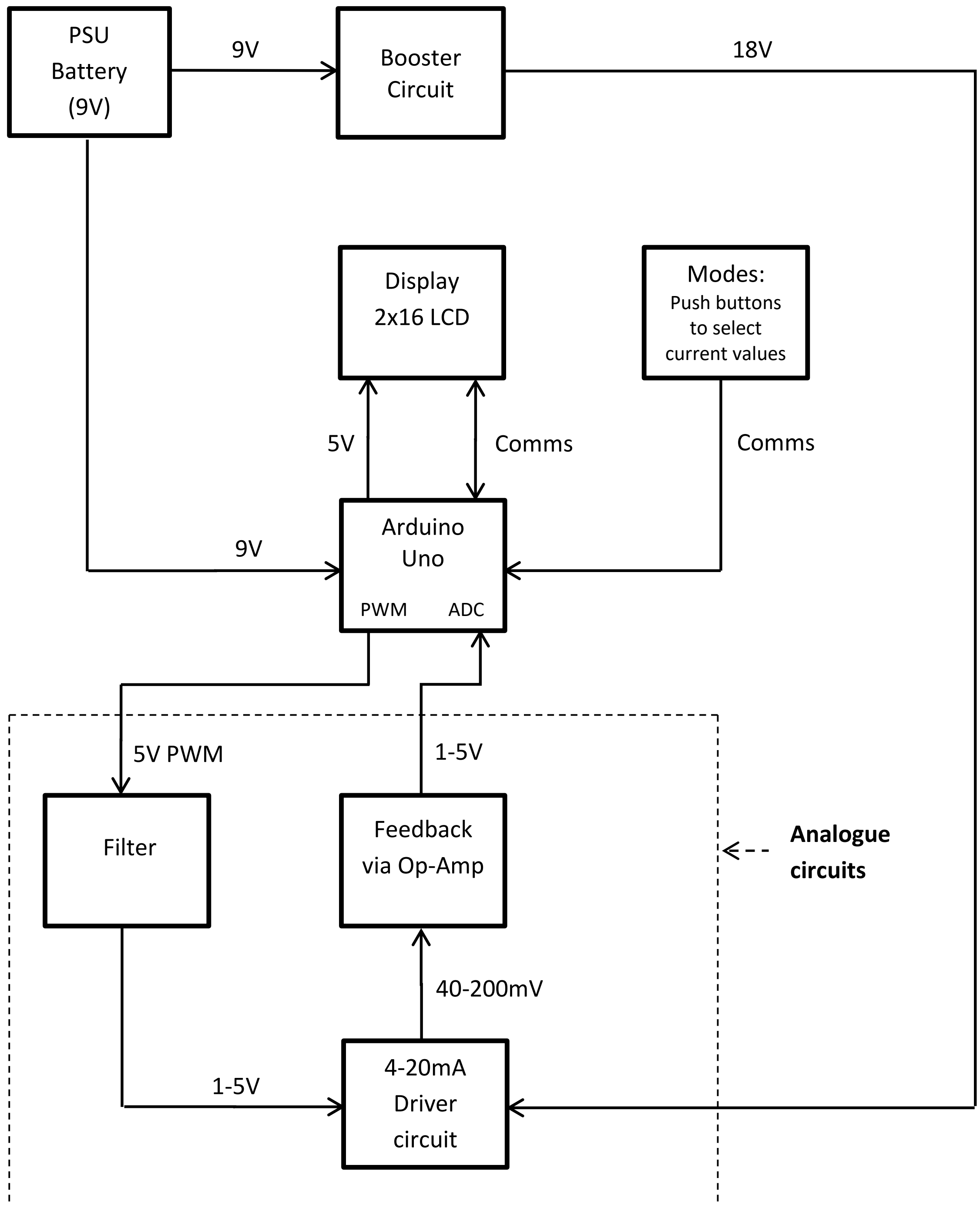

Figure 1.1: Block Diagram of the Portable 4-20mA Current Source

1.3 Specifications

1.3.1 Power Supply

A 9V DC battery was used to provide power to the various circuits of this 4-20mA Current Source.

1.3.2 Booster Circuit (Voltage Doubler circuit)

The 9V supply was not sufficient to energize the Driver and Feedback via Op-amp circuits therefore a Voltage Doubler circuit was necessary to increase this voltage to an adequate 17.4V (≈18V).

1.3.5 Display (2x16 LCD)

The Arduino microcontroller both powered the display, via one of its 5V output pins, and also controlled the current values displayed on it as selected by the user (Set point) using the necessary push buttons .

1.3.6 Modes

Two input modes were used consisting of 4push buttons: two push buttons to select discrete values (4mA, 12mA, 20mA) and the remaining two for the selected discrete values in increments of 0.1mA.

1.3.7 Microcontroller

An Arduino Uno microcontroller was selected, to implement the 4-20mA current source, due to its simplicity and adequacy. This design was achieved by making use of certain internal features of the Arduino microcontroller, such as the Analogue to Digital Converter (ADC) and Pulse Width Modulation (PWM). The Arduino processed the necessary input signals received from the push buttons and produced the corresponding output signals to the Analogue circuits to generate the appropriate amperage and it also displayed this amperage on the 2x16 LCD.

1.3.7.1 PWM

An output PWM analogue pin of the Arduino was used to generate a 5V PWM digital signal. Instead of using a Digital to Analogue Converter (DAC), which required more hardware, the built in PWM feature was combined with a low pass filter to produce the input signal to the Driver circuit.

1.3.7.2 ADC

One of the six analogue input pins of the Arduino Uno, that has a built-in ADC, was selected to convert the analogue feedback signal (Feedback via Op-amp) of 1-5V to a corresponding digital signal that was recognizable to the microcontroller.

1.3.8 Analogue Circuits

The Analogue circuits consisted of the Filter circuit, the 4-20mA Driver and Feedback via Op-amp circuit. Two Op-amps were used, one Op –amp combined with a Transistor formed the Driver circuit and the other formed the Feedback circuit.

1.3.8.1 Filter

To produce a 4-20mA current signal, a Direct Current (DC) analogue signal was required. A low pass filter combined with the PWM signal formed a Digital to Analogue converter that converted the PWM signal to a corresponding DC voltage. This DC voltage was fed into the 4-20mA driver circuit to produce the 4-20mA current signal.

1.3.8.2 4-20mA driver circuit

This Driver circuit received the DC voltage signal from the filter circuit and converted it to a corresponding constant current signal (of 4-20mA) that was delivered to the load. This circuit was configured using components and techniques that enabled the current signal produced to be independent of the load condition (within the operating range of the current source).

1.3.8.3 Feedback via Op-amp

An Op-amp was used to form the feedback signal. The voltage across a resistor in the Driver circuit was used to generate a 40-200mV signal. This voltage was than amplified by the Op-amp to form the 1-5V Feedback signal before it was fed to the microcontroller. The current value generated corresponds proportionately to the feedback signal generated. This feedback signal, which reflected the current signal that was being produced (measured variable) allowed the Arduino microcontroller to monitor and ensure that the correct current signal was produced (Set Point).

1.4 Conclusion

Integrating the 8 main circuits/components described in the block diagram and specifications formed the 4-20mA portable current source.

CHAPTER 2

2.1 Introduction

This was an attempt to build the Power Supply unit that consists of a 9V battery and booster (Voltage Doubler) circuit using the ICL7660.The Arduino was powered through the 9V battery that was internally regulated, by the Arduino, to 5V before energizing itself after which it supplied 5V to the display. The booster circuit was required to approximately double the input voltage (to 17,4V) to a sufficient level enabling it to energize the 4-20mA Driver circuit. The circuit diagram for the power Supply can be found in ***Figure 2.1*** below.

2.2 Power Supply Unit (PSU) Specifications

The PSU was configured as shown in ***Figure 2.1*** below.

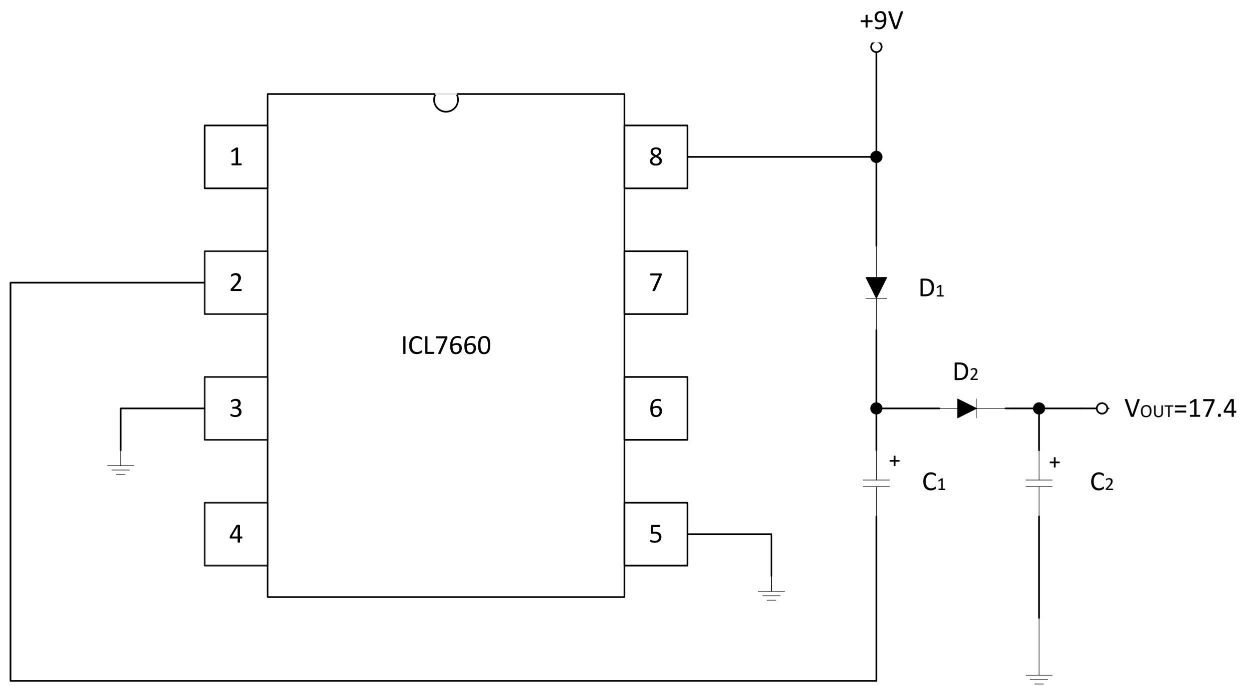

Figure 2.1: Circuit Diagram of Power Supply Unit

2.2.1 9V Battery

A 9V 570mAh Energizer battery was chosen as the input voltage source of the PSU. The ratings of this battery was adequate to supply the Arduino and (through the booster circuit) the 4-20mA current source circuit.

2.2.2 Booster Circuit

The ICL7660 integrated circuit was configured to build the Voltage Doubler (Booster circuit). The 4-20mA Driver circuit required a minimum of 15V to operate efficiently in this application. In order to achieve a voltage within this range this circuit was necessary to increase the insufficient 9V supply voltage to an acceptable 17.4V. The maximum supply voltage to this IC must not exceed 10.5V which was not violated by the 9V input. The maximum supply voltage to this IC must not exceed 10.5V which was not violated by the 9V input. The voltage doubler circuit produced an output voltage approximately 1.4V less than the ideal 18V because of the voltage drop across both diodes. The output was calculated using equation 2-1 below:

$$V_{out} = 2V_{in} - 2V_{diode} \qquad [2\text{-}1]$$

Two silicon diodes and two 10μF 50V capacitors are configured with the ICL7660 as shown in **Figure 2.1** above. The ICL 7660 performed the necessary switching action that was required for the charging and discharging of the capacitors in the circuit (which achieved the desired voltage doubling). An internal capacitor (from now on referred to as C3) within the ICL7660 was first internally connected to pin 8, resulting in a parallel connection to C1 and C2, so all three capacitors (C1, C2 and C3) were charging. For the first cycle the 3 capacitors (C1, C2 and C3) were charged by the 9V supply until the ICL 7660 switched the internal capacitor C3 to pin 2, during the second cycle. This resulted in capacitor C3 to be connected in series to capacitor C1 and the three capacitors, C1 C2 as well as C3 along with the 9V voltage, discharged into the output. The two diodes D1 and D2 allowed for only one direction of current flow in the circuit. Diode D1 prevents the capacitors C1 and C3 from discharging in the direction of the supply, while diode D2 prevents capacitor C3 from being discharged back in the direction of the supply or capacitor C1. The pin diagram of the ICL7660 can be found in ***Figure 2.2*** below [7].

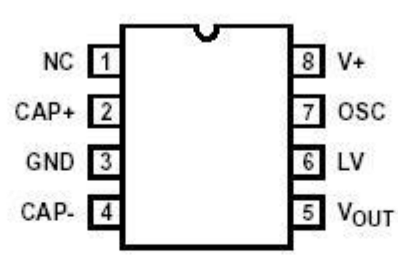

Figure 2.2: Pin Diagram of ICL7660

2.3 Conclusion

The build of this power supply using the 9V 570mAh battery input and the ICL7660 configured booster circuit successfully produced the required power outputs to successfully energize the microcontroller, display and 4-20mA driver circuit.

CHAPTER 3

3.1 Introduction

The Analogue circuits, shown in ***Figure 3.1*** below, consisted of a passive first order Low pass filter, a 4-20mA Driver Circuit (also referred to as the Voltage to Current Converter circuit) and a Feedback Circuit. The filter was constructed using a resistor and capacitor. The 4-20mA Driver Circuit consisted of one Op-amp from the LM324 integrated circuit, one resistor R1 and one 2N2102 Transistor (Q1). The feedback circuit consisted of three resistors (R2, R3, and R4) and a second Op-amp from the LM324 IC configured as a Non-inverting Amplifier.

3.2 Analogue circuit specifications

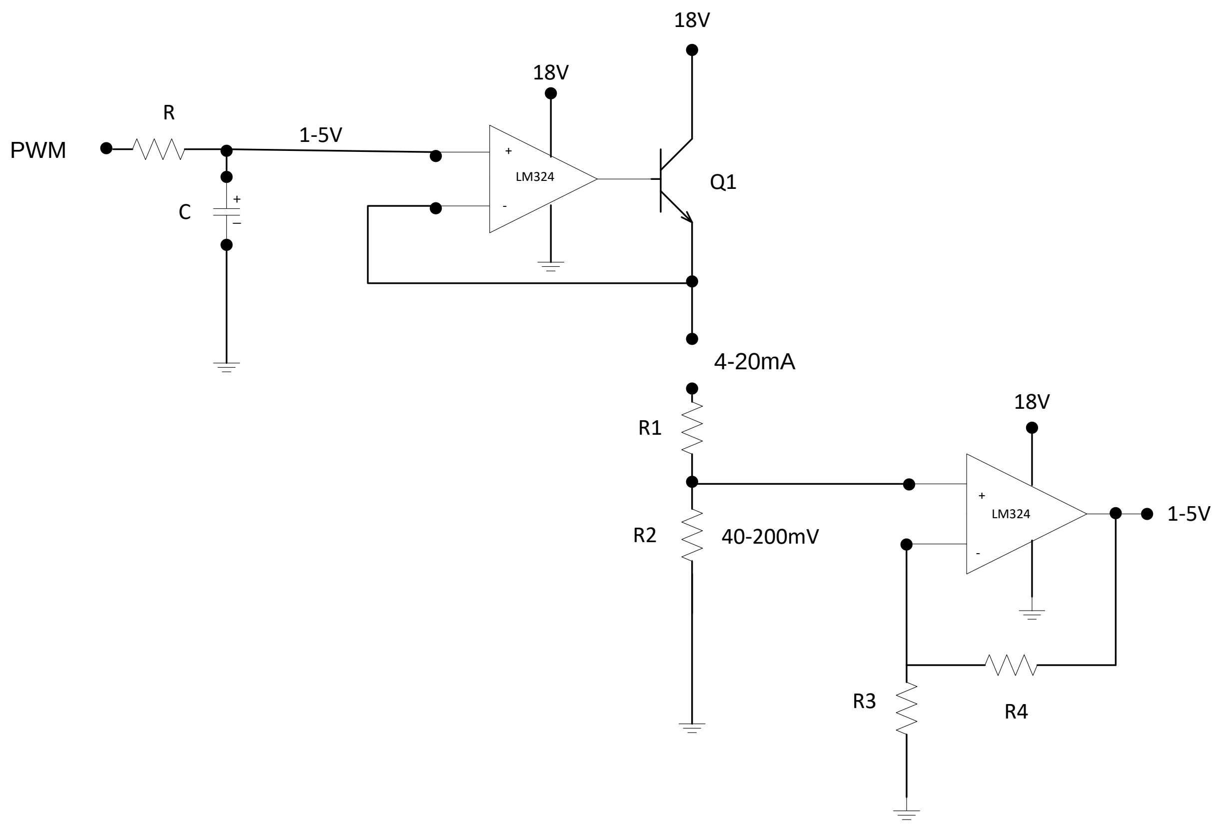

Figure 3.1: Analogue Circuits

3.2.1 Filter circuit

A Low pass filter was introduced to convert the Digital square wave PWM signal to a Direct Current (DC) analogue signal. The PWM and Low pass filter combination functioned as a Digital to Analogue Converter (DAC). The PWM digital (8-bit) signal was converted to a 1-5V DC voltage by the filter circuit, with a resolution of approximately 0.02 volts/bit, before it was fed into the non-inverting pin of the LM234 Op-amp (4-20mA driver circuit). Pin 6 was chosen for the PWM signal from the microcontroller that ran at approximately 980 Hz. This was chosen over the 490 Hz carrier frequency option because a higher frequency resulted in a higher resolution which resulted in improved current control. To obtain the cut-off frequency (f_c) of the filter circuit the equation governing the magnitude bode plot of the filter circuit was derived as follows [14]:

The gain of the filter is (Av),

$$Av = \frac{Xc}{R+Xc} \qquad [3\text{-}1]$$

And,

$$Xc = \frac{1}{j\omega C} \quad [3\text{-}2]$$

Then substituting equation 3-2 into equation 3-1 forms,

$$Av = \frac{\frac{1}{j\omega C}}{R+\frac{1}{j\omega C}}$$

And after multiplying out,

$$Av = \frac{1}{1+j\omega RC} \quad [3\text{-}3]$$

Also,

$$\tau = RC \quad [3\text{-}4]$$

And also considering the angular cut-off frequency(ω_c),

$$\omega_c = \frac{1}{\tau} \quad [3\text{-}5]$$

Then by substituting equation 3-4 into equation 3-5 and thereafter substituting 3-5 into equation 3-3 yields,

$$Av = \frac{1}{1+j\frac{\omega}{\omega_c}} \quad [3\text{-}5]$$

Now because,

$$\omega_c = 2\pi f_c \quad [3\text{-}6]$$

Where f_c is the cut off frequency. And,

$$\omega = 2\pi f \quad [3\text{-}7]$$

Substituting equation 3-6 and equation 3-7 into equation 3-5 results in,

$$Av = \frac{1}{1+j\frac{f}{f_c}} \quad [3\text{-}8]$$

Finally by finding only the magnitude expression of equation 3-8 in terms of decibels renders,

$$|F(j\omega)| = -20\log(\sqrt{1+(\frac{f}{f_c})^2}) \qquad [3\text{-}9]$$

Using equation 3-9 the cut-off frequency was determined to be 98.5 Hz (approximately 100 Hz) by substituting the frequency (f) of 980 Hz (PWM carrier frequency) and -20dB (-20dB/dec) as the magnitude (|F(jω)|). The magnitude -20dB is conventionally used to find the cut-off frequency for first order filters. Once the cut-off frequency was calculated and a value for R chosen as 1kΩ the equation 3-10 below was used to calculate the corresponding capacitor value (C):

$$f_c = \frac{1}{2\pi RC} \qquad [3\text{-}10]$$

The capacitor (C) was calculated to be approximately 10µF. The voltage tolerance of the capacitor was chosen to be 50V which is well above the expected maximum voltage of 5V. The Amplitude versus frequency plot of this Low pass filter can be seen in ***Figure 3.2*** below:

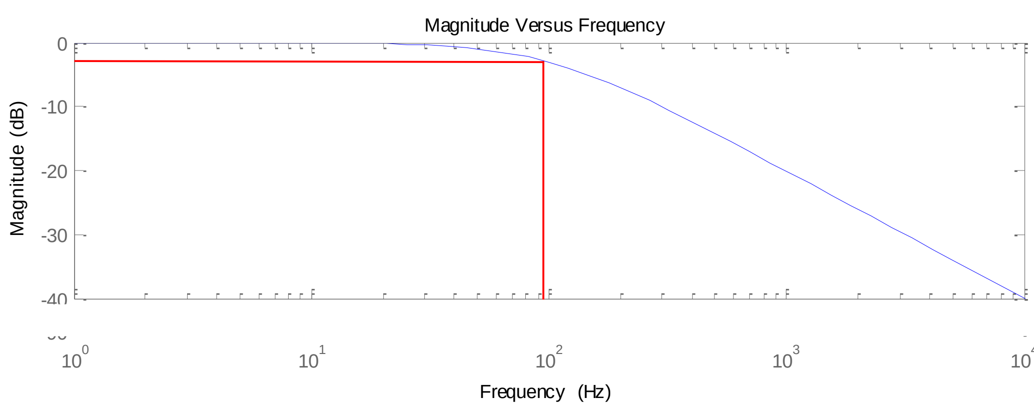

Figure 3.2: Amplitude versus Frequency Plot of the Filter Circuit

3.2.2 4-20mA Driver Circuit

The Driver circuit functioned as a Voltage to Current converter circuit; the input voltage from the filter was converted into a current that formed the base current to the transistor 2N2102. The amount of base current produced controlled the amount of current flowing through the collector (4-20mA signal). The (Ideal) 18V from the voltage doubler, which was well within the maximum tolerable supply voltage of 32V, was used to energize the LM324. The filtered PWM signal is fed into the positive non-inverting pin of the first op-amp in the LM324. The resistors and transistors were configured as indicated in ***Figure 3.1.*** The resistor labelled R1 and R2, which was connected to the emitter of the transistor. These two resistors had the same voltage across it as the input signal to the Op-amp. Also the current in the emitter (I_E) and the current in the collector (I_C) had a difference of base current (I_B) according to the equation below:

$$I_E = I_C + I_B \qquad [3\text{-}12]$$

From a Design perspective, because base current is negligibly small, the current through the collector current was said to be equal to the current through the Emitter:

$$I_E = I_C \qquad [3\text{-}13]$$

Therefore knowing that the voltage was between 1-5V and the desired corresponding current to be produced in the emitter to be 4-20mA, the appropriate resistance of R1 and R2, connected in series, was calculated to be 240 Ω and 10 Ω (250Ω in total) using Ohms Law:

$$V = I \times R \qquad [3\text{-}14]$$

This concludes the driver circuit. The resistor R2 was connected to the emitter of Q1 to form the feedback that was fed into the feedback Op-amp and is discussed below.

3.2.3 Feedback via op-amp circuit

The current signal produced by the Driver circuit needed to be monitored and controlled by the microcontroller to ensure that the correct current amperage was produced; therefore a feedback signal was generated and fed back to the microcontroller. The current from the 2N2102 transistor collector was used to create a voltage drop of 40-200mV on the resistor R2, therefore R2 was calculated to be 10Ω and R1 to be 240 Ω by equation 3-14. The voltage drop produced was then fed through to a second Op-amp, which was configured as a non-inverting amplifier, using the LM324 IC. To obtain a signal of 1-5V feedback signal to microcontroller, a gain of 25 was implemented. This type of amplifier configuration was necessary to produce a non-inverting amplified signal. The following equation governs the Non-inverting amplifier:

$$V_{o = 1 + \frac{R_4}{R_3}} \qquad [3\text{-}15]$$

R3 and R4 were calculated as 10 Ω and 240 Ω respectively. The 1-5V output signal (feedback signal) was then applied to an Analogue to Digital Converter (ADC) to convert it into a Digital Signal before it was fed back to the microcontroller.

3.3 Conclusion

The complete Analogue circuit produced a current of 3.9-19.4mA when tested using a varying input voltage of 1-5V, which is not exactly the desired 4-20mA signal under a single load condition. Due to rounding off errors (using resistor and capacitor values that are not exactly equal to calculated values) and resistor approximations (resistors are not exactly the value they are designed to be) as well as the fact that the filter circuit does not produce a perfectly DC voltage signal, a small discrepancy from the desired is expected.

CHAPTER 4

4.1 Introduction

To display the current values selected by the user a 2x16 Liquid Crystal Display (LCD) screen was programmed and used. It was chosen for its versatility of displaying special and custom characters (was able to display text, symbols and numbers) as well as the fact that it was simple to program.

4.2 Specifications

4.2.1 2 x 16 LCD

The 2x16 LCD was powered by the Arduino microcontroller. This display unit displayed up to 16 characters per line, on 2 such lines, and each character is displayed in a 5 x 7 pixel matrix [15]. It contains 2 registers; the Command register that stores programmed instructions and the Data register that stores data, as ASCII code, to be displayed on the LCD. Line one displayed current values selected and line two displayed the feedback current value in milliamps. The Circuit was configured as shown in ***Figure 4.1*** below [19].

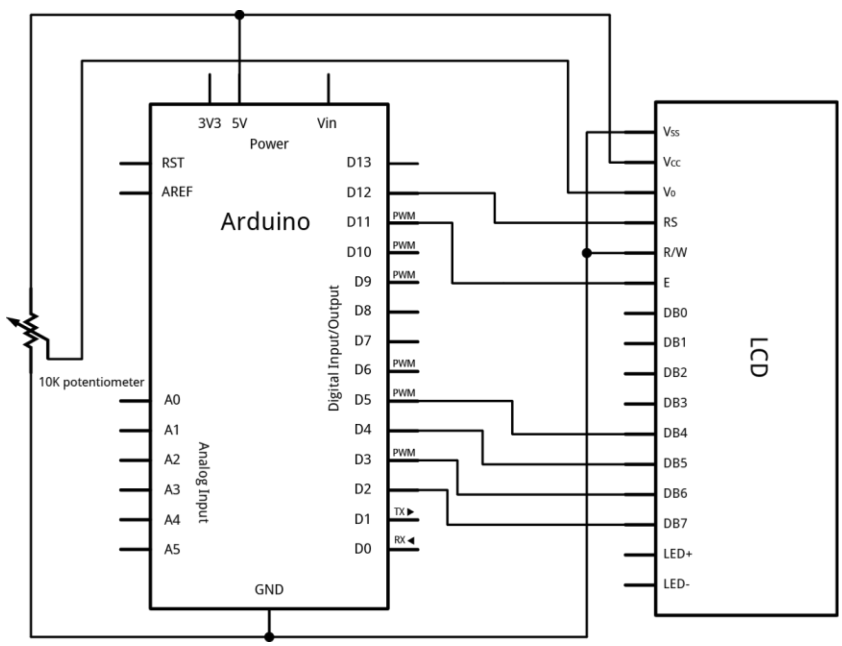

Figure 4.1: Arduino to 2x16 LCD Circuit Diagram

4.3 Conclusion

Using an Arduino uno with a 2x16 LCD, adequately displayed the current values as well and the necessary characters.

CHAPTER 5

5.1 Comprehensive Circuit and Results

By Integrating the 3 circuits above, the portable 4-20mA source was built. The comprehensive complete circuit diagram of the 4-20mA current source can be found in **Annexure: A**. Push buttons P1 and P2 were used to toggle incrementing the current by 4mA and decrementing the current by 4mA respectively. Another 2 buttons were used to increment or decrement the current value by $\pm$ 4 mA with a resolution of 0.1mA.

The ***Figure 5.1*** below shows the desired value (Set-point) entered and the corresponding current produced (Measured value).

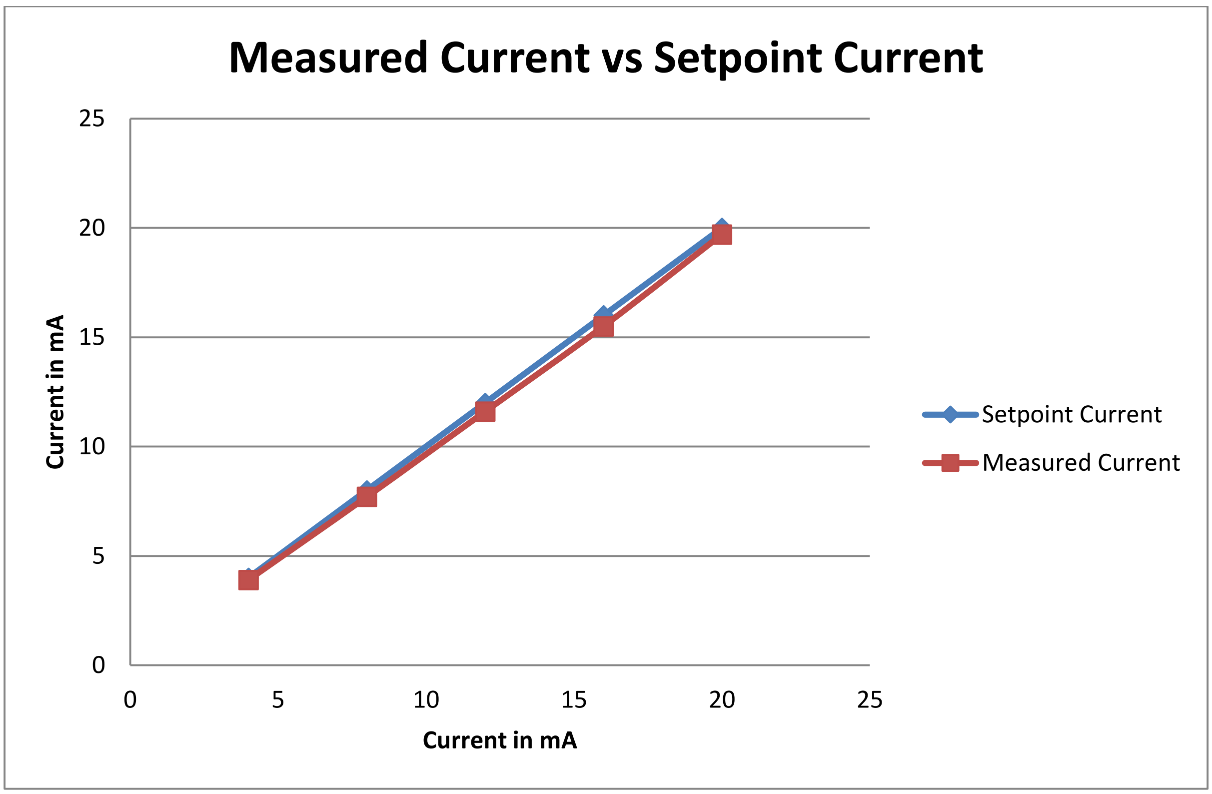

Figure 5.1: Graph Showing Desired Values Compared to Measured Values

CONCLUSION

This 4-20mA current source successfully produced constant current amperage between the range required for varying load conditions. However produced 3.9-19.4mA instead of the desired 4-20mA. For future recommendation, more accurate current signal can be achieved by adjusting the PWM signal of the Arduino, in the code, due to the appropriate current setpoint.

REFERENCES

[1] Arduino analogue output available at:
http://engineeringnotes.blogspot.com/2013/09/analog-output-from-microcontroller-dac.html [10 February 2015]

[2] Pulse width modulation avialable at:
http://digital.ni.com/public.nsf/allkb/294E67623752656686256DB800508989 [11 February 2015]

[3] Driver circuit available at:
http://en.wikipedia.org/wiki/Driver_circuit [9 February 2015]

[4] Booster circuit available at:
http://en.wikipedia.org/wiki/Boost_converter [8 February 2015]

[5] Parameters of ICL 7660 Available at:
http://icl7660.chinaicmart.com/ [19 February 2015]

[6] Voltage doubler circuit configuration Available at:
http://circuit-diagram.hqew.net/Positive-Voltage-Doubling-Circuit-using-ICL7660-Voltage-Doubler_5902.html [19 February 2015]

[7] Datasheet of ICL 7660 available at:
http://www.intersil.com/content/dam/Intersil/documents/icl7/icl7660.pdf [20 February 2015]

[8] The LM324 datasheet available at:
http://www.onsemi.com/pub_link/Collateral/LM324-D.PDF [1 March 2015]

[9] The Transistor 2N2102 datasheet available at:
https://www.fairchildsemi.com/datasheets/MM/MMBT5088.pdf [March 2015]

[10] The RC circuit formula and information available at:
http://www.electronics-tutorials.ws/filter/filter_2.html [4 March 2015]

[11] Arduino specifications available at:
http://arduino.cc/en/Reference/analogWrite [4 March 2015]

[12] Thomas L. Floyd, *Electronic Devices conventional current version,* 9th Edition. Pearson, 2012.

[13] Differential Amplifier circuit and calculations available at:

http://www.electronics-tutorials.ws/opamp/opamp_5.html [9 March 2015]

[14] Low pass filter theory and calculation available at:

http://www.electronics-tutorials.ws/filter/filter_2.html [9 March 2015]

[15] The 2x16 LCD Available at:
http://www.engineersgarage.com/electronic-components/16x2-lcd-module-datasheet [15 March 2015]

[16] Interfacing the 2x16 LCD with Arduino board Available at:
http://arduino.cc/en/Tutorial/LiquidCrystal [15 March 2015]

[17] Pin specifications of 2x16 LCD Available at:
https://electrosome.com/lcd-pic-mplab-xc8/ [15 March 2015]

[18] Table of spin specifications Available at:
http://www.griaulebiometrics.com/en-us/manual/rex-device/adding-peripherals/adding-liquid-cristal-display [15 March 2015]

[19] Circuit Diagram of Arduino connected to 2x16 LCD available at:
http://arduinoliquidcrystal.readthedocs.org/en/latest/liquidcrystal.html [16 March 2015]

Annexure A: Code

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(12,11,5,4,3,2);

const int CoarseUpPin = 7;
const int CoarseDownPin = 8 ;
const int FineUpPin = 13;
const int FineDownPin = 10;
const int PwmPin = 6;
float SetVoltage = 1;
float SetCurrent;
int count = 1;

void setup() {

// initialize inputs and outputs:
pinMode(CoarseUpPin, INPUT);
pinMode(CoarseDownPin, INPUT);
pinMode(FineUpPin, INPUT);
pinMode(FineDownPin, INPUT);
pinMode(PwmPin, OUTPUT);

lcd.begin(16, 2); // initialize lcd communication:

lcd.setCursor(0, 0);
lcd.print("SETPOINT:");
lcd.setCursor(14, 0); //start the letters mA at coloumn 14 of row 0 to allow adequate space for values
lcd.print("mA");

lcd.setCursor(0, 1);
lcd.print("MEASURED:");
lcd.setCursor(14, 1);// start letters mA directly below mA of setpoint
lcd.print("mA");

}

void loop() {

 //Increasing coarse adjustment
  if(digitalRead(CoarseUpPin)== HIGH){
```

```
  count++; //increment count by 1

  if(count==1)
    SetVoltage = 1;
  else if(count==2)
    SetVoltage = 2;
  else if(count==3)
    SetVoltage = 3;
  else if(count==4)
    SetVoltage = 4;
  else if(count==5)
    SetVoltage = 5;
  else if(count == 6){
    count = 5;
    SetVoltage = 5;
  }
}

//Decreasing coarse adjustment
if(digitalRead(CoarseDownPin)== HIGH){

  count--; //decrement count by 1

  if(count==1)
    SetVoltage = 1;
  else if(count==2)
    SetVoltage = 2;
  else if(count==3)
    SetVoltage = 3;
  else if(count==4)
    SetVoltage = 4;
  else if(count==5)
    SetVoltage = 5;
  else if(count == 0){
    count = 1;
    SetVoltage = 1;
  }
}

//Increasing Fine adjustment
if(digitalRead(FineUpPin)== HIGH){
```

```
    SetVoltage = SetVoltage + 0.025; // A change of 0.025 V results in a change of 0.1 mA

    if(SetVoltage > 5)
      SetVoltage = 5;
  }

  //Decreasing Fine adjustment
  if(digitalRead(FineDownPin)== HIGH){

    SetVoltage = SetVoltage - 0.025;
    if(SetVoltage < 1)
    SetVoltage = 1;
  }

  SetCurrent = 20 - 4*((5 - SetVoltage));

  analogWrite(PwmPin,51*SetVoltage); // 51 because total duty cycle is 255. For setpoint values of 1-
5V, pwm value is 51-255

  float RawADC = analogRead(A1); //Read the FeedBack signal in raw data format
  float Feedbackvoltage = 5 *RawADC/1023 ; //Convert it to a voltage value
  float Feedbackcurrent = 20 - (4*(5 - Feedbackvoltage)); //Similar formula as Setpoint to calculate the
measured current value

  lcd.setCursor(10, 0);
  lcd.print("    "); // To clear the space after current reaches double digits to omit the zero that appears
next to the "mA" when the current returns to single digits
  lcd.setCursor(10, 0);
  lcd.print(SetCurrent,1);

  lcd.setCursor(10, 1);
  lcd.print("    ");
  lcd.setCursor(10, 1);
  lcd.print(Feedbackcurrent,1);

  delay(200);
}
```

Annexure B: Complete Circuit Diagram