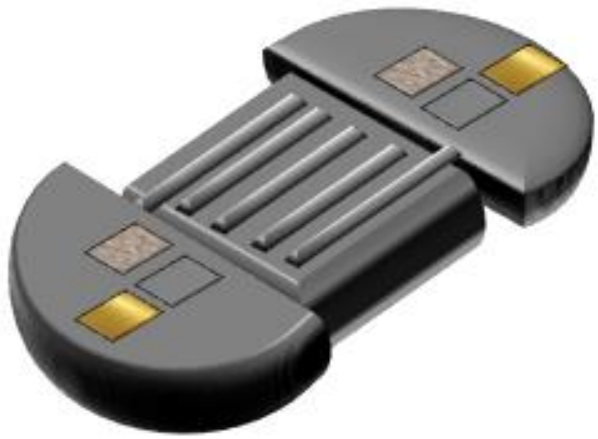
Oven-Control Circuit for TPoS MEMS Oscillator



Senior Design I Project Document

Department of Electrical and Computer Engineering

University of Central Florida

Dr. Lei Wei

Group 13:

Megan Driggers - Electrical Engineering

Heather Hofstee - Electrical Engineering

Michaela Pain - Computer Engineering

Project Sponsor:

Dr. Reza Abdolvand, UCF ECE Associate Professor

November 30, 2018

Table of Contents

List of Figures

List of Tables

1. Executive Summary

This project focuses on designing a temperature control circuit for a thin-film piezoelectric on silicon microelectromechanical systems (MEMS) oscillator. MEMS oscillators have an advantage over traditional crystal oscillators in the sense that they can be fabricated using conventional semiconductor fabrication methods and can often be smaller in size. However, these MEMS oscillators vary in performance depending on ambient temperature. For this reason, this project focuses on designing a printed circuit board that will keep the resonator at a steady temperature to stabilize and optimize the performance of the device. Oven-control circuits for MEMS oscillators have been created before; however, this one is unique because it will use a thin-film piezoelectric on silicon (TPoS) oscillator that is a subject of research for the project sponsor, Dr. Reza Abdolvand.

To keep the resonator temperature stable, current will be passed through it to elevate its temperature beyond the industrial temperature limit using Ohmic resistive heating. The temperature will be controlled with a microcontroller and set via user input. Prior testing data will be used to correlate resistance and temperature, and the microcontroller will be used to program a control loop to keep the resonator temperature stable. The output of temperature and resistance of the resonator will be displayed to an LCD screen so that the user can know the values.

This report details the design plan of the printed circuit board to achieve temperature control and stabilization of the resonator. It will cover an introductory research background for MEMS, oscillators, and resonators as well as project motivation. Project responsibilities will describe which aspect of the project will be completed by which team member; finances will detail the budget, and the schedule will give week by week goals to keep the project moving forward. Some of the initial project theory of operation details are based on the work of one of the team members for her thesis for the Honors in the Major program at UCF, but the microcontroller integration for this circuit is unique and requires a group effort.

In this document, there are also design requirements, which are operational features the project must have, and design constraints, which are imposed by the environment, ethics, manufacturing restrictions, and more. Design standards are also explored. These are different health, safety, and operational limits set by standards bodies that apply to this project. Schematic design, the process of describing how the circuit will function from an electrical engineering perspective in terms of voltages, currents, and components, will be explained to describe hardware function. Each component of the project hardware will be compared and have actual parts chosen for the printed circuit board (PCB) design. There will also be a section on PCB design and how the different inputs and outputs were chosen as well as how the layout and grounding steps have been approached.

Furthermore, the document goes over the power requirements of each component and how the DC-DC conversions will be done. For the software design, the calculations, algorithms, programming languages, voltage control, and more are listed and compared in detail. The chosen control system will also be described, and its predicted characteristics will be explained. The project construction, bill of materials, testing, preliminary tests, characterization, and operational instructions are also included in this document.

2. Project Description

This section provides an overview of the project's background and motivation and requirements.

2.1 Background

Microelectromechanical resonators are microsystems that resonate after being stimulated electrically. These types of resonators have several sensing, filtering, and timing applications. The particular resonator to be used for this project is an oscillator [1].

Oscillators are devices used to produce periodic electric currents or voltages through exchanges in kinetic and potential energy. The electronic signals generated by these devices are often a product of the circuit design and the values of the components; however, they generally assume sine or square waveforms. Oscillators can convert direct current (DC) from a given power supply to an alternating current (AC) signal. They are often distinguished by their output signal frequency and output signal type, and this lends to different applications. For example, oscillators can generate clock signals exercised in computers and broadcast signals utilized in transmitters.

The output signal frequency of a quartz oscillator is affected by the temperature of the quartz crystal, which can impact its resonance frequency. An oven-controlled crystal oscillator (OCXO) is a specific type of oscillator that controls the temperature of an oscillator circuit using an oven. This type of oscillator is often used to provide improved temperature stability and frequency accuracy with respect to a standard crystal oscillator. The drawback to using an oven-controlled crystal oscillator is that it often consumes a lot of power and space, which can be expensive.

A resonator is a device that naturally oscillates at its resonant frequencies with a greater amplitude than at other frequencies. The oscillations can be generated either mechanically or electromagnetically. This allows the resonator to generate and detect specific frequencies. Resonators with mechanically generated oscillations have applications such as stringed and percussion instruments, where specific frequencies are generated utilizing acoustic cavity resonators, guitar resonators, and so forth. Resonators with electromagnetically generated oscillations have applications such as lasers or particle accelerators, which generate certain frequencies by transferring energy using resonator cavities.

The temperature coefficient of resonance frequency (TCF) is used to determine the thermal stability of a resonator. The resonance frequency changes with temperature as a result of shifts in the modulus of elasticity, structural damping, and thermal expansion or contraction of different materials. Thus, thermal stability is important to consider because temperature change affects the resonant frequencies of the system. The TCF is found by placing a test sample within a cavity on a low-loss, low-dielectric constant, and low-thermal expansion material. The cavity is then placed within a temperature chamber, and the resonant frequency is measured at each temperature over the desired range of temperatures. The TCF can then be calculated and expressed in parts-per-million-per-degree Celsius (ppm/°C) [2].

Thin-film piezoelectric-on-substrate (TPoS) resonators have been the subject of Dr. Abdolvand's research for over a decade. These resonators are a type of lateral bulk acoustic resonators (which vibrate via expansion and contraction due to electric signal converted into a force) specifically shown to have high Q-factors in the MHz range [3]. TPoS resonators involve creating piezoelectric components (which will translate electrical energy into mechanical energy) as a part of the silicon bulk [1].

The oscillator to be used here is a microelectromechanical systems (MEMS) TPoS resonator. The resonator in this application acts as a filter for the frequencies and attenuates all but the resonance frequency and several of the surrounding frequencies. This number largely depends on the Q-factor of the resonator. A higher Q-factor results in a narrow attenuation band and, thus, has a better performance by reducing the noise of unwanted frequencies also being fed back to the resonator.

2.2 Motivation

This project focuses on designing an oven-control circuit for a TPoS MEMS oscillator to keep it at a constant resonance frequency. While circuits of this type have been designed before, this one will be unique because of the specific type of resonator used. The TPoS resonator to be used has been designed by researchers at the University of Central Florida (UCF) in the Dynamic Microsystems Lab under the direction of Dr. Reza Abdolvand.

MEMS resonators have been shown to have applications as a smaller, more easily fabricated, and sometimes less expensive oscillator compared to the current crystal oscillators that dominate the market. The challenge with MEMS oscillators currently centers around their performance, especially since they have a relatively high TCF. The TCF details the changes in resonance frequency with respect to temperature and is minimized at the resonance frequency.

Because resonator performance is affected by temperature, having a method of stabilizing the temperature to the value corresponding to the resonance frequency should help optimize and stabilize the resonator performance, minimizing changes in the frequency by keeping the resonator operating at a temperature where the TCF is stable. Because cooling the resonator would require a more elaborate setup than heating it, this project will seek to heat the resonator to a stable temperature above the standard industrial limit of 85ºC. The finished product should stabilize the elevated temperature of the resonator within a distinct range of deviation.

The temperature will be controlled via the current passed through the resonator. Prior research will be used to ascertain the relationship between resonator temperature and current passed through it. A control loop could be used in conjunction with a microcontroller to ensure that the temperature remains stable.

The desire to demonstrate the knowledge and problem-solving skills acquired at UCF is an additional motivation for this project. In addition, the team-based project opportunity allows for the gain of communication and project management experience that is relevant for any type of career. The two semester journey of undertaking a project proposal, design and implementation highlights the challenges and rewards of hard work, timeliness and collaboration.

2.3 Deliverable Requirements

Hardware Deliverables:

1. Low power
2. Parts per billion (ppb) frequency stability
3. Communication (must be able to relay temperature and resistance to user)

Software Deliverables:

1. Controls within ppb accuracy (most likely 0.1-0.2ºC accuracy)
2. Correct speed of program for stability
3. Efficient code

Project responsibilities are outlined as follows:

Megan Driggers (electrical engineer):

- Design PCB
- Oversee power and voltage requirements
- Design control loop

Heather Hofstee (electrical engineer):

- Lead team and manage project
- Act as liaison between team and Dr. Abdolvand
- Design hardware schematic

Michaela Pain (software engineer):

- Design software
- Choose and program microcontroller
- Add additional features for interface

2.4 House of Quality

Figure 2.1 is a type of engineering trade-off matrix called a house of quality. It shows product requirements and engineering requirements and how they affect each other. The tradeoff matrix is designed to show how the optimization of one requirement may improve or degrade another. Some have no correlation and have been left blank. All others show a correlation as indicated by the legend on the right-hand side of the figure. The roof of the house of quality shows the intersection of the different engineering requirements and their correlation, while the center of the diagram shows the tradeoff specifically between product and engineering requirements.

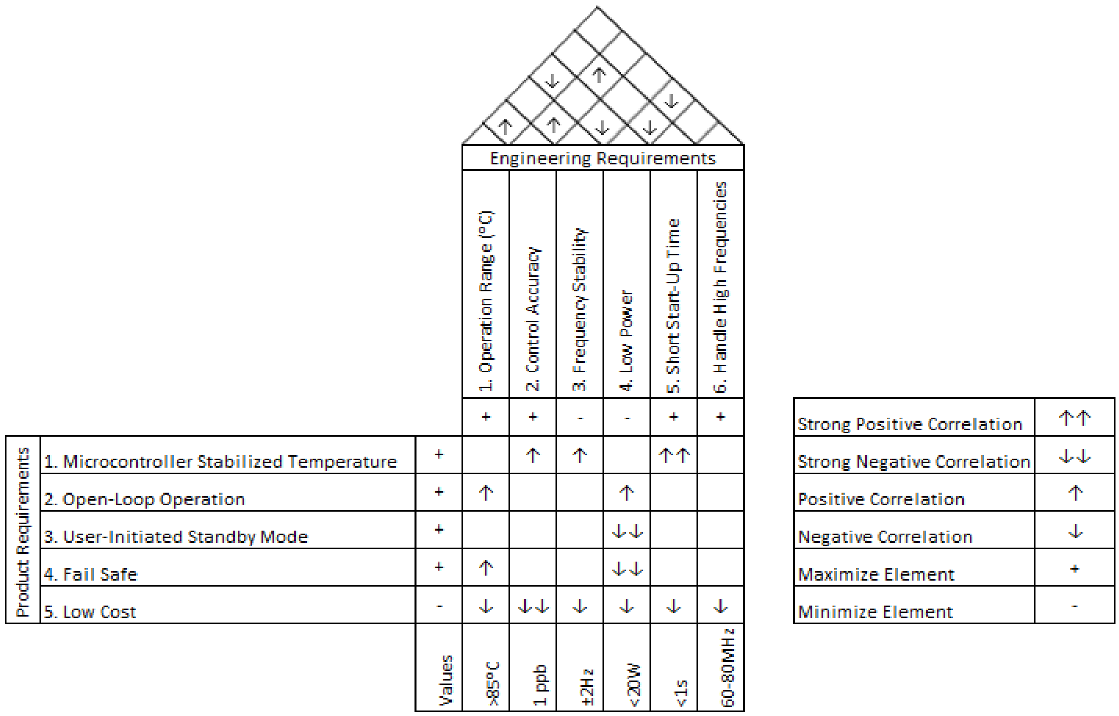


Figure 2.1: House of quality

2.4.1 Engineering Requirements

Engineering requirements have been listed on the top of the house of quality. These are deliverable requirements that have been specified by the project sponsor and are measurable values that can be demonstrated when the final project is complete. These are not necessarily essential to the overall goal of the project but are important to meet optimization standards specified by the project sponsor.

2.4.2 Product Requirements

Product requirements refer to basic needs of the project that must be satisfied to meet the project objectives. For instance, having an open loop operation mode and being fail safe are essential to the project but do not have a specific value. Hence, they are product requirements.

2.5 Overall Project Responsibilities

Figure 2.2 shows the overall project diagram and the responsibilities of each team member. To have each member's experience used in the most beneficial way for the project, the following work assignments were made based on prior qualifications. To reiterate, Megan will design the PCB and support the power supply, Michaela will support the microcontroller and user display, and Heather will design the hardware schematic and lead the team.

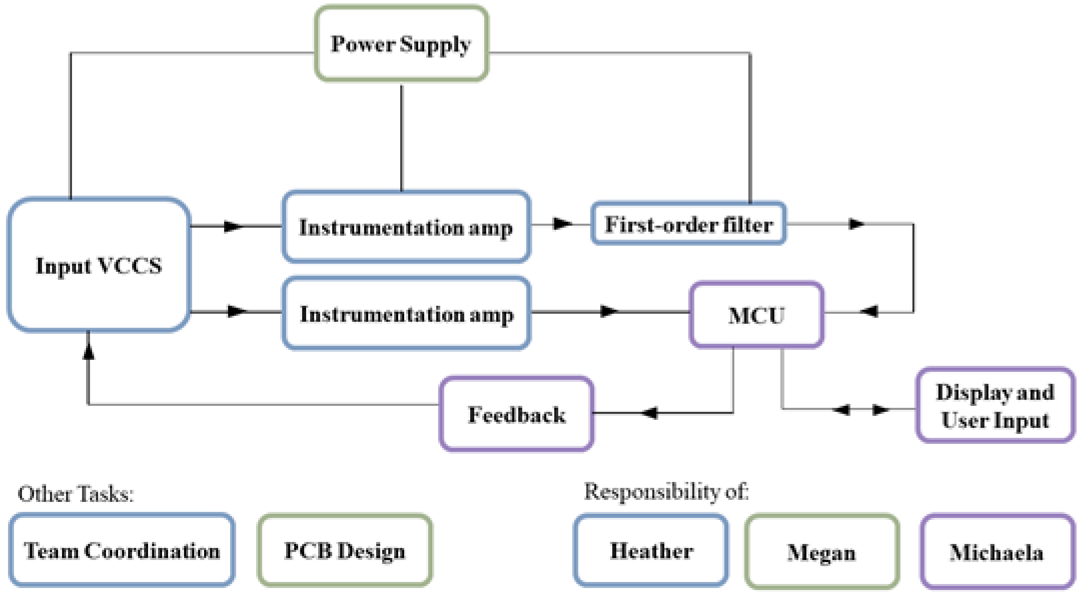


Figure 2.2: Overall project diagram

3. Project Research and Component Selection

This section discusses the components from within the schematic design in further detail. This includes elaboration in terms of their purpose, process of choosing specific parts, and relevant calculations. The components discussed are the resonator, microcontroller, LCD screen, first order filter, instrumentation amplifier, voltage reference, potentiometer, and power supply. In addition, this section will elaborate on the software tools embraced by the team to support increased organization, collaboration and efficiency.

3.1 Resonator

The resonator to be used in this project is a TPoS resonator. Research has shown them to offer high-Q (quality factor) and high-power handling. They have a thin layer of piezoelectric material on a substrate (often silicon) [1]. The largest drawback of these types of resonators is the relatively high temperature coefficient of frequency (TCF). Lightly doped silicon's TCF is usually about -30ppm/°C while MEMS resonators on silicon usually have a TCF of about -50ppm/°C. Since the TCF is a bell-shaped curve, the turnover temperature is where the TCF changes polarity and is the point where the TCF is minimized. This turnover temperature is dependent on the doping concentration and resonant mode [4]. The desire of this project is to stabilize the resonator temperature so that it operates near the turnover temperature and thus operates with a minimized TCF.

While active temperature circuits have been successfully designed and implemented for resonators in the past, this project is unique because of the resonator to be used. The particular type of resonator that the circuit will be designed for is one that has unique properties in regard to its TCF. The image below shows typical examples of a TCF graph with a fairly linear correlation. Thus, the TCF does not have a particular point where it stabilizes.

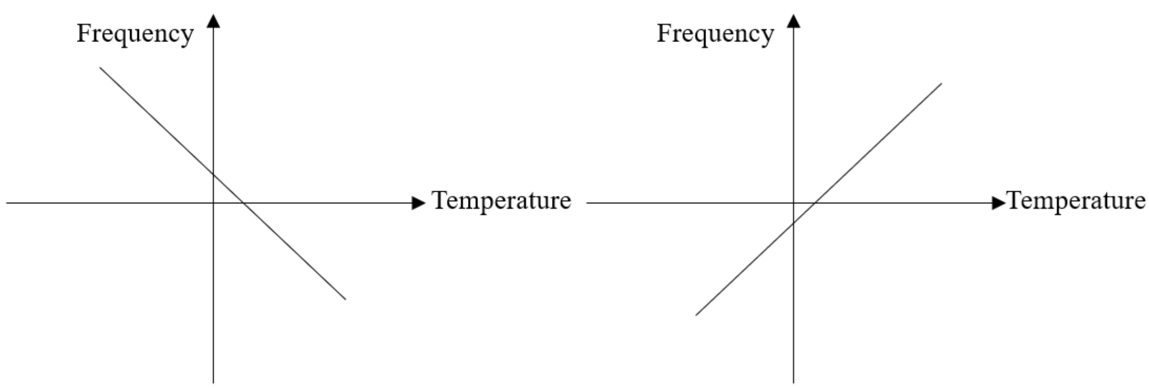


Figure 3.1: Typical TCF curves

In contrast, the device for this project will exhibit a TCF much more like the following graph:

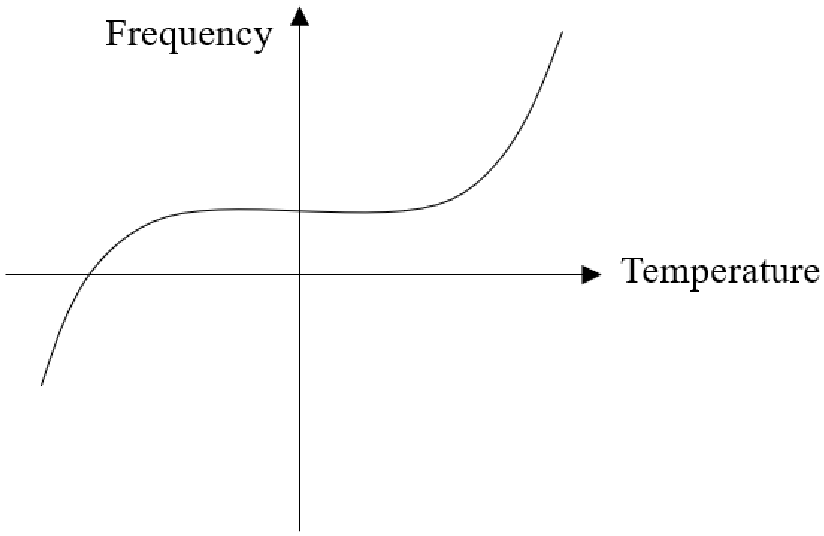


Figure 3.2: TCF curve for resonator of this project

This resonator's TCF is unique because near the turnover temperature (where the slope changes polarity), the change in TCF is close to or equal to zero. Thus, keeping the resonator operating near this turnover temperature will allow for precise control over the device's resonance frequency. It is the design of the resonator that should allow for ppb accuracy. The resonator will be designed to have a turnover temperature at or above 90°C.

The resonator for this project has been designed and fabricated by a Ph.D. student at the University of Central Florida working under the supervision of Dr. Abdolvand. While Dr. Abdolvand's group fabricates a variety of resonators, this particular resonator was selected by Dr. Abdolvand to examine its response to oven control. However, since obtaining that particular resonator is dependent on some of the Ph.D. students' work, if unforeseen circumstances keep the device from being finished before the project is complete, another resonator fabricated by Dr. Abdolvand's students will be used for testing this completed circuit project.

3.2 Microcontroller

The functionality of the microcontroller consists of the calculation of resistance from the passed in voltage and current values, the subsequent conversion to temperature using a lookup table containing predetermined relational resistance and temperature values, the display of the current resistance and temperature, and the calculation of the proposed new current to be passed back into the resonator. The calculated temperature and resistance values would be displayed to the LCD. To determine the most appropriate microcontroller for this application, this functionality was considered along with the technical experience of the software lead for this project. The ideal microcontroller would be simple yet accurate and cost-effective.

The selection of a microcontroller for this project was influenced by the following requirements: an appropriate number of data bits to store the incoming data, communication interface support for the transmission of the data, and a favor for accuracy with respect to speed. Microcontrollers are normally available in these different bit rates:

8-bit, 16-bit, and 32-bit. The bit rate alludes the number and size of the data lines on the microcontroller. The performance of the microcontroller and the bit size are proportional. For this project, the number of bits necessary to store the data is 12-14 bits. Therefore, the system was narrowed down to a 16-bit or 32-bit microcontroller. Due to the level of complexity of the project, the 16-bit microcontroller was selected for the application. The microcontroller's central processing unit (CPU) has a 16-bit reduced instruction set computer (RISC) architecture that is composed of 16 registers. The function for each of the 16 registers is shown in Figure 3.3.

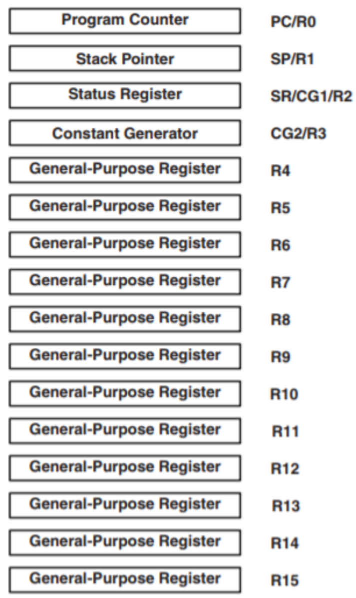


Figure 3.3: Register functions for the MSP430's CPU [5]

3.2.1 Communication Interface

Wired communications are used to establish connection between hardware components in a given system. For example, wired communication is utilized for the data transfer between a microcontroller and an LCD. The process of transmitting data one bit at a time in sequential order over a communication channel is referred to as serial communication. Serial communication will be used in this application for transmitting data between the microcontroller and other hardware components of the project. There are different options for standardized serial communication technologies including I^2C, SPI and UART. The former two are characterized as synchronous serial communication processes. These methods are potentially advantageous for interfacing the microcontroller to hardware subsystems. The latter is an asynchronous process that can be beneficial for sending and receiving data between external devices. The utilization of these serial communication technologies is discussed below.

3.2.1.1 I^2C

The Inter-Integrated Circuit (I^2C) is a serial communication protocol that utilizes a serial bus to advance communication between devices so that multiple masters are able to interface with multiple slaves. The method is purposed for short distance applications especially communication between integrated circuits within a common PCB. Further, it is often used at low speeds. An I^2C bus uses a 7-bit addressing scheme and two signals: Serial Data (SDA) and Serial Clock (SCL). The former is the data signal that transmits data bidirectionally between masters and slaves at the pace of the clock signal which is able to be controlled by the master or slave. The signals are transmitted using designated encoded messages and the desired slave's 7-bit address. Unlike SPI and UART, the I^2C signal bus lines are open-drain and maintain low signals so I^2C is an active-low method. I^2C is a relatively popular and robust method of serial communication and is supported by many devices.

3.2.1.2 SPI

The Serial Peripheral Interface (SPI) is a serial communication method also intended for small distance usage and characterized by its simplicity and multipurpose quality. It is considered to be a standard method of serial communication for embedded systems especially for LCD screens. Similar to I^2C, SPI implements the master and slave process. However, instead of two signals, it contains four: Slave Select (SS), Serial Clock (SCLK), Master In Slave Out (MISO) and Master Out Slave In (MOSI). The SPI method supports a single master and multiple slaves. The full-duplex design of SPI makes a more simple and faster alternative to I^2C. However, it requires more signal wires and slaves are not able to communicate with each other.

3.2.1.3 UART

The Universal Asynchronous Receiver/Transmitter (UART) is a hardware device that facilitates serial communication involving the configuration of data format and transmission speeds. The UART contains a parallel bus with multiple data lines and two serial lines: a receiver (RX) and a transmitter (TX). Furthermore, the UART is intended for converting data acquired through the parallel bus into serial data that is transmitted through the transmitter serial line, converting data acquired through the receiver serial line into data transmitted through the parallel bus, checking parity and handling interrupts. It supports simplex, half-duplex and full-duplex operations. Unlike the I^2C and SPI, the UART uses a single wire to transmit the clock and data, and it is a physical circuit in a microcontroller rather than a communication protocol.

3.2.1.4 Communication Interface Selection

In summary, the microcontroller must have at least one communication interface to transmit data from the resonator. The types of serial communications frequently used in combination with microcontrollers include serial peripheral interfaces (SPI), inter-integrated circuit buses (I^2C), and asynchronous serial communication. The former two are often used to exchange data between a microcontroller and other devices on the same PCB while the latter communicates with devices such as a PC. SPI and I^2C can be used to communicate with ADCs and DACs, sensors with digital output and other processors. Although SPI and I^2C comparable applications, the latter is a true bus devised to accommodate many devices and supports half-duplex transmission. On the other hand, SPI uses two lines for data transmission and data can be sent in either direction at the same time. SPI uses more wires and offers a simple and fast interface.

For this application, the SPI communication will be used to interface the LCD to the microcontroller. The LCD can receive data by configuring the SPI on the microcontroller. This will be achieved through initialization of the LCD screen and transmission of the data periodically to display the updated variables when the screen is refreshed. The resistance and subsequent current value would be transmitted between the microcontroller and resonator.

3.2.2 Accuracy

The objective for this project is to stabilize the temperature of resonator to achieve peak performance. Thus, there is an emphasis on accuracy rather than speed with regard to this application to ensure that high precision is executed. The selected microcontroller needs to have an appropriate ADC range and offer performance stability after each use.

3.2.3 MSP430 Series

The microcontroller families under consideration for this application were the Texas Instruments MSP430 and MSP432, the Microchip PIC24F, and the Silicon Labs Gecko. The Texas Instruments MSP430 family of microcontrollers offers an extensive variety of low-power consumption and integrated analog and digital devices designed for sensing and measurement applications. The MSP430 employs a 16-bit RISC CPU, 16-bit registers and constant generators that allow for efficient code implementation. Furthermore, its configurable features enable it to be a practical solution for many low-power applications. The MSP430 offers many series of this product that can be selected in terms of hardware interfaces, architecture, and memory specific to this project.

The advantages to this microcontroller include its low cost, online community support and familiarity in terms of language and Integrated Development Environment (IDE) due to its presence in the curriculum. On the other hand, the disadvantages include the programming intricacies associated with integrating an LCD and reading in analog inputs.

3.2.4 MSP432 Series

The Texas Instruments MSP432 microcontroller family offers a wide variety of low-power operation devices intended for high-performance applications. The MSP430 employs a 32-bit ARM Cortex-M4F processor, integrated precision ADC, and pin-to-pin scalability that allows for scalable code implementation. Further, its configurable features enable it to be an adaptable solution for many low-power applications. Similar to the MSP320, the MSP432 offers various series of this product that can be selected specific to this project.

3.2.5 PIC24F Series

The Microchip PIC24F microcontroller family presents a cost-effective, low-power and high-performance solution to embedded applications. The PIC24F features potential for precision time measurement, capacitive touch implementation and an integrated graphical or segmented display. In addition, it captures rich analog integration and serial communications that are necessary for this project. The broad product line ranges from low-power microcontrollers to high performance dual-core digital signal microcontrollers.

The advantages to this microcontroller include its ability to be incorporated with customer applications and provide low-cost and reduced time solutions. In addition, there is an emphasis on precision time measurement and the integration of displays. The disadvantages include its nonconfigurable complex features that are not necessary for this application.

3.2.6 Gecko Series

The Silicon Labs Gecko microcontroller family is designed for battery-operated applications and high performance and low-power systems. The Gecko is a 32-bit microcontroller that is based on the ARM Cortex-M3 core and offers an extensive variety of low-power and efficient devices designed for response and power-sensitive applications. The Gecko family offers many series of this product with specifications that can be configured specific to this project.

The advantages to this microcontroller include its mathematical capabilities, digital and analog peripherals and configurable LCD controller. The disadvantages include its other unnecessary features and the lack of background and community with this type of microcontroller.

3.2.7 Series Selection

The specifications and characteristics for the microcontrollers detailed above are shown in Table 3.1 in a more readable format.

Table 3.1 MCU Comparison

Feature	**MSP430**	**MSP432**	**PIC24F**	**Gecko**
Operating Voltage	1.8 V – 3.6 V	1.62 V to 3.7 V	2.0 V – 3.6 V	1.98 V – 3.8 V
Comm. Interfaces	UART, SPI, I^2C	UART, SPI	UART, SPI, I^2C	UART, SPI
Pin Count	24	40	26	32
Bit Count	16-bit	32-bit	16-bit	32-bit
Low Power	Yes	Yes	Yes	Yes
Power Consumption in Active Mode	330 µA/MHz	95 µA/MHz	300 µA/MHz	63-225 µA/MHz
Approx. Board Price	$14.99	$12.99	$4.99	$29.99

To select the family of microcontrollers that would be considered for this application, the aforementioned requirements along with the background of the software engineer was considered. The microcontroller chosen to be used in this project is the Texas Instruments

MSP430. This family of microcontrollers has peripheral sets designed for a range of different applications. These microcontrollers are simple 16-bit devices that encompass the desired communication interfaces in order to display the information to the user. The architecture allows for extended battery life for portable measurement applications such as this one. In addition, the selection of a familiar type of microcontroller allows for more time to be spent on additional features for the application.

Further, the two microcontrollers within this product line that had potential to be used in this project were the MSP430FG47x, the MSP430G2x and the MSP430F552x. The microcontrollers share the low-power feature and the ability to support a range of modules for added functionality such as wireless and displays. The advantages for the MSP430FG47x include its extremely low cost, 16-bit ADC and 12-bit DAC.

The advantages for the MSP430G2x include its low cost and simplicity. The amount of flash and RAM for this microcontroller is sufficient for measuring and displaying values. The disadvantages include its relatively lower ADC size than the other option and it has no extra pins for additional functionality.

On the other hand, the MSP430F552x is designed with the development of low-power, PC-connected applications in mind. The advantages to this microcontroller include increased Flash storage and RAM along with a 12-bit ADC for increased accuracy. The disadvantages include the unnecessary space and other capabilities. The specifications and characteristics for the microcontrollers detailed above are shown in Table 3.2 in an easier-to-read format.

Table 3.2: MSP430 Family Comparison

Feature	MSP430FG47x	MSP430G2x	MSP430F552x
Operating Voltage	1.8 V – 3.6 V	1.8 V – 3.6 V	1.8 V – 3.6 V
Temperature Range	-45 °C to 85 °C	-45 °C to 85 °C	-45 °C to 85 °C
Comm. Interfaces	UART, SPI, I^2C	UART, SPI, I^2C	UART, SPI, I^2C
Pin Count	48	20	63
Bit Count	16-bit	16-bit	16-bit
Additional features	Five low-power modes, digitally controlled oscillator	On-board buttons and LEDs, modules for added functionality	On-board emulation for programming and debugging
Board Price	$6.20	$9.99	$12.99

The specific model of microcontroller that was chosen for this application was the MSP430FG47x. This is an easy-to-use microcontroller that is perfect for low-power and cost-effective microcontroller applications. Its features include on-board emulation for programming and debugging purposes, on-board buttons for user interaction and pinouts that allow the board to support a variety of modules for implementations of wireless and displays. The other microcontroller options did not have enough pins and had more features and memory footprint than necessary for this application, respectively.

3.3 Liquid Crystal Display

A liquid crystal display (LCD) is a low-power solution for the MSP430 to provide a user interface to present textual information to the user. LCD technology will be used to present various readings from the resonator to the user. The LCD will receive input from the microcontroller. LCDs can be categorized into three classes: segmented LCDs, character-based LCDs and fully graphical LCDs. When selecting a compatible LCD screen for the microcontroller, the following factors were taken into consideration:

Manufacturer: The ideal LCD manufacturer brand will have a good reputation for quality and performance.

Structure: The structure of the LCD refers to the number of lines that the device can display as well as the number of characters per line. For instance, a 16x2 LCD screen is able to display a maximum of 16 characters per line and has up to two lines available. The application requires sufficient characters to project the temperature and resistance with respect to the resonator. This device will be used in the development phases. As a result, the objective is to choose a low-cost solution as component selections may change to meet application requirements in the future.

Current Draw: The desired current draw for the chosen LCD is as low as possible in standby and active modes. The aim is to conserve battery and remain a low-power application. In addition, the selection of a low-current device will enhance the battery life and the allocation of power to other resources.

Power: The selected LCD is required to run off of 5V in order to maintain the low-power quality of the application.

Interface: The LCD interface to the microcontroller is required to have compatibility with the MSP430FG47x.

Cost: The objective is to find the most cost-effective device that meets the given requirements since the project is in the prototyping phase. The selected LCD screen will preferably be simple and not have unnecessary features.

The options for the LCD screen were narrowed to the Lumex LCM-H01604DSF, Electronic Assembly EA 8081-A3N, TinSharp TC1602A-09T, Microtips Technology NMTC-S20200BMNHSGW-12, Gravitech LCD-20x4Y and Newhaven Display NHD-0216K1Z-FL-YBW.

3.3.1 Choosing Series for Comparison

The following table describes six different LCD screens that were selected for part comparison. To make the optimal selection, research was done for each component to assess its potential benefits and detriments.

On the website of Mouser (an electronic parts distributer), LCD screens were searched for and then filtered for LCD Character Display Modules. These are screens whose characters are made of seven different segments.

Table 3.3: Comparison of LCDs

Product	Manufacturer	Driver Voltage	Character Arrangement	Number of pins	Display Type	Price
LCM-H01604 DSF	Lumex	5V	16x4	16	STN, Transflective	$27.92
EA 8081-A3N	Electronic Assembly	5V	8x2	14	Neutral, Blu-Contrast, STN, Reflective	$16.97
TC1602 A-09T	TinSharp	5V	16x2	16	STN, Transmissive, Negative, Blue	$9.95
NMTC-S20200 BMNH SGW-12	Microtips Technology	4.5V	20x2	16	STN, Transmissive, Negative	$15.74
LCD-20x4Y	Gravitech	4.7V	20x4	16	STN yellow green	$14.35
NHD-0216K1 Z-FL-YBW	Newhaven Display	5V	16x2	16	STN yellow green, Transflective	$10.50

There are couple of different types of LCD screen that should be defined before the parts are looked at more closely. A STN (super twisted nematic) display has a 180°-270° light twist and uses birefringence to absorb and pass particular wavelengths. The “twisted” component of the name refers to the rotation of molecules from one plane to another). An STN LCD screen is a type of passive display, and its display is usually monochromatic. Transistors in a passive display usually activate an entire row or column rather than an individual pixel like an active display. Negative and neutral refer to the type of lighting contrast used between the characters and the background of the LCD.

Transmissive displays have a backlight that make the screen very readable indoors but can be hard to read in direct sunlight. Reflective displays have a mirror layer that provides the back light by reflecting the ambient light. However, it usually has lower resolution that makes it difficult to read in dim lighting. Transflective displays can both reflect and transmit light. This means that it generally uses a backlight in ambient indoor lighting as the primary backlight, but outdoors, it can reflect light, so that becomes the primary backlight source. This compromise of using both transmission and reflection also means that the display is typically dimmer in all lighting conditions [6].

The driver voltage is a measure of drain voltage. Since LCDs are transistor based (typically using field effect transistors), this is a measure of voltage at the drain of the FETs. This voltage is the minimum amount of voltage that must be applied to the transistor to turn on the LCD and thus can be considered the positive power requirement. If the LCD has too low of a driver voltage applied to it, the characters may seem too dimly lit, and if the driver voltage is too high, then segments that are off may also appear to be partially on.

The LCD screens compared here have an arrangement of the characters in a manner that resembles the following figure, with the number in the above table represented as AxB, where 'A' is the number of characters per row, and 'B' is the number of rows on the LCD.

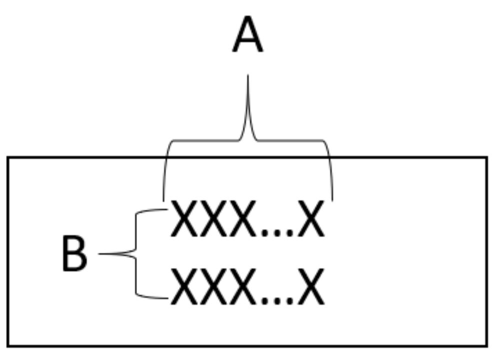


Figure 3.4: LCD arrangement of characters

Sometimes, the LCD screens need a separate controller to operate, although some LCD screens contain a controller. The controller has a clock, registers, voltage waveforms, and bias voltages that control the operation of the LCD per user programming. Some LCD controllers also need to be used in conjunction with a microcontroller.

3.3.2 LCM-H01604DSF

This LCD screen offers a 62 mm x 25.6 mm viewing screen with characters 4.75 mm x 2.95 mm. It is available in bulk packages and is RoHS compliant [7]. Out of the options brought into consideration for the project, it is one of only two options with a transflective display and the only one that can display 64 characters, which is the second highest number of characters of the options inspected here. However, it also has the most expensive price.

3.3.3 EA 8081-A3N

This LCD screen offers a package size of 40 mm x 20 mm with a letter height of 5.11mm and is interfaced via an 8-bit data bus [8]. It includes a controller, the ST7066. It is available in bulk packages and is RoHS compliant. Out of the options brought into consideration for the project, it is the only option with an 8x2 character display and the only one that can display 16 characters, which is the lowest number of characters of the options inspected here. However, it also has the second most expensive price.

3.3.4 TC1602A-09T

This LCD screen can be interfaced with a controller and is interfaced via a 4 or 8-bit parallel data bus [9]. It is available in COB (Chip on Board) packages and is RoHS compliant. Out of the options brought into consideration for the project, it is one of only two options with an 16x2 character display, which is just beneath the average number of characters of the options inspected here. However, it also has the least expensive price.

3.3.5 NMTC-S20200BMNHSGW-12

This LCD screen offers a viewing area of 82.2 mm x 18.2 mm and is interfaced via a 4-bit data bus [10]. It is available in bulk packages and is RoHS compliant. Out of the options brought into consideration for the project, it is the only option with an 20x2 character display and the only one that can display 40 characters. It also has the lowest driver voltage and a price point in the middle of the options.

3.3.6 LCD-20x4Y

This LCD screen offers a viewing area of 76 mm x 25.2 mm and can be interfaced with a controller and is interfaced via a 4 or 8-bit parallel data bus [11]. It is available in bulk packages and is RoHS compliant. Out of the options brought into consideration for the project, it is one of only two options with a 20x4 character display and the only one that can display 80 characters, which is the highest number of characters of the options inspected here. It also has a price point in the middle of the options.

3.3.7 NHD-0216K1Z-FL-YBW

This LCD screen offers a package size of 80 mm x 36 mm and is interfaced via an 8-bit data bus [12]. It contains a built-in controller, the ST7066. It is available in bulk packages and is RoHS compliant. Out of the options brought into consideration for the project, it is one of only two options with an 16x2 character display, which is just beneath the average number of characters of the options inspected here. It also has the second lowest price out of all the options.

3.3.8 Series Selection

In this project, the purpose of the LCD is to display the temperature and resistance readings. Since this information would ideally include numerical and alphabetical characters and symbols, the most appropriate type of display would be the character-based LCD. The compatible display selected to present the measurement readings is a 16x2 character-based LCD interface card. This will allow flexibility in displaying one measurement at a time with toggling functionality or both measurements simultaneously on the LCD in an easy-to-read format.

To select a 16x2 bit interface card, the communication interface, number of general I/O pins and cost were considered. As mentioned above, the desired communication interface to be used is the serial peripheral interface (SPI). The desired number of I/O pins necessary is approximately ten pins. Lastly, the ideal cost would be less than $20 for this display. After thorough consideration, the specific model display chosen is the TC1602A-09T. This LCD is relatively inexpensive as it remains below half of the allotted budget for this part. In addition, it provides a standard HD44780 LCD, with a 16x2 interface, which can be fully controlled with only 6 lines. This display is compatible to the chosen microcontroller for this project as seen in Figure 3.5 below, which is directly from the datasheet for the part.

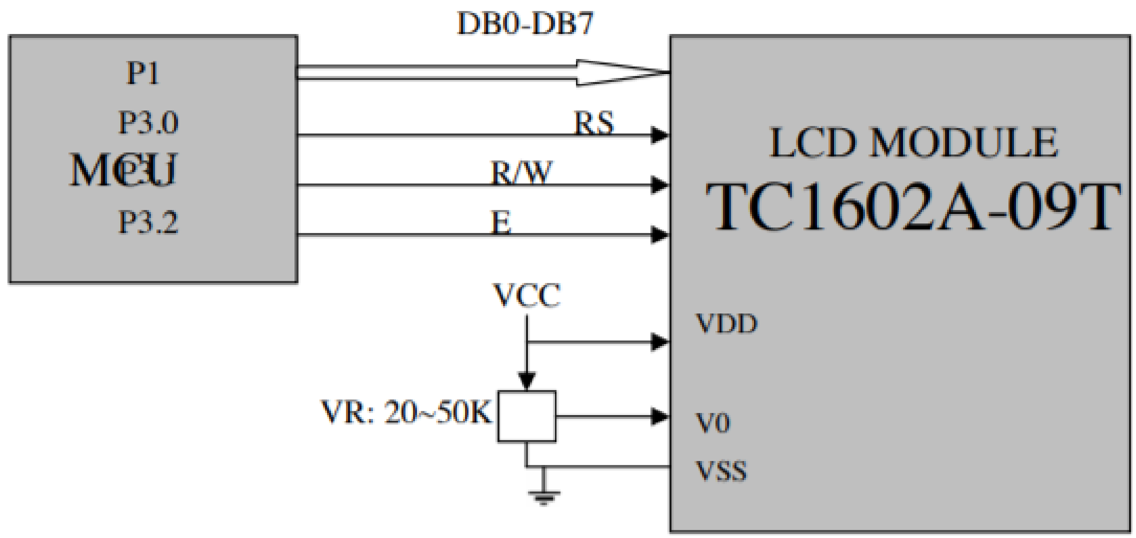


Figure 3.5: LCD inputs and connection to MCU [9]

3.4 First Order Filter

Filters are frequency-selective circuits that pass or attenuate certain input signals depending on a particular frequency band. For this application, first-order filters will be utilized to suppress undesired noise from the voltage signal of the resonator. First-order systems can be implemented as high-pass or low-pass filters, which is determined by the component voltage designated as the output. Further, the input voltage is then applied across the series combination of the elements. High-pass filters consist of a capacitor followed by a resistor and permit frequencies above the given cut-off frequency. Since high-pass filters are often used to remove distortion as a result of low frequency, it will be beneficial for ensuring the amplifier is low noise.

The following first order filter in Figure 3.6 is one that can be simply set up to work in conjunction with op-amps. For an instrumentation amplifier such as the one used in this project, the output voltage can be calculated according to the following equation, where R_I is the resistance inside the op-amp that helps determine gain:

$$(V^+ - V^-)\left(1 + \frac{R_I}{R_G}\right) + V_{REF} = V_O$$

If $V_{REF} = 0V$, then the following equation is true:

$$(V^+ - V^-)\left(1 + \frac{R_I}{R_G}\right) = V_O$$

Thus, the cutoff frequency can be calculated as:

$$f_{cutoff} = \frac{1}{2\pi(R_1 + R_2)C}$$

The capacitor for the filter will be ceramic, as this offers a low losses and high stability, which is especially useful for this application involving resonators. The following image depicts a first order filter as will be used for this project on the instrumentation amplifiers to filter noise from the circuit.

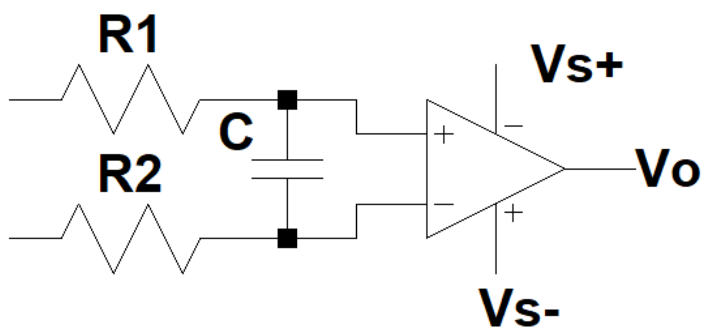


Figure 3.6: First order filter

3.5 Instrumentation Amplifiers

Instrumentation amplifiers use three op-amps to measure the voltage difference between the inputs. A sample of the inside of an instrumentation amplifier is shown below.

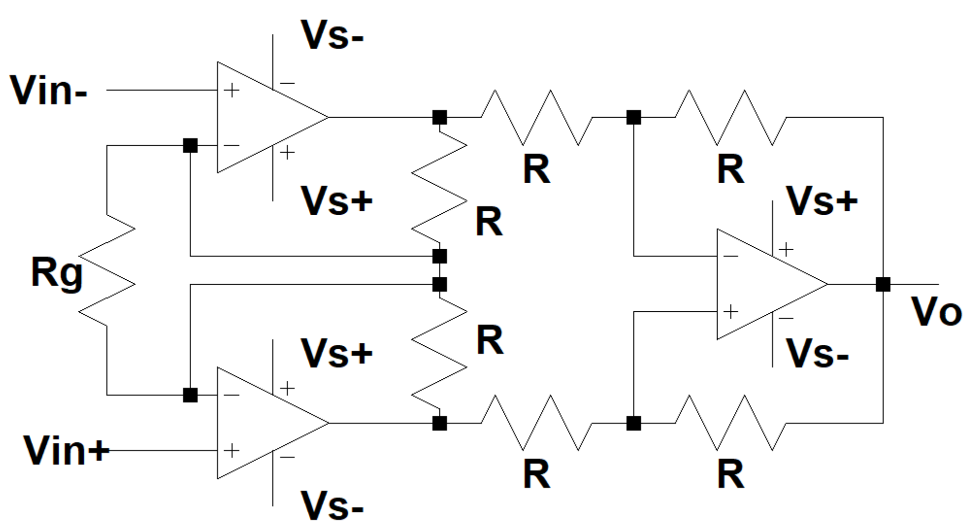


Figure 3.7: Sample instrumentation amplifier

In Figure 3.7, R_g determines the gain such that the following equation represents the behavior of the sample instrumentation amplifier:

$$V_o = (1 + \frac{2R}{R_g})(V_{in+} - V_{in-})$$

Each instrumentation amplifier contains a setup similar to Figure 3.7, although they all have varying gain equations that are determined by the resistors inside of the package. Many instrumentation amplifiers are also rated for single supply operation and thus have a voltage reference terminal to shift the range of operation.

Instrumentation amplifiers boast a higher input impedance and thus a lower input bias current than a differential amplifier. For a precision application such as this one, an instrumentation amplifier is a suitable choice. There are a variety of instrumentation amplifiers available in integrated circuit packages that will help fulfill the accuracy requirements for this project.

The output voltage can be calculated according to the following equation:

$$(V^{+} - V^{-})\left(1 + \frac{R_I}{R_G}\right) + V_{REF} = V_O$$

The inputs to the positive and negative terminals are V^+ and V^-, R_I is the internal resistance that is used to set the gain (shown above as 2R), and V_{REF} is the reference voltage. There are a few requirements from the datasheet that should be noted and obeyed: the reference input must be a low impedance connection. Even a few Ohms can degrade the performance of the instrumentation amplifier. Furthermore, 0.1µF tantalum bypass capacitors should be added in parallel to the power supplies.

3.5.1 Choosing Series for Comparison

Choosing an instrumentation amplifier involved visiting the websites of different manufacturers and distributers. For parts from Renesas, the website of Mouser, an electronic component distributer, was visited. Then, single channel precision instrumentation amplifiers were investigated. One with zero drift was chosen for closer comparison.

Analog Devices' website was searched for instrumentation amplifiers, and then the single channel products were considered. Options with precision or low noise were then located and selected for final consideration.

On Texas Instrument's website, options for instrumentation amplifiers include low noise, low power, low offset, and high voltage. Since precision is the main concern for this project, the low noise series was selected. From there, the search was narrowed to amplifiers with only one channel, since each instrumentation amplifier did not need to have more than one set of V^+ and V^- inputs. Three products were then selected from this list for comparison. A table comparing all of the inspected products has been created below, and the following sections describe some of the features of each type of instrumentation amplifier.

Table 3.4: Comparison of Instrumentation Amplifiers

Product	Manufacturer	CMRR (dB)	V_s Range (V)	Max. Gain	Input Offset (µV)	Gain Error (%)	Input Offset Drift (µV/°C)	Price
ISL28635	Renesas	138	±1.25 to ±2.75	1k	5	0.4	0.050	$6.36
AD8422	Analog Devices	128	±2.3 to ±18	1k	60	0.2	0.3	$2.10
AD8428	Analog Devices	140	±4 to ±18	2k	60	0.2	0.3	$6.60
INA828	Texas Instruments	140	±4.5 to ±36	1k	50	1	0.15	$2.15
INA128	Texas Instruments	120	±4.5 to ±36	100	50	2	1	$3.41
INA217	Texas Instruments	100	±4.5 to ±36	10k	250	0.7	0.5	$2.80

Common mode rejection ratio (CMRR) is a measure of the non-idealities of the amplifier. Since the positive and negative terminals are ideally equal, the ideal CMRR is infinity. However, since there are non-idealities in the device, the CMRR will have a finite value. Its numerical value is found in decibels using the following equation, where A_d is the differential gain (the gain when the positive and negative inputs are equal to inverses of each other) and A_{cm} is the common mode gain (the gain when the positive and negative inputs are equal to each other):

$$CMRR = 10\log\left[\frac{|A_d|^2}{|A_{cm}|^2}\right]$$

V_s is the supply voltage range that will allow the instrumentation amplifier to be operational. However, it should also be noted that the input range usually differs from these values and is dependent on the common mode rejection ratio and thus the internal transistor biasing of the device. Thus, the datasheet should be consulted to ascertain the input voltage range and if that will meet design requirements.

The maximum gain is set internally and externally. There is a resistor inside of the package that sets the maximum gain. Externally, the gain can be adjusted by a resistor on the specified terminals for the instrumentation amplifiers, but the performance of the amplifier will start to degrade once the gain exceeds the specified limits. However, the output of the amplifier cannot exceed a certain range of the input voltage as specified in the datasheet. Thus, the usefulness of a high gain is therefore dependent on the application.

Input offset voltage describes the voltage difference required between positive and negative input terminals to drive a zero output. The ideal value for this would be 0V, but instead the input offset voltage has a nonzero voltage because there will always be a slight mismatch between the terminals since ideal cases are unrealistic. The lower the input offset voltage is, the better.

The gain error refers to the expected closed loop gain versus the actual closed loop gain and relates back to the open-loop gain. Essentially, the value for the closed loop gain is calculated using the feedback ratio, β, using the following equation:

$$A_{CL} = \frac{1}{\frac{1}{A_{OL}} + \beta}$$

The ideal value of the open loop gain (A_{OL}) is infinity, but, it actually has a finite number associated with it that changes with frequency. Thus, there will be some error between the ideal and actual closed loop gain. Lower the percentage error between these two is better.

The input offset drift is a measure of how much the input offset voltage will change with temperature. The higher the input offset drift, the more sensitive the input offset voltage is to changes in surrounding temperature. Thus, a lower value for input offset drift is desirable.

3.5.2 ISL28635

This instrumentation amplifier has a programmable gain and offers rail-to-rail output. It is available in packages of 14-lead Thin Shrink Small Outline Packages (TSSOP). Its recommended uses are for sensing applications. Out of the options brought into consideration for the project, this one has the lowest input offset voltage and input offset drift and the second highest CMRR. However, it has the most limited supply voltage range and second most expensive price [13].

3.5.3 AD8422

This instrumentation amplifier also has a programmable gain and allows for outputs within ±0.2V of the supply voltages. It is available in packages of 8-lead (Micro Small Outline Packages) MSOP and Small Outline Integrated Packages (SOIC). Suggested applications include data acquisition, process controls, and transducer communications. This device has the lowest gain offset, second lowest input offset drift, and lowest price out of the ones compared here. However, its maximum gain is only 1000 [14].

3.5.4 AD8428

This instrumentation amplifier has a fixed gain and offers rail-to-rail output. It is available in packages of 8-lead (Micro Small Outline Packages) MSOP and Small Outline Integrated Packages (SOIC). Its recommended uses include sensing applications and monitoring for medical applications. Out of the options investigated for this project, it has the lowest gain error and highest CMRR, but also the most expensive price [15].

3.5.5 INA828

This instrumentation amplifier has a programmable gain of up to 1000 and output within ±0.15V of the supply voltages. This provides a wide operating range for the device. It is available in a package of 8-lead Small Outline Integrated Packages (SOIC). Its recommended uses include process controls, medical applications, and circuit breakers. Out of the options investigated for this project, it has the second lowest input offset drift and highest CMRR, but also the second highest gain error [16].

3.5.6 INA128

This instrumentation amplifier has a programmable gain of up to 1000 and output within ±0.9V of the supply voltages. This provides a good operating range for the device, although it is not as wide as some of the other options. It is available in a package of 8-lead Small Outline Integrated Package (SOIC). Its recommended uses include amplifier circuits, medical applications, and data acquisition [17]. Out of the options investigated for this project, it has the lowest gain but the second lowest CMRR and highest gain error and input offset drift. It's price point sits in the middle of the others (which are under $3 or above $6).

3.5.7 INA217

This instrumentation amplifier has a programmable gain of up to 10,000 and output within ±1.8V of the supply voltages. This provides a good operating range for the device, although that is the most restrictive limit on the output voltage of the parts compared here. It is available in a package of 16-lead Small Outline Integrated Package (SOIC) or 8-lead Dual In-line Package (DIP). Its recommended uses include microphone preamplification, receivers, and amplifier circuits [18]. Out of the options investigated for this project, it has the highest gain but the lowest CMRR, highest input offset voltage, and second highest input offset drift. It's price point sits closer to the middle but is on the lower end of the prices compared here.

3.5.8 Final Selection

The instrumentation amplifier chosen for this project was the INA828. It is a low noise, low power instrumentation amplifier that requires a supply voltage of ±4.5V to ±18V. The target supply voltage for this circuit is ±10V; thus, the INA828 is a suitable choice. The input voltage can range from $V^- + 2V$ to $V^+ - 2V$ before the performance will start to degrade. This instrumentation amplifier also offers the highest CMRR and has a reasonable price point.

3.5.9 Measure Voltage Across Resonator

To measure the voltage across the resonator, an instrumentation amplifier will be used. There should not be any gain on the input signal as the exact value is desired to be passed into the microcontroller. Thus, the voltage measured across the resonator should equal:

$$R_{res} * I_{res} = V_{res}$$

I_{res} will be set by the voltage input and controlled by the microcontroller.

3.5.10 Measure Voltage Across 10Ω Resistor

In a similar manner, the voltage passing through the 10Ω resistor will be equal to:

$$10\Omega * I_{res} = V_{1k\Omega}$$

This value will also be passed into the microcontroller. Even though the current could also be calculated using the initial VCCS, this will give a more accurate representation of what the current actually is. The microcontroller should then calculate the resistance value of the resonator using an equation similar to the following:

$$V_{res} * \frac{V_{10\Omega}}{10\Omega} = R_{res}$$

This resistor must be very precise (with a TCR very close to zero) so that the voltage measurement here will be accurate.

3.6 Voltage Reference

The voltage reference source will be used to provide a stable voltage for the proper biasing of the transistor. A variety of different options were considered from different distributers and manufacturers.

3.6.1 Choosing Series for Comparison

Choosing a voltage reference involved visiting the websites of different manufacturers and distributers. The website of Mouser, an electronic component distributer, was visited for comparison of different options. Then, precision shunt voltage references and adjustable precision shunt voltage references were examined closer, with a filter applied such that the maximum output voltage of the reference was at least 8V.

The search was then examined for voltage references that either had options for an 8V or higher breakdown voltage or that were adjustable to a value beyond 8V. Options from different manufacturers were inspected closer based on their options and compatibility to other products.

An additional search was made on Texas Instrument's website since comparing their products on Mouser was more difficult that doing so on the company's website. Options for power management and then voltage references were selected. From there, the search was narrowed to references with an 8.192V or higher option. Two products were then selected from this list for comparison. A table comparing all of the inspected products has been created below, and the following sections describe some of the features of each type of voltage reference.

Table 3.5: Comparison of Voltage References

Product	Manufacturer	Typical V_Z	Max. Cathode Current	Temperature Coefficient	Max. Power Dissipation	Price
LM4040	Texas Instruments	8.192V	15mA	100ppm/°C	306mW	$2.74
LM4050 A	Texas Instruments	8.192V	15mA	50ppm/°C	280mW	$2.68
LT1236	Analog Devices	10V	10mA	10ppm/°C	200mW	$6.55
LT1021	Analog Devices	10V	10mA	10ppm/°C	200mW	$6.57
ZHT431 FMTA	Diodes Incorporated	2.5V-20V	150mA	67ppm/°C	330mW	$1.35
TL431A	ON Semiconductor	1.24-18V	150mA	50ppm/°C	700mW	$0.70

A voltage reference is a circuit element that provides a certain voltage value for an indefinite amount of time. Shunt references usually have two (or three if the reference is adjustable) terminals and most are bandgap devices. They come in small packages and usually work over a range of currents and loads. Series references typically have three or more terminals and most consume a fixed current value and only conduct load current when the load requires it and are useful in designs with large loads. Zener diodes are similar to voltage references but have poorer current regulation than a reference source. Shunt references are compared here. The desired value for the voltage reference must exceed 8V; thus, only references with a stable value of greater than 8V or adjustable to a value greater than 8V were considered.

The cathode current describes the current that must pass through the current of the voltage reference for the device to work. It must be greater than a certain minimum value for the reference to turn on but less than a specified maximum value to prevent the device from having permanent damage. Often, the datasheet records an absolute maximum cathode current and recommended limit.

Here, temperature coefficient is used to predict the change in output voltage with respect to temperature changes. It is often a nonlinear value since it is the result of idealities internal to the device. However, the temperature coefficient is a reliable estimate about the device's response to external temperature stimuli. Lower values are better for this value.

The maximum power dissipation is the maximum amount of power that may be handled by the voltage reference. It is important for device functionality that this limit be adhered to. Depending on the application, a higher power dissipation value may be necessary, or it may be fine as a smaller value.

3.6.2 LM4040

This voltage reference has an option compared here with a set value of 8.192V. It is available in packages of TO-92 (Transistor Outline Package, Case Style 92), SOT-23 (Small Outline Transistor), and SC70 (also a type of Small Outline Transistor). Its recommended uses include energy management, instrumentation, and testing [19]. Out of the options brought into consideration for the project, this one has the third most expensive price point, although it is still less than half of the price of the most expensive parts. It also has the highest temperature coefficient and the third highest power dissipation.

3.6.3 LM4050

The constant 8.192V option voltage reference of the LM4050 family was also selected for comparison. It is available in packages of SOT-23 (Small Outline Transistor). Its recommended uses are for testing, energy management, and data acquisition systems [20]. Out of the options brought into consideration for the project, this one has an average price, temperature coefficient, power dissipation, and price.

3.6.4 LT1236

This voltage reference source's option has a constant gain of 10V. The other choice for this device is the 5V option. Since this is not high enough, it is not considered as a choice here. It is available in packages of N8 (PDIP or Plastic Dual Inline Package) or S8 (SO or Small Outline packages), which both have eight pins. Its recommended uses include DACs and ADCs as well as precision regulators and digital voltmeters [21]. Out of the options brought into consideration for the project, this one has the lowest temperature coefficient and input offset drift and the second highest CMRR. However, it had a high voltage value and the second most expensive price.

3.6.5 LT1021

This voltage reference source's option has a constant gain of 10V. The other choices for this device are the 5V and 7V options. This device is available in packages of 8-Lead TO-5 Transistor Case (TO-5) or in packages of N8 (PDIP or Plastic Dual Inline Package) or S8 (SO or Small Outline packages), which both have eight pins. Its recommended uses include navigation systems and DACs and ADCs as well as precision regulators [22]. Out of the options brought into consideration for the project, this one has the lowest temperature coefficient and power dissipation. However, it had a high voltage value and the most expensive price.

3.6.6 ZHT431FMTA

This voltage reference has a programmable value up to 20V, allowing the user to determine the reference voltage value. It is available in packages of SOT-23 (Small Outline Transistor), type DN, and has three pins. Its recommended uses are for replacing Zener diodes, particularly in circuits that require temperature stability. Out of the options brought into consideration for the project, this one has the highest cathode current and the second to lowest price. However, it also has the second highest power dissipation [23].

3.6.7 TL431A

This voltage reference has a programmable value up to 18V, allowing the user to determine the reference voltage value. It is available in packages of TO-92 (Transistor Outline Package, Case Style 92), eight lead (PDIP or Plastic Dual Inline Package), Small Outline Integrated Package (SOIC), or a Micro8 package, which is a special type of packaging unique to ON Semiconductor. Its recommended uses are for automotive applications [24]. Out of the options brought into consideration for the project, this one has the lowest price and the maximum cathode current. However, it also has the highest power dissipation.

3.6.8 Final Selection

The voltage reference to be used for this circuit is the LM4050. This reference was chosen because of its 8.2V option and ability to offer a stable voltage reference at a reasonable price. Since the voltage supply will be 10V, the reference must be less than that. However, to have a certain range of current, the following equation must be satisfied:

$$I_{setM}(R_{set} + R_{mea}) + V_{resM} + V_t = V_{refM}$$

V_t is the turnon voltage of the transistor, and V_{resM} is the maximum voltage across the resonator (occurs when the current I_{set} is at its highest value(referred to here as I_{setM}) due to V_i being at its highest value). Using estimated values from preliminary data, the proposed value of V_{refM} (the maximum voltage needed from the reference source) is estimated to be close to 8V. Thus, an 8.2V reference is a suitable choice for this circuit.

Furthermore, the voltage reference will be set up in a manner similar to the follow figure:

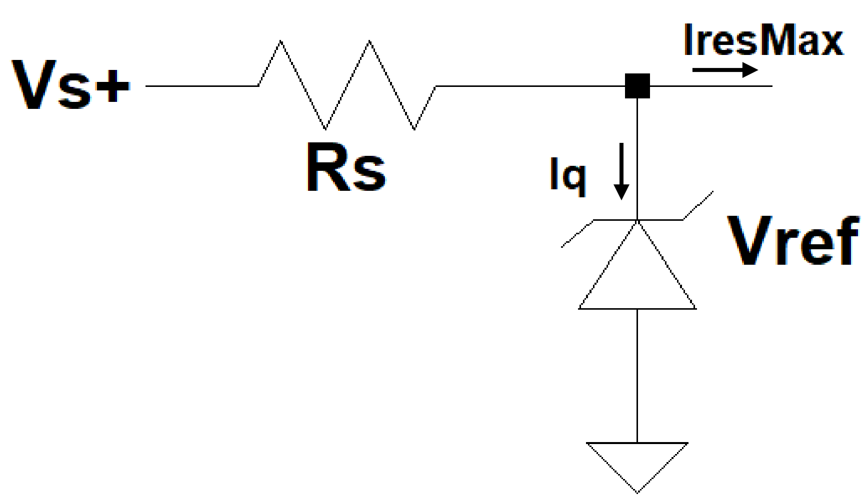


Figure 3.8: Sample voltage reference set up

For this setup, the following equation must be satisfied:

$$\frac{V_{s+} - V_{ref}}{R_s} = I_{resMax} + I_q$$

R_S should be determined such that current I_q is greater than the minimum operating current but less than the maximum reverse current. Thus, the follow equation must also be satisfied, with the bounds coming from the LM4050 datasheet:

$$I_{Rmin} < I_q < 15mA$$

Hence, the value for R_{refMin} must be chosen to meet both equations' standards to function as desired. The value for I_{Rmin} is given in the datasheet as 74µA [20].

3.7 Potentiometer

A potentiometer (variable resistor) will be added to the reference pin of the microcontroller to increase the range of voltage able to output by the DAC converter. This will allow for more stable current control, with parts per billion (ppb) accuracy, of the current passing through the resonator. This potentiometer will need to be manually tuned. Hence, the following sections detail different options for a manual potentiometer.

3.7.1 Choosing Series for Comparison

On the websites of Mouser and Digikey, both electronic component distributers, potentiometers (variable resistors) were filtered so that rotary potentiometers were displayed. A filter was then applied so that options with tolerances of 3% or less were displayed. The following table provides an overview of the choices that will be compared.

Table 3.6: Comparison of Potentiometers

Product	Manufacturer	Linearity	Tolerance	Resistance Range	Temperature Coefficient	Price
3549S-1BA-202A	Bourns	±0.2%	±3%	100Ω-20kΩ	50ppm/°C	$16.85
3590S-2-103L	Bourns	±0.25%	±5%	200Ω-10kΩ	50ppm/°C	$13.68
534B1502JC	Vishay Spectrol	±0.25%	±5%	100Ω-100kΩ	20ppm/°C	$16.18
882-MW22B-3-10K	ETI Systems	±0.5%	±5%	100Ω-100kΩ	20ppm/°C	$19.17
3386P-1-105	Bourns	±0.2%	±10%	10Ω-1MΩ	100ppm/°C	$2.34
774-284TBCF504A26A1	CTS Electronic Components	±2%	±10%	1kΩ-500kΩ	100ppm/°C	$12.61

There are many different types of potentiometers, many of which involve a mechanical wiper moving along a resistive material. The wiper's position determines the resistance seen by the circuit. The most common type is a rotary potentiometer, where the wiper moves along a circular path. All the potentiometers compared here are rotary with wipers except for the 3386P-1-105, which is a trimming potentiometer that requires a screwdriver to adjust the resistance value.

Wirewound potentiometers essentially have a wire coiled inside of them that the wiper slides along. The number of turns describes how many rotations it takes for the wiper to go from its minimum to maximum position. Thus, more turns will allow for higher precision.

Linearity is the relationship between the position of the wiper or knob and the resistance value. There are four types of linearity associated with potentiometers, but independent linearity is what is described in the datasheet. It is the maximum allowable deviation between any output value the corresponding linear value [25]. Thus, have a lower linearity percentage value is better since it means that the output and expected value align more closely.

The tolerance describes the percent error of the expected and actual resistance provided by the potentiometer. Accuracy, especially for this application, is crucial. Thus, lower tolerances are best. The resistance range describes what resistance values the potentiometer can provide.

Here, temperature coefficient is used to predict the change in the expected resistance value with respect to temperature changes. It is often a nonlinear value since it is the result of idealities internal to the device. However, the temperature coefficient is a reliable estimate about the device's response to external temperature stimuli. Lower coefficients are better for this value.

3.7.2 3549S-1BA-202A

This potentiometer has an adjustable resistance and can be tuned using wipers. It is available in packages of panel mounts. It is made resistive using a wirewound element with 10 turns and is recommended for use in harsh environments [26]. Out of the options brought into consideration for the project, this one has the tightest linearity and tolerance. However, it has a range of resistance that is below average for the parts chosen and a price point that is at the higher range of the options.

3.7.3 3590S-2-103L

This potentiometer has an adjustable resistance and can be tuned using wipers. It is available in bulk packages. It is made resistive using a wirewound 10 turn rotary metal [27]. Out of the options brought into consideration for the project, this one has nearly the tightest linearity and tolerance. However, it has a range of resistance that is less than average for the parts chosen and a price point that is in the middle of the options.

3.7.4 534B1502JC

This potentiometer has an adjustable resistance and can be tuned using wipers. It is available in bulk packages. It is made resistive using a wirewound 10 turn rotary metal

[28]. Out of the options brought into consideration for the project, this one has nearly the tightest linearity and tolerance. However, it has a range of resistance that is average for the parts chosen and a price point that is a few dollars above the median price of the options.

3.7.5 882-MW22B-3-10K

This potentiometer has an adjustable resistance and can be tuned using wipers. It is available in bulk packages. It is made resistive using a wirewound 3 turn rotary metal [29]. Out of the options brought into consideration for the project, this one has a median linearity, tolerance, and resistance range. However, it also has the lowest temperature coefficient and highest price of the options.

3.7.6 3386P-1-105

This through hole potentiometer has a trimmable resistance and can be tuned a knob or slot screwdriver. It is available in tape and reel, tube, and ammo packages. It has a single turn that adjusts the resistance [30]. Out of the options brought into consideration for the project, this one has the tightest linearity and but the loosest tolerance ratings. However, it has a range of resistance that is much higher than the other options, which helps balance out the higher tolerance and linearity, and a price point that is lower than the median price of the options. Trimming potentiometers have a much more limited lifespan (usually only about 200 turns) than rotary potentiometers, which usually can support up to a million uses.

3.7.7 774-284TBCF504A26A1

This potentiometer has an adjustable resistance and can be tuned using wipers. It is available in bulk packages. It is made resistive using a wirewound 1 turn rotary metal [31]. Out of the options brought into consideration for the project, this one has the highest linearity and tolerance and the narrowest resistance range. However, it also has the highest temperature coefficient and a medium price compared to the other options.

3.7.8 Final Selection

For this circuit, the 3590S-2-103L was chosen as the potentiometer. This option allows for tight tolerances, a long lifecycle, and wide resistance range while keeping the price comparatively low.

3.8 Power Supply

The main power supply will be used to provide the appropriate amount of voltage and current to every part within the circuit. The best option for main power supply for this application would be a bench power supply because they are very reliable, and the voltage value can be changed easily in case it needs to be adjusted during part testing. To determine the voltage needed for the main power supply, the voltage and current requirements for every hardware component of the circuit must first be analyzed. The main power supply must be greater or equal to the largest voltage needed within the circuit and will then be stepped down to a lower voltage using several DC to DC conversion to obtain the needed power for each component.

The Agilent E3631A triple output DC power supply will be used as the main power supply and will be lent by the sponsor for testing and demonstration. It has three separate terminals that can be used as DC voltage sources. The first terminal ranges from 0 to +6.18V, the second terminal ranges from 0 to +25.75V, and the third terminal ranges from 0 to -25.75V [32].

3.9 Software Tools

Team collaboration is essential to personal growth and delivering quality results, and there are many tools that have been developed to overcome challenges that often arise within a team environment. The tools utilized by this group are intended to foster effective team collaboration and enhance productivity. The tools are purposed to facilitate scheduling, tasking and organization while still maintaining the values of flexibility, responsibility and professionalism. The software tools used by this team are categorized into communication, development and documentation categories.

3.9.1 Communication

Effective communication is an integral skill for any type of team project. The use of collaborative platforms for messaging, asking questions and sharing documents allows for groups to remain on the same page during the development process. This team has chosen to use Microsoft OneDrive and GitHub to enhance communication and collaboration.

3.9.1.1 Microsoft OneDrive

OneDrive is a cloud storage solution developed by Microsoft. It has been adopted as a common place for sharing and storing documents used throughout the project lifecycle. This software allows for the creation of shared folders that are stored in an online drive. These folders are easily accessible from anywhere and are saved and updated in real-time given there is a source of internet. The folder shared between members of this team is used to store schematics and diagrams, documentation and relevant reading material.

In addition, OneDrive offers user-friendly and collaborative versions of Microsoft Office documents which are helpful in managing the budget, parts and meeting notes. These files created are then easily accessible and manageable within OneDrive. OneDrive is one of the most popular cloud storage services offered for free for students. It is familiar to all members of the team so it is natural that this provider is utilized for sharing information.

3.9.1.2 GitHub

GitHub is a development platform that allows for source code to be easily organized and shared between team members. It encourages code reviews, social coding and proper documentation through code review tools integrated into pull requests and project management-based features. Each repository will manage commits and show contributors and sources. Precise commits while allow for changes to be found in revision history quickly and detailed commits will contribute to the documentation throughout the software development phases.

In addition, GitHub provides free and private repositories for students. This will allow the project to remain within budget while taking advantage of quality and cost-effective tools.

Furthermore, GitHub is widely used by software developers in the industry. The practice of using this platform in a team environment will provide collaboration skills specific to software engineering that translate directly to the field.

3.9.2 Development

The development of a project from its initial concept to prototyping to a deliverable product is facilitated with the right tools. The integration of hardware and software components into a common project requires a great deal of collaboration. Development tools can be used to support debugging as well as design hardware components. The team has chosen to use Eagle and Code Composer Studio for hardware and software development, respectively.

3.9.2.1 Code Composer Studio

The choice of using a Texas Instruments microcontroller dictated the decision to adopt the Code Composer Studio IDE. Code Composer Studio is a derivation of the Eclipse IDE that integrates the advanced embedded debugging features from TI. It was created with the purpose of supporting applications for TI microcontrollers and embedded processors, and it includes a compiler designed for these embedded systems. In addition, it includes libraries and community forums for further support of the devices.

3.9.2.2 EAGLE

This project requires PCB design in order to implement specifically defined functionality and features. It can be difficult to find a program that is within budget and competent for the requirements of a project such as this one. Autodesk EAGLE is an electronic design automation software purposed for PCB design. It allows PCB designers to easily integrate schematic diagrams, components, and routing. Furthermore, it includes a user-friendly schematic editor, a powerful set of PCB layout tools and a support community. Autodesk provides a free three-year license for students.

3.9.3 Documentation

Documentation within a team can be challenging when the goal is to maintain consistent formatting and a unified quality. The documentation was initially created within a Google Doc for the sake of ease and real-time editing. However, with the preference for some of the more complex features offered by Microsoft Word, the team made a decision early on to transfer the documentation file to a shareable Microsoft Word document and use that from that point forward.

3.9.3.1 Microsoft Word

Microsoft Word is a work processing software and document creator that contains various tools for collaboration and editing. Microsoft Word contains development tools such as automatic and dynamic table of contents and lists, spelling and grammar checking and track changes which eliminates the overhead related to document organization and revisions. In addition, documents are able to be saved in OneDrive and shared with others. The ability to work together at the same time is integrated in Microsoft Office Online.

4. Design Requirement Specifications, Constraints, and Standards

This section discusses design requirement specifications, constraints, and standards relating to this project.

4.1 Design Requirement Specifications

Some design requirement specifications imposed by the project sponsor include a variety of operational requirements that the final design must satisfy.

Table 4.1: Design Requirement Specifications

Description	Value	Related Standards/Purpose
Project Cost Ceiling	$500	No Related Standards. Low cost without limiting project material quality. Designated by sponsor.
Time	31 Weeks (Total)	No Related Standards. Time limitation of Senior Design 1 and 2 courses to create, design, and build a working prototype of project.
Accuracy	Less than ~0.07Hz per 1°C of temperature change	No Related Standards. Required to ensure parts per billion (ppb) accuracy which is a requirement set by sponsor.
Operating Temperatures	Room temperature (approximately 23 °C) to greater than 85 °C (approximately 90 °C)	Standard Industrial Limit is 85 °C (per MIL-STD-810G). "Ensure the range of temperatures and rate of change of the test item's skin temperature is adequate to achieve the test profile. A typical range is -40°C to +85°C; "Necessary to go above this standard value to test the effects of temperature on resonance frequency." [33]
Resonant Frequency	70MHz with minimal deviation (+/ a few Hz)	No Related Standards. Specific to resonator used within project which is set by sponsor.

4.1.1 Frequency

In the final design, the resonator should resonate around 70MHz. This is the resonance frequency expected for the devices being fabricated by researchers in the Dynamic Microsystems Lab at the University of Central Florida working under the supervision of Dr. Abdolvand. By keeping the device operating at resonance frequency, the TCF can be minimized and device performance optimized. 60-80 MHz is typical for the resonance frequencies of these resonators; thus, 70MHz is the estimated value at which this device should operate.

4.1.2 Temperature

To optimize frequency stability, the temperature needs to be kept constant. The chosen method is to heat the circuit above industrial operating range and keep the temperature stable there. Thus, the resonator will need to operate above 85ºC, the limit of industrial operating range. The most critical part of this set point is the parts per billion (ppb) accuracy desired. An examination of resistance versus temperature characterization will determine how many degrees the device temperature can vary to obtain this accuracy.

4.1.3 Display

The design should incorporate a display to show operational resistance and temperature. This will allow the user to know the temperature and resistance and verify the stability and accuracy of the control loop. There will need to be a method of toggling cycling which one is displayed. This will need to be connected to the microcontroller for calculations and functionality.

4.1.4 Modes of Operation

The design should incorporate a stand-by mode when the control loop is not active to save power and demonstrate how the resonator operates without temperature control. In this stand-by mode, the microcontroller will still measure and display the resistance and temperature but will not seek to control the temperature. This contrasts with operational mode when the device is heated to its desired temperature and kept there using a control loop.

4.1.5 Accuracy

The operation of the device should exhibit parts per billion (ppb) accuracy. This can be calculated into a finite number using the following equation, where f is in Hertz, and ΔT represents temperature change:

$$f * \frac{1\, part\ per\ {}^{\circ}C\ of \Delta T}{10^{9}\ parts}$$

Thus, using the estimated resonance frequency of 70MHz, the resonance frequency should not change more than 0.07Hz per degree of temperature change.

4.2 Design Constraints

Design constraints must be explored from different perspectives. These are conditions imposed from outside sources that must be followed to have a successful design. This section explores the economic, manufacturability and sustainability constraints, the social, political and ethical constraints, and the environmental, health and safety considerations for this project.

4.2.1 Economic, Manufacturability, and Sustainability

The economic constraint imposed for this project is to stay underneath of $500. This includes all PCB design and components. Components are expected to consume less than half of this budget. Thus, the main part of this budget will be taken up by the printing of the PCB.

Manufacturing considerations must meet the needs of the PCB manufacturing. Routing and grounding and traces must be laid out properly so that the circuit will be efficient and function properly. Thickness of the PCB and number and layout of vias and pads must also be considered for optimal creation of the board. Furthermore, the resonator is being fabricated by Dr. Abdolvand's research group; thus, the exact resonator to be used will be dependent on project status of his students' work.

Sustainability constraints related to the robustness of the design will be considered for this project. First, the PCB must exhibit electrical sturdiness, such that it is able to handle the input powers appropriately and not short circuit or exceed component limits easily. A failsafe mode must also be created so that the resonator is protected from input voltage or current errors. Additionally, good wire bonding practices should be followed so that the circuit has stable connections and the resonator can be interchanged. The user should be able to connect another resonator to the circuit and still have it function properly.

4.2.2 Social, Political, and Ethical

For this design, there are no social or political constraints that apply to this project. The application for the product is limited since it is to be used as a method for members of the Dynamic Microsystems Lab group to optimize performance of their devices and is not at this time meant for wide-spread production.

The ethical constraints for this project relate to legal constraints as the device details of the resonator may not be published yet and/or are proprietary. Thus, care should be taken to use a resonator in the project but not disclose its operation or fabrication in detail.

4.2.3 Environmental, Health, and Safety

Environmental and health constraints should focus on using RoHS compliant parts. RoHS (Restriction of Hazardous Substances) was first a 2003 European Union directive that required heavy metals in products including lead and mercury to be replaced by safer alternatives in future products. Since then, the directive has been expanded and adopted by many countries globally [34]. RoHS parts are now readily available and used extensively. Thus, parts that comply with RoHS regulations should be used here to maximize environmental responsibility of the project.

To meet safety constraints, the circuit must be well grounded and offer stable power connections. Both constraints are important for keeping the user safe from electric shock.

4.3 Standards

This section analyzes standards that are relevant to this project.

4.3.1 Safety Standards

IEC 60950-1:

This international standard will be referenced for its general principles of safety regarding engineering design and operating equipment.

According to this standard, "Designers shall take into account not only normal operating conditions of the equipment but also likely fault conditions, consequential faults, foreseeable misuse and external influences such as temperature". These principles will be incorporated into the project design process in several ways. Datasheets for every component are analyzed and considered during the design process, such as maximum and minimum operating temperature, voltages, and current values [35].

This standard also discusses heat related hazards including "burns due to contact with hot accessible parts" and due to "degradation of insulation and of safety-critical components". The standard offers suggestions on how to avoid these hazards including avoiding having accessible high temperature components and providing markings to warn users when there are high temperature parts accessible [35].

4.3.2 Testing Standards

MIL-STD-810G:

This military-grade standard will be referenced for its test method standards regarding temperature and/or temperature change.

According to this standard, the typical industrial maximum product testing temperature is 85°C. The standard states to "Ensure the range of temperatures and rate of change of the test item's skin temperature is adequate to achieve the test profile". A typical range is -40°C to +85°C. However, to achieve the test profile of this specific resonance frequency, it is important to go above this standard temperature value. The standard also states "the rate of change may be as high as 4°C/min" which will be followed to ensure a slow transition between each temperature so that the resonance frequency can stabilize [33].

4.3.3 Operating Standards

The circuit should be able to function as specified in the industrial operating range. A primary advantage of this type of temperature control is that the circuit will not need to be under vacuum but can be used in conditions of atmospheric pressure.

4.3.4 Software Standards

This section discusses standards that relate to software used for this project.

4.3.4.1 Standard SystemC Language Reference Manual Standard

The *IEEE Std 1666-2011* is an IEEE standard used to define the SystemC library. SystemC is described as "an ANSI standard C++ class library for system and hardware design for use by designers and architects who need to address complex systems that are a hybrid between hardware and software" [36]. In addition, this standard provides a comprehensive SystemC class library definition such that the development of a SystemC application can rely solely using this reference. As the complexity of the technology industry advances, SystemC serves as a modeling language able to maintain these systems. Unlike traditional hardware description languages, SystemC has the capability "for modeling hardware and software together at multiple levels of abstraction" [36]. For this application, this standard will prove beneficial for the software component of this project. The microcontroller will implement the key functionality of the application using the C/C++ programming languages, so this library will serve as an important reference.

4.3.4.2 Design Impact of Standard SystemC Language

This standard will allow for complex systems to be able to be properly addressed should the opportunity arise. This project integrates software and hardware to achieve the specified objectives, so this library should allow for more elaborate code to be written. The standard will not inhibit the software design component of this project but serve as reference containing recommendations for improved code usage.

4.3.4.3 Software Testing Standard

ISO/IEC/IEEE 29119 is a software testing standard intended for use for any software development life cycle. The implementation of this standard will allow for "a high-quality approach to testing that can be communicated throughout the world" [37] to be used for the project. This standard serves as a reference to provide guidance regarding the proper performance of testing processes and approaches as well as software documentation. For this project, this standard will allow for the software component of this application to be up to industry standards and thoroughly tested.

There are five distinct parts associated with this standard: concepts and definitions, test processes, test documentation, test techniques and keyword-driven testing. The sections that are applicable to this application are the second, third and fourth parts. These are described in detail, with their relation to the project, below.

Part 2: Test Process

ISO/IEC/IEEE 29119-2 is a subsection of the software testing standard that delves into a test process model that is able to be used in the software testing phase of any software development cycle. Further, the standard elaborates on the software testing processes at the organizational, test management and dynamic test levels. Testing is often used to mitigate risk in software development; thus, the standard emphasizes "a risk-based approach to testing" [37]. This standard will be beneficial to this project since this is a widely-used approach to testing that prioritizes the most significant capabilities of the application.

The most applicable testing process for this application is the one at the organizational level. Within the documentation, the standard elaborates on the procedures for the development and evaluation of organizational test specifications and requirements. The team of three students for this project serves as the "organization" for this test procedure.

The process of test management can be defined in terms of three individual phases: test planning, test monitoring and test completion. Furthermore, existing test plans can be implemented independently or as an integrated whole. It is important to note that phases following this one follows the same structure regardless of the test plan implementation.

The actual testing stage will begin after the creation of a test plan. Test monitoring will serve as a phase dedicated to address any issues that emerge during the testing. The test plan should be modified to account for unexpected results should these types of issues arise. The final test plan should ensure that organization test guidelines are met, and other requirements are fully satisfied. The final stage, test completion, is comprised of test processes regarding design and implementation, environment, performance and documentation. The purpose of this stage is to ensure a comprehensive testing process and addressment of the results.

Part 3: Test Documentation

ISO/IEC/IEEE 29119-3 is a subsection of the software testing standard that elaborates on the appropriate methods of software testing documentation. The standard provides test documentation examples that correspond to the process levels of the second part of the standard. For the organizational test level, the standard defines the documentation for Test Policy and Organization Test Strategy.

The Test Policy "defines the objectives and principles of software testing to be applied within the organization" [37]. Further, it clarifies the goals to be achieved by testing, but it does not delve into specifics regarding the testing procedure. The given template provides a foundation for establishing and reviewing the organization's policy. For this application, this will allow for a comprehensible definition of the purpose of the testing to benefit all members of the project.

The Organization Test Strategy serves as "a technical document that provides guidelines on how testing should be carried out within the [project], i.e. how to achieve the objectives stated in the Test Policy" [37]. In other words, this document should define the 'how' for the 'why' defined in the template mentioned above. This template is generic and able to be conformed to different types of projects. The implementation of this strategy will allow for test sub-processes to be identified and defined. For this application, this portion of the standard will allow for concise and complete organization regarding the proposed testing procedure.

Part 4: Test Techniques

ISO/IEC/IEEE 29119-4 is a subsection of the software testing standard that defines various software test design techniques that can be utilized during the phase described in the second part of the standard. The techniques are described by three different test technique categories: specification-based, structure-based and experience-based. For these test techniques, the main source of information is what produces the expected results. Further, the combined use of these different techniques can improve the effectiveness of the testing procedure.

Specification-based test techniques use the test basis as the primary information source in the design of test cases. Here, the test basis includes requirements and specifications, models and user preferences. In addition, they emphasize the functional requirements of the system. The standard describes many different techniques that are able to be used to implement specification-based testing. It is the tester's responsibility to determine which technique is applicable to the application.

Structure-based test techniques use the test item structure as the main information source in test case design. The structure of the test item can include the source code or the model structure. Similarly, there are many different techniques to consider that can implement structure-based testing.

Experience-based test techniques the tester's knowledge and experience is utilized as the primary source of information for designing test cases. Due to the variance of the tester for this technique, there is no uniform definition for this type of test. The standard encourages error-guessing to be applied by the tester. This means that the tester uses their previous experience to design test cases where the input may cause failure.

4.3.4.4 Design Impact of Software Testing Standard

ISO/IEC/IEEE 29119 will allow for a comprehensive and standardized approach to testing the software component of the project. The universal intended use of this standard encourages this application to declare full conformance in using this standard. Further, the promotion to adopt a tailored conformance is appreciated since not all techniques apply to this application.

The organizational test process will support risk mitigation throughout the testing phase of this project. It will encourage all team members to participate in code review and evaluation which will ensure that hardware members are educated in the software component of the code. In addition, it will allow for different perspectives to be utilized in the achievement of an efficient program. Furthermore, it will ensure a complete evaluation for the application specifications and requirements and provide a structure to approaching the testing stage.

The organizational test documentation will prove beneficial in highlighting the important objectives that are to be gained through testing. This will allow for the reinforcement of the criteria needed to be met in the development stages of this project. In addition, it will promote a uniform expression of the testing process.

The test techniques provide a thorough method of testing by exploring different perspectives. The main source of information will be altered as the project moves through the different target test techniques. The specification-based test techniques will ensure that the key requirements for the project are being satisfied. The structure-based test techniques will confirm the robustness and efficiency of the code. The experience-based test techniques will utilize the personal experience and knowledge of the software engineer of the project to develop distinct test cases.

4.3.4.5 C Standard

ISO/IEC 9899 is an international standard that is used to define the C programming language. The standard "specifies the form and establishes the interpretation of programs written in the C programming language" [38] by defining its representation, semantic rules, representation of input and output and language syntax. Furthermore, the standard does not specify the implementations of applications of C programs. This standard is broken up into four subsections: the preliminary elements, the characteristics of C programming environments, the language syntax and the library facilities, all of which are relevant to the software component of this project. Although the standard, as a whole, is applicable to this application, the most pertinent information from the standard are elaborated below.

Most programming languages are defined by their terms, definitions, symbols, notations and concepts which are the basic building blocks of any programming language. This standard delves into the relevant terms and provides concise definitions and concrete examples for each of them. As a result, the programmer is able to know the effect of each line in the code. In addition, the standard addresses the behavior exhibited upon program execution. This will be beneficial during the testing and debugging stages of the project. The programmer will be able to further understand the interpretation of the code by the compiler. In addition, the standard establishes the syntax and notation specific to the C language. This will aid in the construction of code for the microcontroller.

4.3.4.6 Design Impact of C Standard

Ultimately, this standard will enable the software component of this application to implement efficient C programming to meet project requirements. The standard encourages the adoption of programming techniques that are essential to creating an effective solution. Furthermore, this standard will act as a reference for the syntax and notation for the microcontroller software. Although the standard does not serve as a teaching guide for the C programming language, the various elaborations on the different applications of identifiers, types and directives will be sufficient in supporting the development of technically correct software.

The software for this project will be heavily dependent on the control flow logic and arithmetic. The application will need to continuously loop and adjust its mode based on calculations done with the inputted parameters. Furthermore, the application will need to perform calculations in order to determine the correct modification for the ideal temperature, and this will rely on simple arithmetic calculations. The implementation of the control flow loop will determine the preciseness, accuracy and efficiency of the software, which are important objectives for this project. This standard will facilitate the development of effective software by addressing the correct usage of C syntax and notation. In addition, the standard will encourage the use of correct arithmetic techniques and the development of a sound program structure.

This software component of this project will rely heavily on this standard to implement industry standard C programming code. As a result, the software design utilizing the C programming language for this application will be significantly impacted by this standard.

5. Project Design

This section details aspects of the general project design. The design and hardware of individual components was discussed further in Section 2.5.

5.1 Analog Schematic Design in LTSpice XVII

Schematic design was started in LTSpice XVII since this was the most familiar program for the user. This is a program designed by Linear Technologies that allows for the response of the circuit to be tested. Additional parts, such as the LM4050, not found in the built-in library were installed to inspect the performance of the circuit. Simulations allowed for debugging from a hardware perspective. The simulation used for this project was a DC simulation. Different voltage values were input to see how the circuit responded. Figure 5.1 shows the initial schematic of the circuit design in LTSpice. This is an analog version of the circuit focusing on the control and precision of the current. Later, a digitally controlled schematic will be introduced that focuses on the integration of a microcontroller into the design. This portion is based on the work of one of the team members for her thesis for the Honors in the Major program at UCF [39]. The digitally controlled design is completely the work for this project.

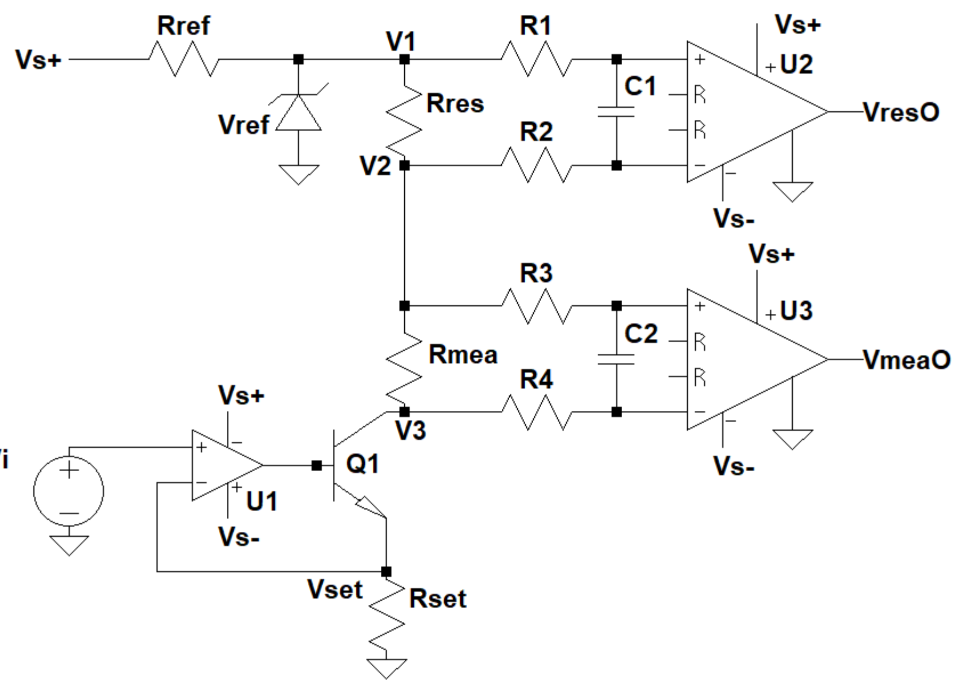


Figure 5.1: Schematic of analog circuit [39]

The schematic shown above operates in the following manner: A voltage (V_i) is input to U1, which is a precision op-amp, and this voltage is converted to a current by the following equation:

$$I_{set} = \frac{V_{set}}{R_{set}}$$

Because of the NPN transistor (Q1), the current through R_{mea} and R_{res} should now equal

$$\frac{\beta}{(\beta + 1)} I_{set} = I_{res} = I_{mea}$$

With a sufficiently large β, I_{set} approximately equals I_{res} and I_{mea}. The voltage reference source helps keep a stable voltage across the resonator so that there is always enough voltage to be dropped across R_{mea}, the resonator, the transistor, and R_{set}, even when the set current is operating at its highest allowable value.

The first order filters across the instrumentation amplifiers help filter noise from the resonator and R_{mea}. These instrumentation amplifiers measure the voltage across the resonator and R_{mea}. These values are then passed into the microcontroller and used to calculate the resistance of the resonator.

The values for each of the components listed above on the schematic are given in Table 5.1.

Table 5.1: Schematic Component Values

Name	**Value**
Vs+	10V
Vs-	-10V
U1	Op-amp
Rset	25Ω
Q1	NPN Transistor
Rmea	1Ω
Rres	N/A
R1	1.6kΩ
R2	1.6kΩ
R3	1.6kΩ
R4	1.6kΩ
C1	1µF
C2	1µF
Vref	8.2V
Rref	36Ω
U2	Instrumentation amplifier
U3	Instrumentation amplifier

After simulations with the following inputs, the outputs of the circuit were given in the simulation as Table 5.3 indicates. The only input variables for this design are the voltage input and the power supplies. All other values are calculated or attained via circuit function.

Table 5.2: Schematic Simulated Inputs

Measurement	Input Value
Vi	1.25V
V+	10V
V-	10V

Table 5.3: Schematic Simulated Operating Points

Measurement	Simulated Value	Expected Value	Unit
V1	8.22728	8.2V	V
V2	2.28662	2.2V	V
V3	1.79156	1.7V	V
Vset	1.25001	1.25V	V
VresO	5.94125	6V	V
VmeaO	0.495104	0.50	V
I(Rset)	0.0500006	0.050	A
I(Rmea)	0.0495055	0.050	A
I(Rres)	0.0495055	0.050	A

5.2 Digitally Controlled Schematic Design in LTSpice XVII

The process was repeated with the intention of integrating a microcontroller into the system. The following image depicts a digitally controlled version of the analog circuit.

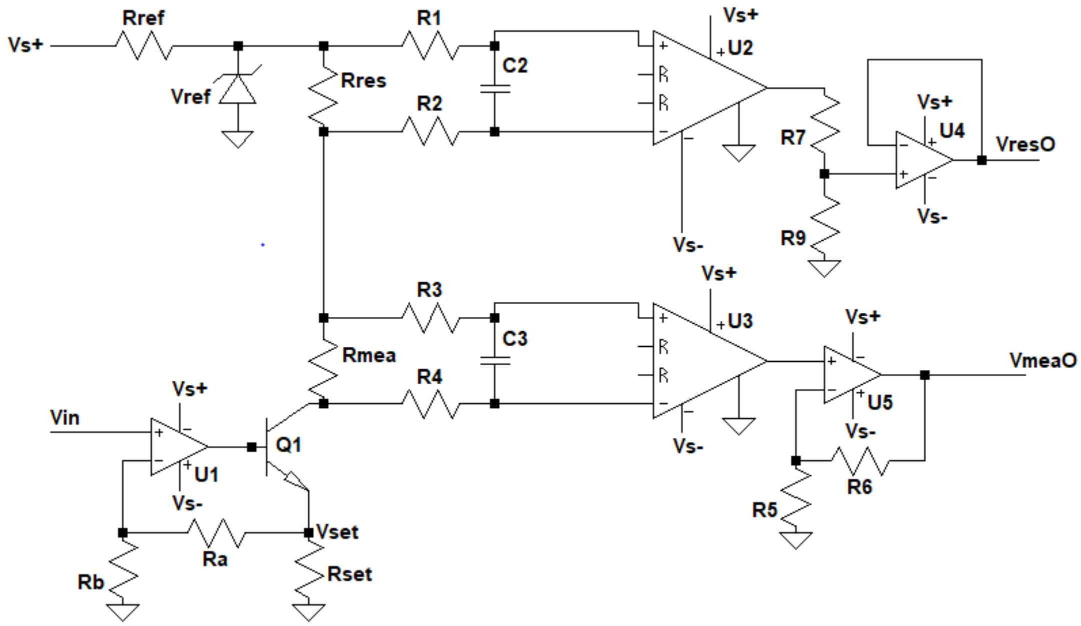


Figure 5.2: Schematic of digitally controlled circuit

The voltage input will now be set by the microcontroller, and the outputs for the measurements of the voltage across the resonator and resistor have been scaled using divider and amplifier circuits, respectively, so that the values measured at VresO and VmeaO will be within the voltage range for the microcontroller's analog to digital converter (0-3.3V). Using the values from Table 5.1, the following values can be added to complete the above schematic.

Table 5.4: Additional sample schematic values

Name	Value
R5	1kΩ
R6	5kΩ
R7	1kΩ
R9	1kΩ

After simulations with the following inputs, the outputs of the circuit were given in the simulation as Table 5.6 indicates. The only input variables for this design are the voltage input and the power supplies. All other values are calculated or attained via circuit function.

Table 5.5: Schematic Simulated Inputs

Measurement	Input Value
Vin	1.25V
V+	10V
V-	10V

Table 5.6: Digitally Controlled Schematic Simulated Operating Points

Measurement	Simulated Value	Expected Value	Unit
V1	8.22728	8.2V	V
V2	2.28662	2.2V	V
V3	1.79156	1.7V	V
Vset	1.25001	1.25V	V
VresO	2.98555	3V	V
VmeaO	2.97062	3V	V
I(Rset)	0.0500006	0.050	A
I(Rmea)	0.0495055	0.050	A
I(Rres)	0.0495055	0.050	A

Once the schematic had been built and tested in LTSpice, the design was then built in Autodesk Eagle. This is a PCB design software that was chosen because of its availability to students.

5.3 Schematic Design in Autodesk Eagle

The schematic design in Autodesk Eagle was initially done with components chosen to meet function. Later, optimal parts where chosen and inserted. Initially, however, the design focused on component layout and wiring. When the circuit is moved from being a schematic to a printed circuit board design, there are considerations that must be thought through. This section focuses on the inputs and outputs required for the finished project and the mediums by which this information will be translated.

The following figure shows an analog version schematic of the circuit. This focuses on the layout of the components and their connections before being connected to the microcontroller. The final digitally controlled circuit schematic in Eagle will be shown in Section 5.4.

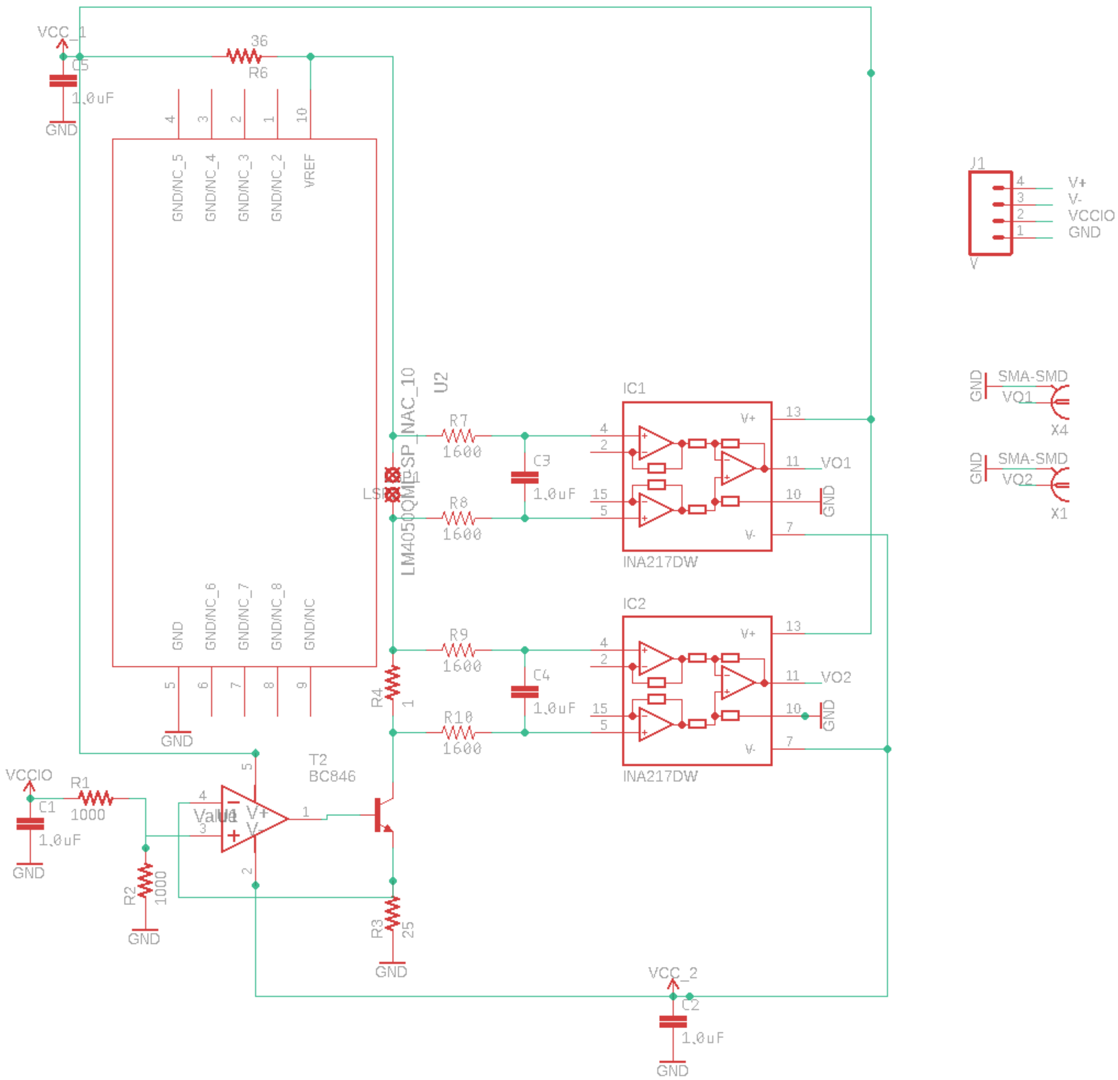


Figure 5.3: Schematic of analog circuit [39]

5.3.1 Circuit Inputs

For this schematic design, there are several inputs. The temperature setpoint is set by the user. The voltage input to the circuit, V_i, is controlled by the microcontroller's programmed PID control loop and is based on the temperature input. For powering the circuit, a bench supply will be used to pass in a single voltage that will be stepped down according to the necessary values. To connect this to the circuit, JST (Japan Solderless Terminal) connectors will be used.

5.3.2 Circuit Outputs

For this schematic design, the output will focus on the LCD screen. The resistance will be displayed on this display and serves as the user output. The analog portion of the circuit will feed its data into the microcontroller for digital control and programming.

5.3.3 Adding Libraries

While Eagle offers many different component footprints and symbols with the installation of the program, not all components needed for this design are automatically included in the Eagle software. Thus, components had to be added as necessary. This process is called adding libraries. Prior to the beginning of this step, Ultra Librarian by Accelerated Designs was installed to ensure smooth conversion of the downloaded model into a usable library in Eagle. Version 8.1.204 was used for this project.

To begin importing libraries to Eagle, first, the BXL file was located. This is a binary Xlator (BXL) file for the part that can be downloaded from many semiconductor company websites. Since Texas Instruments components were used for this project, it was simple to navigate this process. On their website page for the component to be used, there is a tab called "Quality and Packaging". On that page, there was a table named "CAD/CAE symbols". Then, there was a column called "CAD File (.bxl)", and the correct part's model was selected and downloaded.

Next, the BXL file was opened in Ultra Librarian software. All CAD packages that were not Eagle were unchecked. Then, "Export to Selected Tools" was clicked. The location of the exported files was verified, Ultra Librarian was closed, and Eagle was opened. "File", then "New", and then "Library" was selected. In the window that opens, "File" and then "Execute Script" was selected. The ".scr" file that was exported from the previous step was now opened and populated a new library.

Then, the part was closed. From the main control panel, "Library" and then "Open Library" was selected. The library file (.lbr) was then opened, and then "Library" and "Create Managed Library" was selected. Now, the downloaded part was listed as a component and available to be used in the schematic.

5.4 Final Schematic Design

This section discusses the final schematic design incorporating the hardware power requirements and DC-to-DC power conversions discussed within Section 5.5.

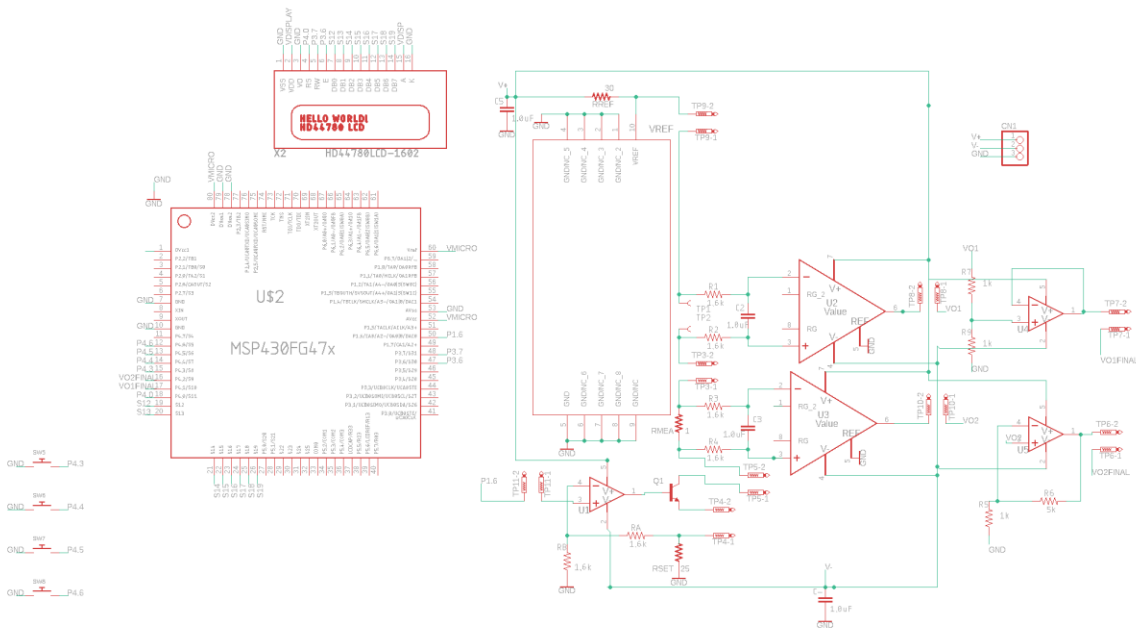


Figure 5.4: Final schematic design of digitally controlled circuit

On the left are the switches and the microcontroller. The top left component is the LCD display. On the far upper right are the inputs. The rest of the circuit follows the LTSpice schematic with the addition of test points for testing the first revision of the PCB. They are pairs of test points so that jumper wires can be soldered on to the board to test each portion of the circuit.

5.5 DC-to-DC Power Conversions

Several DC-to-DC conversions are needed to provide the components with the necessary operating voltage and current values.

5.5.1 Main Power Supply (10V) to Microcontroller (3.3V/2mA)

A conversion from the main power supply to the MSP430FG47x microcontroller power supply is designed using Texas Instrument's WEBENCH Power Designer software. The minimum input voltage is set to 9.99V and the maximum input voltage is set to 10.01V to account for the variation in the main power supply. The desired output voltage is set to 3.3V with a nominal current of 2mA. Texas Instrument's WEBENCH Power Designer software offers one hundred and sixty-eight total designs that achieve this DC-to-DC conversion; however, there are many factors to consider in choosing the best option for this project.

Beginning with efficiency, only twenty-five of the one hundred and sixty-eight designs are over 82% efficient, which is desired. A high efficiency ensures that less energy is being absorbed during the conversion process, which is undesirable because the wasted energy could be converted to thermal energy (heat) within the circuit. Next considering price, of those twenty-five designs only thirteen designs cost $3.80 or less for the bill of materials. Since there will be several DC-to-DC conversions and many other expenses to complete this project, low cost for this component is desirable. Next considering area, of those thirteen designs there are only four designs that occupy an area of 41 mm^2 or less. Since many other components will also be present within the PCB design, it is desirable that the DC-to-DC conversions do not occupy a significant amount of space. Lastly, considering the amount of parts, two of those four designs only have three components whereas the other designs have more. A design with less parts is desirable because it will make the PCB design easier when routing components and there will be less parts to test if something does not function properly. The last two designs are identical in area, part count, and efficiency. However, one is cheaper than the other, so that is the one that will be used for this design.

The EAGLE design files for the main power supply to microcontroller power supply conversion are downloaded and incorporated in the final PCB schematic design. The schematic design for the DC-to-DC conversion can be seen in Figure 5.5 below.

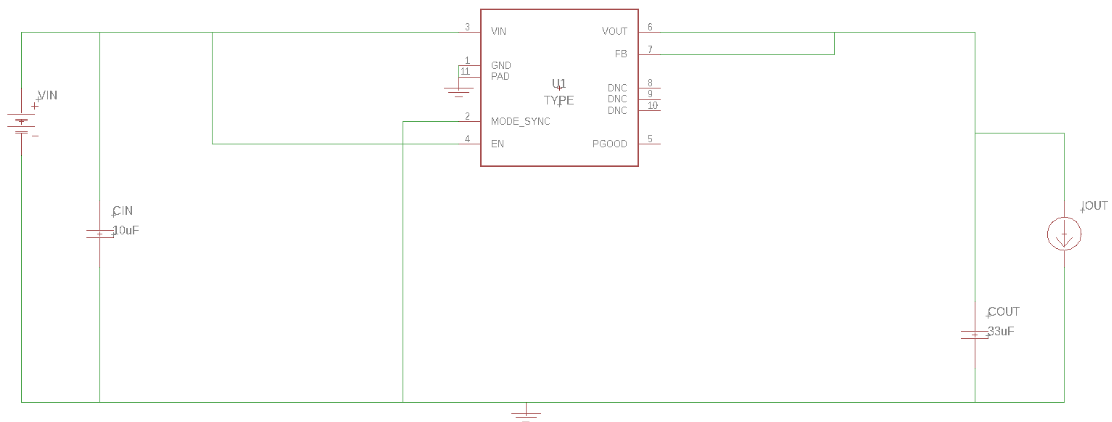


Figure 5.5: Main power supply to MSP430FG47x power supply conversion

This design utilizes the LMZM23600V3SILR step-down DC/DC power module. This device has a 3.3V fixed output voltage which will be used for the microcontroller's supply voltage. The datasheet of this component is examined further to ensure that the effect of input voltage and load resistance does not have a significant effect on the output voltage. From the datasheet for the LMZM23600V3SILR as shown in Figure 5.6, the output voltage is shown as a function of input voltage (expressed as different colored lines) and output current (which is related linearly to the output load resistance). No matter what input voltage (from 5V-36V) and output current (from 0A to 0.5A) are applied to the component, the output voltage only ranges from 3.3V-3.35V, which is very minimal and can be used as the supply voltage for the microcontroller [40].

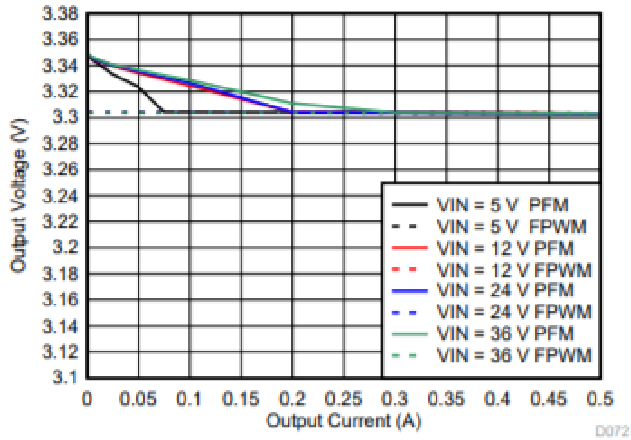


Figure 5.6: Line and load regulation for 3.3V output voltage [40]

5.5.2 Main Power Supply (10V) to LCD Screen (5V/35mA)

A conversion from the main power supply to the TC1602A-09T display power supply is designed using Texas Instrument's WEBENCH Power Designer software. The minimum input voltage is set to 9.99V and the maximum input voltage is set to 10.01V to account for the variation in the main power supply. The desired output voltage is set to 5V with a nominal current of 35mA. Texas Instrument's WEBENCH Power Designer software offers one hundred and seventy total designs that achieve this DC-to-DC conversion; however, there are many factors to consider in choosing the best option for this project.

Beginning with efficiency, only forty-four of the one hundred and sixty-eight designs are over 85.6% efficient which is desired. A high efficiency ensures that less energy is being absorbed during the conversion process, which is undesirable because the wasted energy could be converted to thermal energy (heat) within the circuit. Next considering price, of those forty-four designs only nineteen designs cost $1.90 or less for the bill of materials. Since there will be several DC-to-DC conversions and many other expenses to complete this project, low cost for this component is desirable. Next, considering area, of those nineteen designs, there are only five designs that occupy an area of 58 mm^2 or less. Since many other components will also be present within the PCB design, it is desirable that the DC-to-DC conversions do not occupy a significant amount of space. Lastly considering the amount of parts, one of those five designs only has five components whereas the other designs have more components. A design with less parts is desirable because it will make the PCB design easier when routing components and there will be less parts to test if something does not function properly.

The EAGLE design files for the main power supply to display power supply conversion are downloaded and incorporated in the final PCB schematic design. The schematic design for the DC-to-DC conversion can be seen in Figure 5.7 below.

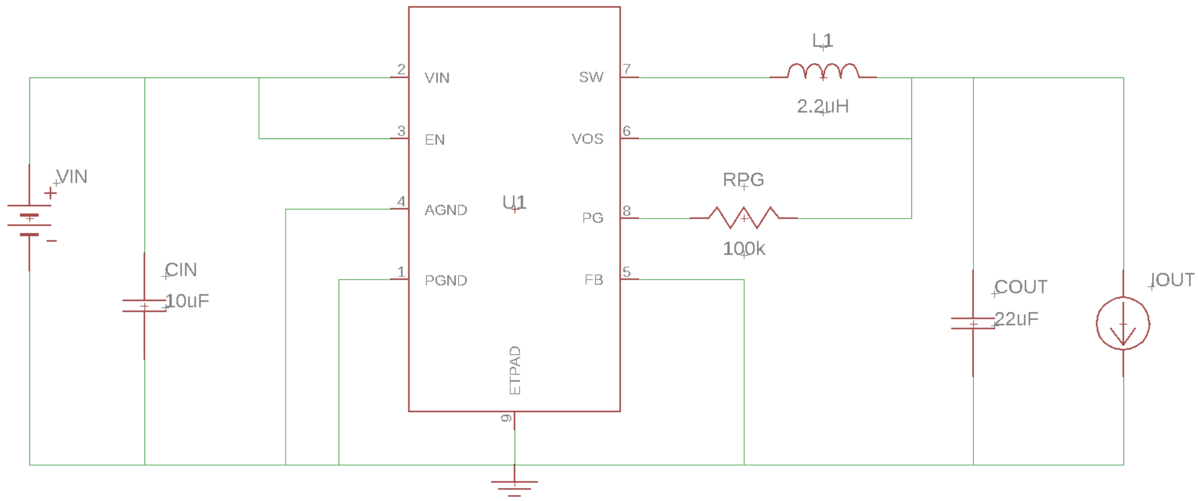


Figure 5.7: Schematic for main power supply for power supply conversion

This design utilizes the TPS62173DSGR step-down convertor. This component has a set output voltage of 5V which will be used to provide the supply voltage of 5V and supply current of 35mA to the LCD screen. The load and line regulation of this component should be analyzed to ensure acceptable operating conditions for the LCD screen. From the TPS6217X datasheet, it is found that the DC output voltage load regulation typically changes by 0.05% per amp and the DC output voltage line regulation changes by 0.02% per volt. This is a very minimal change and because the supply voltage for the display can range from a minimum of 4.8V to a maximum of 5.2V, this allows a change of 4% in output voltage from the typical 5V output. This component will not reach this level of error and therefore is an acceptable choice [41].

5.6 Printed Circuit Board Hardware Power Requirements

To begin the PCB schematic design process, the hardware's voltage and current requirements are first evaluated to ensure each component receives the required amount of power. Beginning with the main power supply, it will need to be stepped down several times to meet each component's power needs.

5.6.1 Main Power Supply

The target supply voltage for this design is ±10V. The main power supply source will be the Agilent E3631A triple output DC power supply. One terminal will be set to +10V and a second terminal will be set to -10V.

The change in output voltage for any line/load change (within the datasheet limitations) is given by <0.01% + 2 mV. Therefore, since the main power supply is set to ±10V then the change in voltage will range from ±9.997V to ±10.003V [32].

5.6.2 Microcontroller

The MSP430FG47x microcontroller is a great option regarding power because it has an ultra-low power consumption and has a low supply voltage range of 1.8V to 3.6V. The absolute maximum diode current at any device pin is ±2 mA.

The supply voltage is chosen to be 3.3V because that is when the processor frequency reaches its highest maximum value of 16 MHz. The active mode supply current (into the supply voltage terminal) is chosen based on the digitally controlled oscillator (DCO) frequency that is needed. The internal DCO within the MSP430FG47x microcontroller provides a fast turn-on clock source that stabilizes in less than 1 µs. A DCO frequency of approximately 8 MHz is desired for our application. Therefore, the supply current should be 2 mA. However, the supply current has a maximum value of approximately 4 mA, which would result in a DCO frequency of 16 MHz.

5.6.3 Display

The TC1602A-09T will be used as the LCD screen for this project. The supply voltage for this component ranges from a minimum of 4.8V and a maximum of 5.2V, where the typical supply voltage is 5V. In addition, the maximum supply current is 40mA and the typical is 35mA [9].

Therefore, the supply voltage is chosen to be 5V as it falls perfectly between the upper and lower bounds, and the supply current is chosen to be 35mA because that is the typical operating values for this display. With these supply voltages and currents considered, the voltage supply is chosen to be ±10V because it matches the main power supply voltage and remains within the operating range.

5.6.4 Instrumentation Amplifiers

The instrumentation amplifier chosen for this project was the INA828. It requires a supply voltage of ±2.25V to ±18V. However, the nominal voltage supply at 25°C (which is assumed to be the room temperature when operating this device) is ±15V. The input voltage can range from $V^- + 2V$ to $V^+ - 2V$. The absolute maximum current to the signal input terminals is 10mA. The typical current for the power supply is ±10 mA, and the maximum is ±12 mA [18].

Given these voltage and current ranges, the voltage supply is confirmed to be ±10V because it matches the main power supply voltage and remains within the operating range of the amplifier.

5.6.5 Voltage to Current Converter

The OPA828 operational amplifier will be used for the voltage to current converter. It has a supply voltage of ±2.25V to ±18V. The maximum current for the input signals is 10mA. The typical power supply current is 5.5mA while the maximum is 7.1mA with an ambient temperature between 0 °C to 85 °C [42]. Therefore, the voltage supply is confirmed to be ±10V because it parallels the main power supply voltage and remains within the operating range.

5.6.6 Power Supply for Components

After analyzing each component individually, the final chosen power supply values for each component can be found in Table 5.7.

Table 5.7: Voltage and Current for Components

Part Number	Power Supply Voltage	Power Supply Current	Maximum Supply Current
INA828	±10V	Expected ±10mA	±12mA
MSP430FG47x	3.3V	2mA	4mA
OPA828	±10V	Expected ±5.5mA	±6.5mA
TC1602A-09T	5V	35mA	40mA

5.7 Designing the PCB in Eagle

This section details the design of the printed circuit board.

5.7.1 Physical Component Layout

Once the schematic design has been completed, the board must also be designed, and the ideal electrical model must be converted into a useable circuit board. For this step, the components must be arranged so that the inputs and outputs are easily accessible.

5.7.2 Grounding

The grounding for the circuit will include a ground pour on the top and bottom layers. This ground pour layer is used for heat sinking and easier connections to ground. With these layers in place, grounding a component is as simple as creating a via and labeling it "GND". This will allow for connections to the ground pours.

5.7.3 Routing

The PCB routing for the project has been done in Autodesk EAGLE. This has been done using a combination of manual and auto routing. For this circuit, the design rule check (DRC) must be satisfied. This is a list of the design requirements for the manufacturer regarding details such as trace widths, clearances, and other similar details. The PCB manufacturer chosen for this project have DRC requirements listed in the following table.

Table 5.8: Manufacturer Routing Specifications

Requirement	Minimum Value (Mils)
Inner layer clearance	10”
Copper to edge of PCB	10” (outer layers), 15-20” (inner layers)
Pad size	10” (finished hole size for vias), 14” (finished hole size for component holes)
Annular ring	5” (for vias), 7” (for component holes)
Hole size	±5”
PCB Thickness	20” (2 or 4-Layer), 31” (6-Layer), 47” (8-Layer), 62” (10-layer)
Outer layer tolerances	±10”
Copper trace width	5” (1 oz. finished copper weight), 6” (2 oz. finished copper weight), 10” (3 oz. finished copper weight), 12” (4 oz. finished copper weight)
Air gap	±20% or ±2” (whichever is greater)
Soldermask swell	2.5” on each side
Slot width	31”
Tab rout spacing	100”
Silkscreen (legend) minimum line width	5”

The different requirements for the DRC are described here. “Inner layer clearance” is the distance that the through holes on multilayer boards have to be cleared from internal planes. “Copper to edge of PCB” is the distance between the edge of the board and the pattern. Traces connecting different layers use vias, usually placed on a pad. “Pad size” is the minimum width of the pad for the vias. The “Annular ring” is the area of a pad that surrounds a via. “Hole size” is the width of the finished holes in the board after the drills have been completed. “PCB thickness” describes in the minimum thickness of the overall board.

“Outer layer tolerances” refer to the routing of the circuit. The “Copper trace width” describes the distance between two conductive copper lines that will carry current in the circuit. This value is related to the weight of the copper. “Air gap” is the distance between two adjacent traces. This distance is crucial because of the heat generated by current in a circuit. If two traces are too close together and heat up because of the current passing through them, an issue may arise. “Soldermask swell” refers to the width by which solder mask will expand. The solder mask is a protective layer places on the bare PCB to prevent accidental bridging during assembly. It is what gives the board its distinct color. This will expand throughout production. Thus, it is important to have extra clearance around the edges of the PCB so that there is no shorting when this occurs.

“Slot width” describes the width of a slot in the board. Although similar to an elongated hole, these have a different minimum width because different tools are often used for drilling the slots (mills for slots and drills for tools). Some slots will be covered in metal, while others will be left unmetallized. “Tab rout spacing” is the distance between different tab routes. These can be used when the manufacturer combines several boards together into a panel of PCBs. Essentially, the designs can then be snapped apart using perforated breakaway tabs instead of having to cut them. This is an alternative to creating smaller boards entirely separately. The “Silkscreen (legend)” refers to the layer on top of the PCB that designates the different part numbers.

The following image depicts the finished routing of the printed circuit board for the project. This configuration satisfies the DRC and has ground pours. The red is a copper ground pour on the top layer, and the blue is a bottom layer copper ground pour.

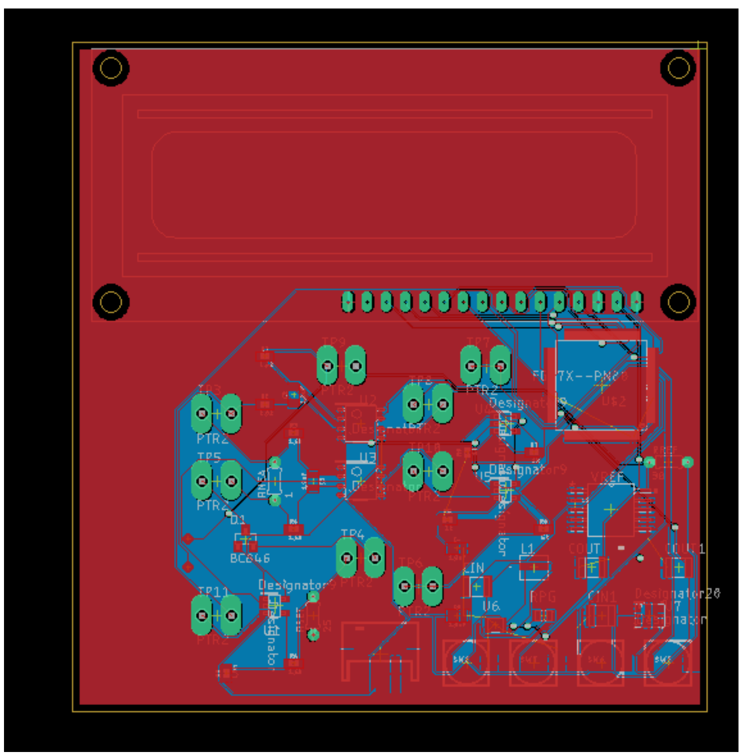


Figure 5.8: Current routed PCB image

5.7.4 Gerber File Generation

Gerber files are used by the PCB manufacturers to create the board. They can be generated in the design software (EAGLE). There are a number of checks and ways to minimize errors for this step. Some common errors, as listed by the PCB manufacturers, are a missing aperture list (that describes which tool to use), a missing excellon drill file (that describes hole size and location), a missing tool list (which details what tools are needed for drilling holes), insufficient annular ring (when a drill pierces a copper layer because of improper annular ring specifications), an insufficient copper trace width, and insufficient inner clearances. Minimizing these errors will aid in the creation of the PCB.

The following figure shows an image of what the Gerber file is creating in its "Top Copper" file. This is the copper layer that will be on the top layer of the PCB.

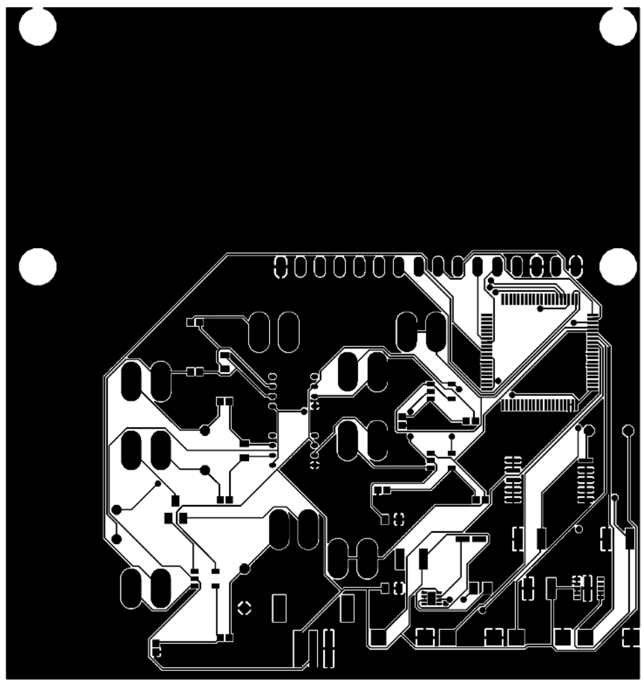


Figure 5.9: Top Copper layer from Gerber File

The following steps can be used to generate the Gerber files in EAGLE. First, the board file (.brd) was opened in EAGLE. Next, "File" and then "Cam Processor…" was selected. After that, the "Local CAM jobs" button in the top left was selected, and the "gerb274x.cam" was selected. Next, "Process Job" was selected. The exports the Gerber files.

The following steps can be used to generate the drill file in EAGLE. First, the board file (.brd) was opened in EAGLE. Next, "File" and then "Cam Processor…" was selected. After that, the "Local CAM jobs" button in the top left was selected, and the "excellon.cam" was selected. Next, "Process Job" was selected. The exports the Gerber files.

Table 5.9: Gerber Files

File name	Purpose
SD1.dri	Drill station info file
SD1.sol	Bottom copper
SD1.pls	Bottom silkscreen
SD1.sts	Bottom soldermask
SD1.cmp	Top copper
SD1.plc	Top silkscreen
SD1.stc	Top soldermask
SD1.drd	NC Drill
SD1.gpi	Photoplotter info file

5.8 PID Controller Design

This section discusses the design of the PID Controller which will be used to stabilize the resistance and thus the current passing through the resonator, which will in turn stabilize the temperature of the resonator.

5.8.1 Damping and Maximum Overshoot

The maximum overshoot is defined as the peak value of the system response and can be expressed as a percentage relative to the steady-state value. It is preferred for this project's applications that there is no maximum overshoot since a specific rise time is needed to ensure that the resonator's temperature is stabilizing as it approaches steady state. In order to eliminate any overshoot, the controller will be modeled as a second-order critically damped response. This will allow the system to reach steady state faster than an overdamped response, and not have the overshoot of an underdamped response.

In the s-domain, a second-order system transfer function is given by:

$$C(s) = K * \frac{{\omega_n}^2}{s^2 + 2\zeta\omega_n s + {\omega_n}^2}$$

which gives two poles: $p_{1,2} = -\zeta\omega_n \pm \omega_n \sqrt{\zeta^2 - 1}$

Where K is the system's desired steady state value, ζ is the damping ratio, and ω_n is the undamped natural frequency. For an undamped system $\zeta = 0$, for an underdamped system $0 < \zeta < 1$, for a critically damped system $\zeta = 1$, and for an overdamped system $\zeta > 1$. Since the controller will be designed as a critically damped system therefore $\zeta = 1$, then the transfer function can be simplified to:

$$C(s) = \frac{Y(s)}{R(s)} = K * \frac{{\omega_n}^2}{(s + \omega_n)^2}$$

which gives two poles: $p_{1,2} = -\omega_n$

Where Y(s) is the output of the system in the s-domain and R(s) is the input of the system in the s-domain because the transfer function of a system is defined as the system's output divided by the system's input.

5.8.2 Step Response

The system's step response can be analyzed by setting the system's input equal to the unit step function in the s-domain, $R(s) = 1/s$. The equation above can then be solved for the system's step response:

$$Y(s) = K * \frac{1}{s} * \frac{{\omega_n}^2}{(s + \omega_n)^2}$$

The inverse Laplace transform of the critically damped system's step response can be taken to analyze in the time domain:

$$y(t) = K * (1 - e^{-\omega_n t} - \omega_n t e^{-\omega_n t}), t > 0$$

During resonator testing, it will be found that the resonator's temperature stabilizes at a rate of R °C/s where R is a constant determined by resonator resistor and temperature correlation from testing data. The natural damping frequency (ω_n) will be chosen such that the slope of the transient response does not exceed R °C/s. The system's transient response slope for a unit step input can be found by taking the derivative of the step response function. This is given by:

$$\frac{d\, y(t)}{dt} = slope\ (\frac{°\mathrm{C}}{\mathrm{s}}) = K{\omega_n}^2 t e^{-\omega_n t}$$

The maximum slope can be found by taking the derivative of the slope equation (or the second derivative of the step response function) and setting it equal to zero as shown below:

$$\frac{d\, slope}{dt}\ (\frac{°\mathrm{C}}{s^2}) = K{\omega_n}^2 e^{-\omega_n t} - K{\omega_n}^3 t e^{-\omega_n t} = 0$$

The equation is solved for t which will be the time when the maximum slope occurs. The maximum slope occurs at $t = \frac{1}{\omega_n}$ seconds. This value is then substituted into the slope equation which gives:

$$Max\ Slope\ (\frac{°\mathrm{C}}{\mathrm{s}}) = K\omega_n e^{-1} \approx 0.36787944117 * K\omega_n$$

For example, if the desired steady state value (K) is set to 1 and the natural damping frequency (ω_n) is set to 1 then the maximum slope would be 0.36787944117 °C/s. This maximum slope would occur at $t = \frac{1}{\omega_n} = 1$ second when $\omega_n = 1$. The rate of temperature change/slope (°C/s) vs. time when $K = \omega_n = 1$ is shown in Figure 5.10 below.

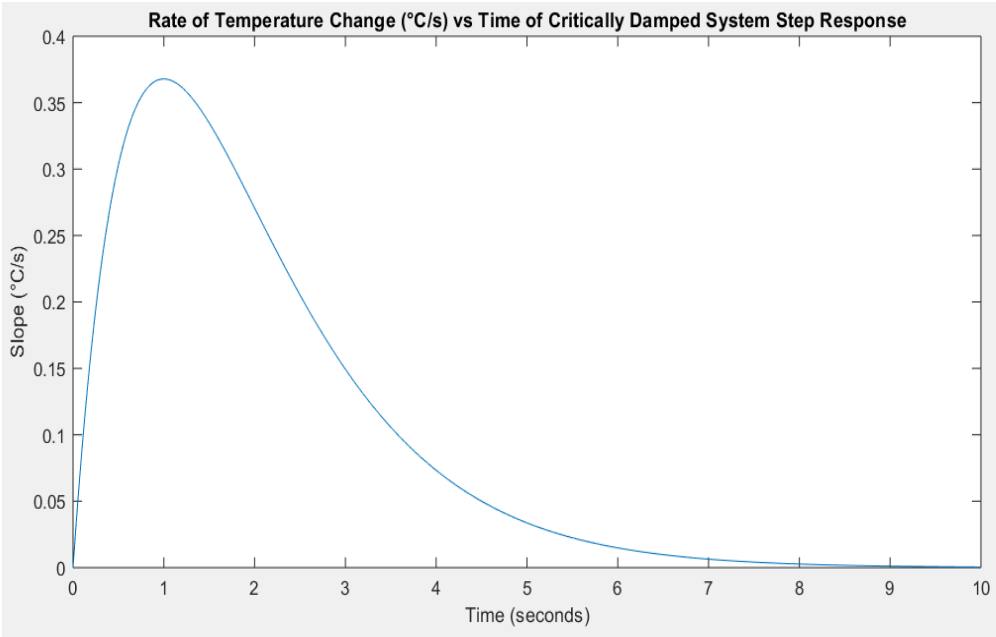


Figure 5.10: Rate of temperature change vs. time

For the same system, the system output of temperature (°C) vs. time when $K = \omega_n = 1$ can be seen in Figure 5.11 below.

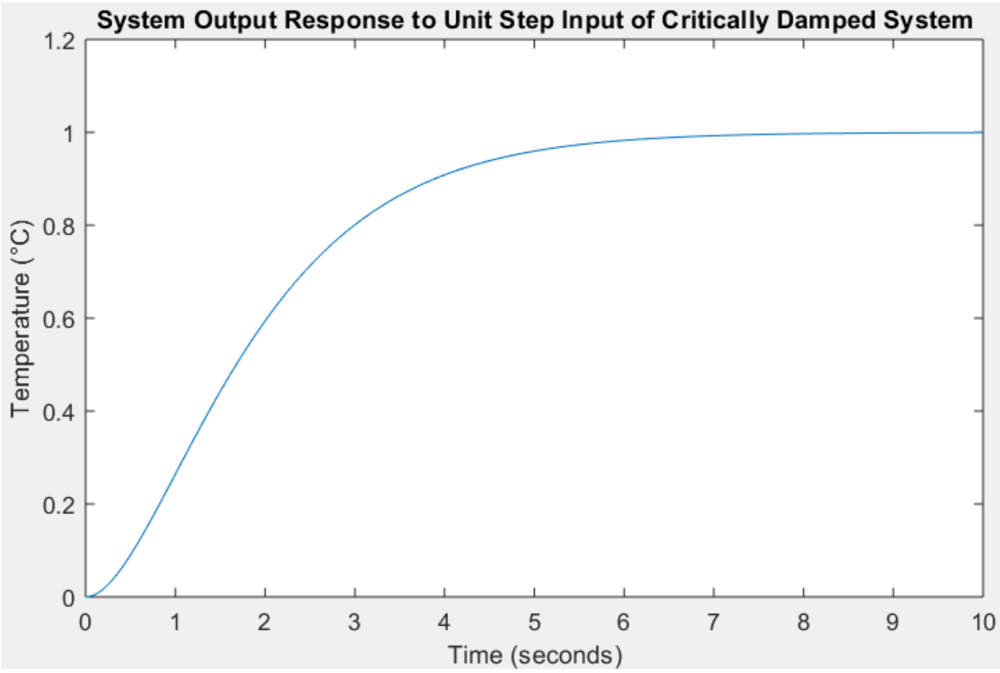


Figure 5.11: System output vs. time

In order to achieve the fastest transient response possible while ensuring that the resonator's rate of temperature change does not exceed R °C/s, the maximum slope is set equal to R °C/s:

$$Max\ Slope\ \left(\frac{°C}{s}\right) = R\ (\frac{°C}{s}) = K\omega_n e^{-1} \approx 0.36787944117 * K\omega_n$$

This equation is then solved for the natural damping frequency (ω_n):

$$\omega_n = \frac{R}{Ke^{-1}} \approx \frac{R}{0.36787944117K}$$

5.8.3 Rise Time

The rise time for a critically damped system is defined as the amount of time that it takes for the system response to go from 5% to 95% of the steady-state value. Typically, a short rise time is desired; however, for this project's applications, it is important that the resonator has time to stabilize in temperature and does not heat up too quickly.

The time that it takes for the system to go to 5% and to 95% of the steady-state value (K) can be found by setting the critically damped system's step response in the time domain equal to 0.05*K or 0.95*K as shown in the equations below:

Time of 5% steady-state value $\quad K * (1 - e^{-\omega_n t} - \omega_n t e^{-\omega_n t}) = 0.05 * K$

Time of 95% steady-state value $\quad K * (1 - e^{-\omega_n t} - \omega_n t e^{-\omega_n t}) = 0.95 * K$

Where $\omega_n = \frac{R}{Ke^{-1}}$.

Both equations are then solved for time (t). The rise time is calculated by subtracting the amount of time that it takes for the system to go to 5% of the steady-state value (K) from the amount of time that it takes for the system to go to 95% of the steady-state value (K).

For example, if the desired steady state value (K) is set to 1 and the natural damping frequency (ω_n) is set to 1 as shown in Section 5.8.2 then the above equations can be solved to find the system's rise time. The amount of time that it takes the system to rise to 0.05 °C (5% of the steady-state value K) is found to be approximately 0.355247755 seconds. The amount of time that it takes the system to rise to 0.95 °C (95% of the steady-state value K) is found to be approximately 4.743915436 seconds. The rise time is calculated by subtracting the amount of time that it takes for the system to go to 5% of the steady-state value (0.355247755 seconds) from the amount of time that it takes for the system to go to 95% of the steady-state value (4.743915436 seconds) which gives a total rise time of 4.388667681 seconds for this example.

An alternative method for finding the amount of time that it takes for the system response to go to 5% or to 95% of the steady-state value is to plot the system's step response in MATLAB as shown in Figure 5.11. The 'Data Cursor' tool can be used to determine the amount of time it takes for the system to reach 5% and 95% of the steady-state value. However, the 'Data Cursor' tool is not as accurate as the calculated method. Figure 5.12 below shows how to find when the system reached 95% of the steady-state value from the previous example using the alternative method. The closest data point is at 94.96% of the steady-state value which is achieved at 4.734 seconds. Compared to the calculated solution of 4.743915436 seconds found using the equations, there is only a 0.21% error in using this alternative, faster method. This approach should be repeated to find when the system reached 5% of the steady-state value in order to solve for the rise time of the system.

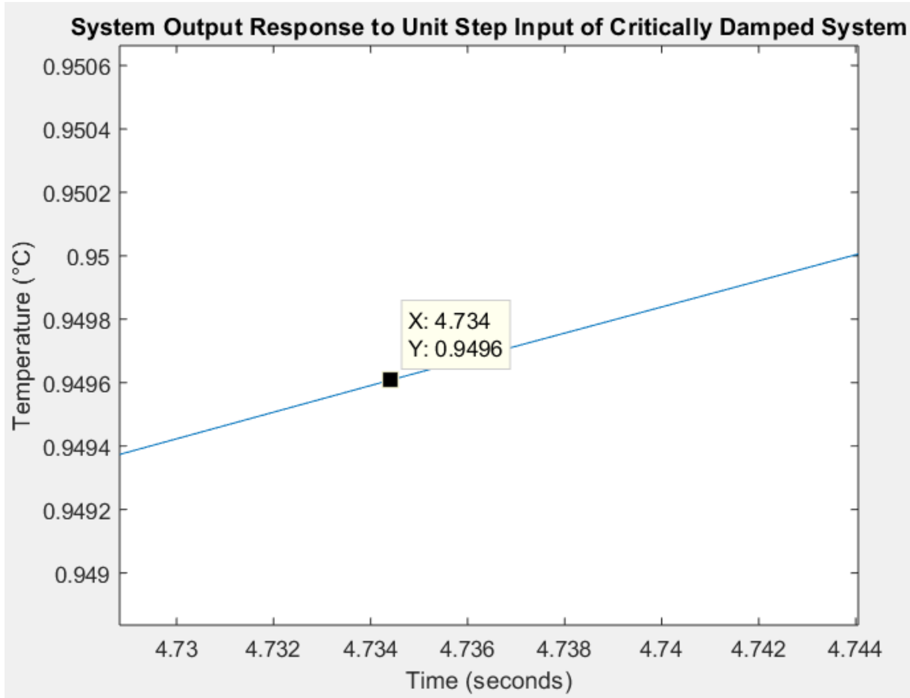


Figure 5.12: Time to 95% steady-state value using MATLAB 'Data Cursor'

There is another, even more precise, alternative method for finding the amount of time that it takes for the system response to go to 5% or to 95% of the steady-state value also using a plot of the system's step response in MATLAB. It involves using the 'Zoom in' tool to magnify the plot to the precise position where the system output reaches the desired value of 5% or 95% of the steady-state value. Continue to 'Zoom in' on the position until the desired precision is achieved. An example of this is shown in Figure 5.13 below to find the amount of time it takes for the system to reach 95% of the steady-state value for the previous example. The amount of time is found by comparing where the desired output temperature is achieved on the x-axis, as shown in red in Figure 5.13. Using this approach, the amount of time that it takes the system to reach 95% of the steady-state value is found to be 4.743915436 seconds for this example which exactly matches the calculated value. This approach should be repeated to find when the system reached 5% of the steady-state value in order to solve for the rise time of the system. Although this method offers more precision than the previous method, it takes more time than the previous method but less time than the calculation method.

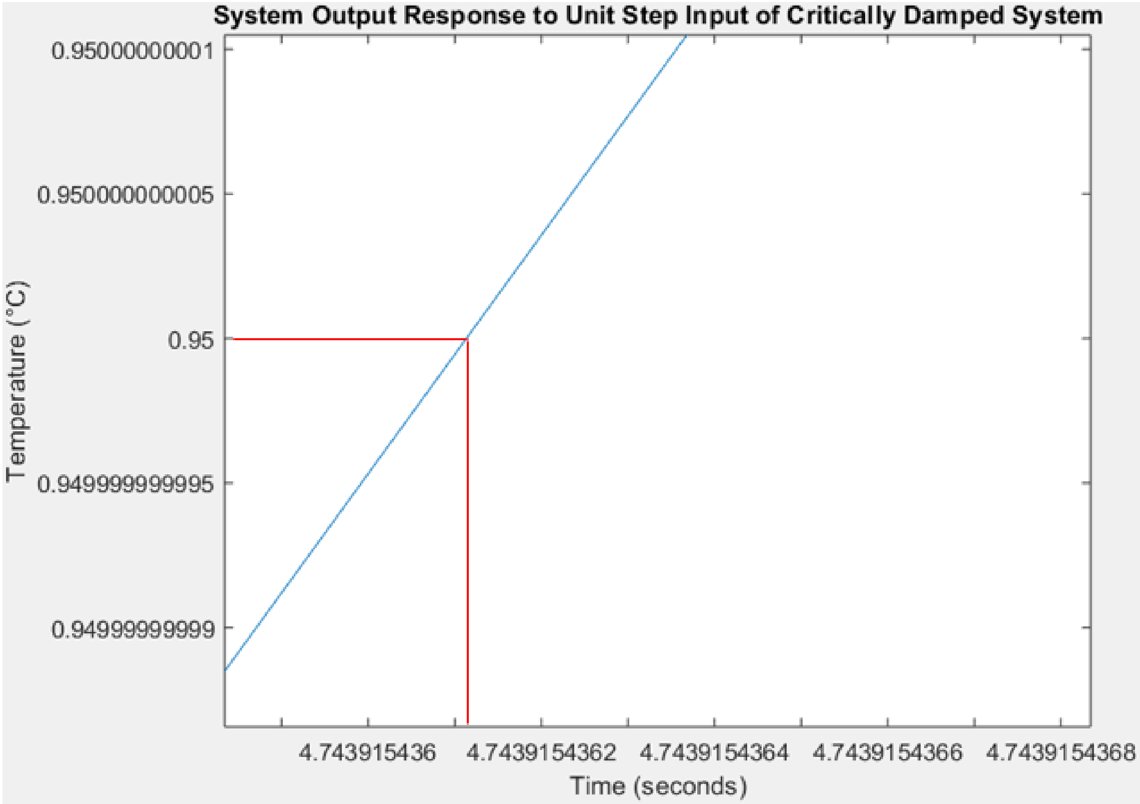


Figure 5.13: Time to 95% steady-state value using MATLAB 'Zoom in'

This method achieves the same results as the calculated values but is much quicker and minimizes human calculation error. Either of the alternative methods can be used to find approximations for the time that it takes for the system response to go to 5% and to 95% of the steady-state value. Once both values are found through calculations or using MATLAB, the rise time is calculated by subtracting the amount of time that it takes for the system to go to 5% of the steady-state value from the amount of time that it takes for the system to go to 95% of the steady-state value.

5.8.4 Settling Time

The settling time is defined as the amount of time that it takes for the system response to reach and stay within a specified error band of the steady-state value. The temperature steady-state value is very important for this project, so an error band of ±1% will be used. Since the controller is designed as a critically damped system, the step response does not oscillate. Therefore, once the 99% of the steady-state value (K) is initially reached, then the system response will stay between 99-100% of the steady-state value thereafter.

The time that is takes for the system to go to 99% of the steady-state value (K)can be found by setting the critically damped system's step response in the time domain equal to 0.99*K as shown in the equation below where $\omega_n = \frac{R}{Ke^{-1}}$:

$$K * (1 - e^{-\omega_n t} - \omega_n t e^{-\omega_n t}) = 0.99 * K$$

The equation is then solved for time (t) and that value is the settling time (in seconds).

For example, if the desired steady state value (K) is set to 1 and the natural damping frequency (ω_n) is set to 1 as shown in Section 5.8.2 then the above equation can be solved to find the system's settling time. The amount of time that it takes the system to rise to 0.99 °C (99% of the steady-state value K) is found to be approximately 6.63840792 seconds. Therefore, the settling time for the system is 6.63840792 seconds for this example.

An alternative method for finding the amount of time that it takes for the system response to reach 99% of the steady-state value is to plot the system's step response in MATLAB as shown in Figure 5.11: System output vs. time. The 'Data Cursor' tool can be used as previously mentioned in Section 5.8.3. However, the 'Data Cursor' tool is not as accurate as the calculated method which is shown in Figure 5.14 below which shows how to find when the system reached 99% of the steady-state value from the previous example using this method. The closest datapoint is at 98.99% of the steady-state value which is achieved at 6.628 seconds. Compared to the calculated solution of 6.63840792 seconds found using the equations, there is only a 0.16% error in using this alternative, faster method.

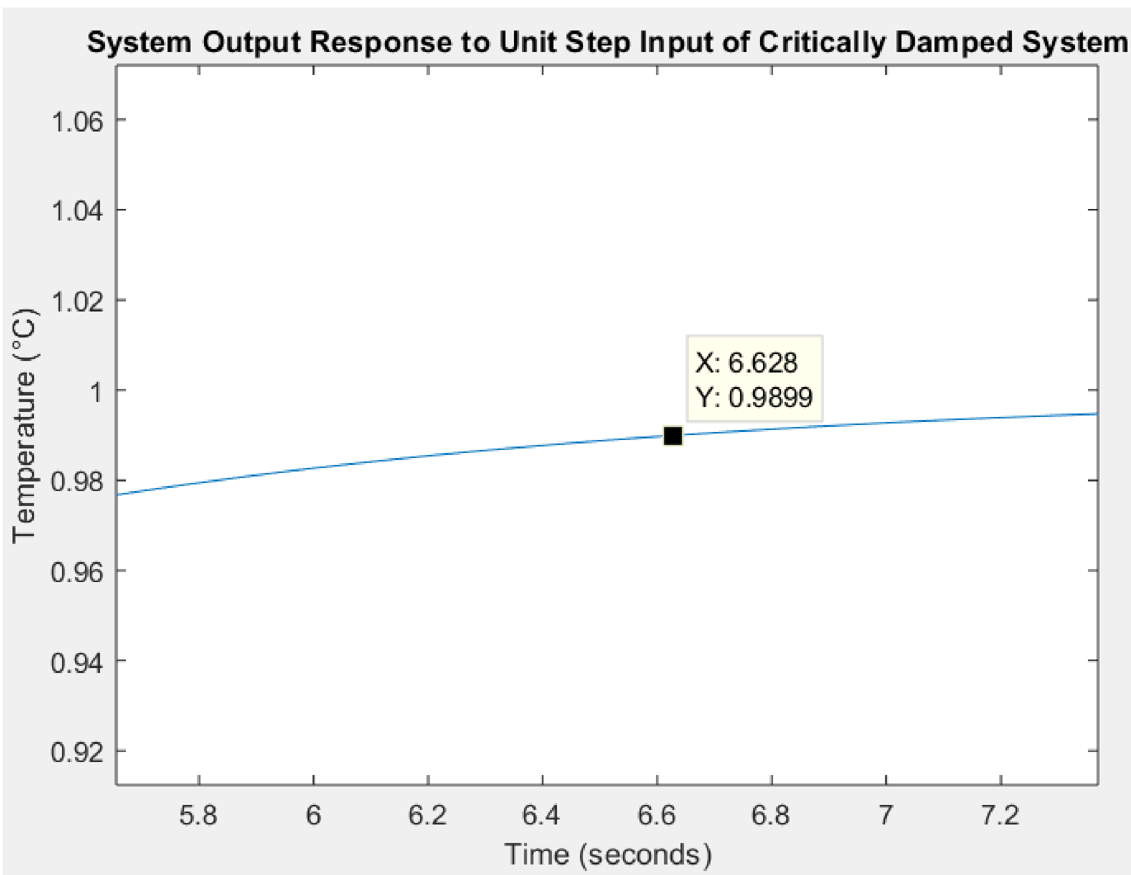


Figure 5.14: Time to 99% steady-state value using MATLAB 'Data Cursor'

There is another, even more precise, alternative method for finding the amount of time that it takes for the system response to go to 99% of the steady-state value also using a plot of the system's step response in MATLAB which was mentioned in section 5.8.3. It involves using the 'Zoom in' tool to magnify the plot to the precise position where the system output reaches the desired value of 99% of the steady-state value. Continue to 'Zoom in' on the position until the desired precision is achieved. An example of this is shown in Figure 5.15 below for the previous example. The amount of time is found by comparing where the desired output temperature is achieved on the x-axis, as shown in red in Figure 5.15. Using this approach, the amount of time that it takes the system to reach 99% of the steady-state value is found to be approximately 6.63840792 seconds for this example which exactly matches the calculated value. Although this method offers more precision than the previous method, it takes more time than the previous method but less time than the calculation method.

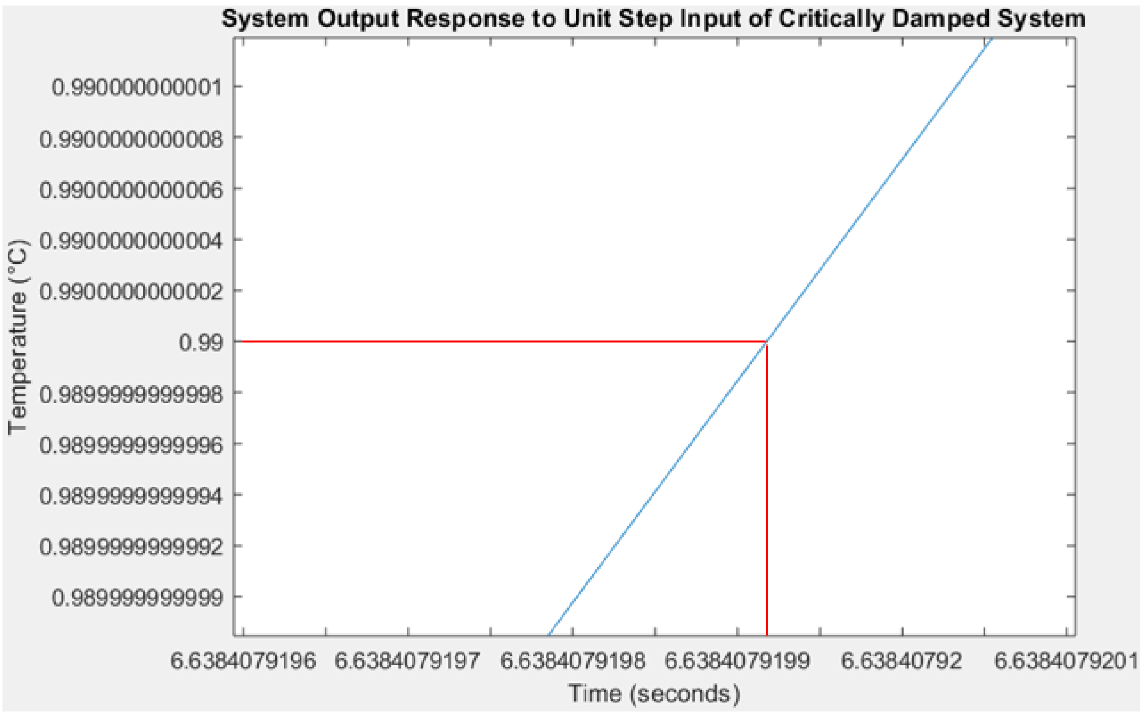


Figure 5.15: Time to 99% steady-state value using MATLAB 'Zoom in'

This method achieves the same results as the calculated values but is much quicker and minimizes human calculation error. Either of the alternative methods can be used to find approximations for the time that it takes for the system response to reach 99% of the steady-state value or the value can be calculated using the given equations. This value represents the setting time of the system with an error band of ±1%.

5.8.5 Stability

After designing the PID controller to meet the desired percent overshoot, settling time, and rise time, the last thing to verify is the stability. This is very important because if the PID controller causes an unstable system then the current passed into the resonator could increase until the system is forced into fail-safe mode.

5.9 Introduction to Software Design

The software component of this project will be implemented on the Texas Instruments MSP430FG47x mixed-signal microcontroller. The microcontroller will input voltage and current measurements and output the adjusted current that should be passed into the resonator on the next iteration. The program will carry out the following objectives: calculate the resistance from the voltage and current values passed in by the resonator, translate the calculated resistance to its corresponding temperature, display the temperature and resistance readings to the user and determine whether the current should be increased or decreased to remain at a stable temperature.

5.10 Agile Methodology

The agile software development and project management approach will be embraced by this team in the development of the microcontroller code for the temperature control system. The agile method emerged relatively recently as a new framework to creating software that addresses the unpredictability of developing software. Furthermore, it places an emphasis on adaptive planning, early delivery, continuous improvement and the ability to respond efficiently to change. The benefits of agile development include the delivery of high product quality and client satisfaction, enhanced project control and reduced risks. Since this is the team's first time implementing some of the technologies used in this project, there is potential that particular components may not be compatible with the desired system and will need to be substituted. Furthermore, there is also a possibility that chosen software libraries or testing environments will not be well-suited to support the application. The agile approach enables teams to efficiently react to change by frequently evaluating completed tasks and gathering feedback.

5.10.1 Conventional Agile Methodology

The conventional agile methodology implements iterative sequences of planning, designing, implementing, testing and feedback to a project in order to appropriately respond to the ever-changing realm of software development. The method progresses through this planning to testing loop until a deliverable is achieved, and this applies to each phase of development. This process will allow for the team to continuously adapt to new changes or knowledge.

Furthermore, the team will be utilizing the scrum framework within the agile method of development to manage the software development process. The scrum framework encourages flexibility and not micromanaging as it implies that the software team or lead is responsible for delivering a working product as opposed to the manager or project lead providing detailed instructions regarding the task at hand. In addition, the team working within a scrum environment is cross-functional which lends to a self-organizing and collaborative team. The ideals and practices of this framework fits this team well.

In scrum development, the creating of the software for the microcontroller will be divided into multiple sprints. A sprint is a time interval allotted and designated to a specific project task. The length of the sprint will be dependent on the level of complexity of the task but will be one to two weeks on average. At the beginning and end of each sprint, the team will hold a meeting led by the scrum master in order to gauge the task project and discuss any changes or obstacles. The role of the scrum master will be assigned to the software lead, Michaela Pain. This framework will add structure to the software development process and ensure that the defined tasks are being implemented on a semi-weekly basis. In addition, it will encourage a designated group meeting for software specifically on a frequent basis. Furthermore, the use of this methodology will be beneficial to every group member because it is extensively utilized within the software development practice. It will be advantageous to have experience with this type of framework prior to delving into the industry as many companies have embraced or are in the process of embracing this process. Lastly, the agile methodology is not only applicable to software development. It will be used in the software component of this project and evaluated to extend its usefulness to other parts of the project.

5.11 Software Functionality

The purpose of the software can be divided into the following three tasks: calculating the resistance and temperature of the resonator, displaying the desired information to the user, and controlling the current passed into the resonator. These tasks will be accomplished through functions programmed into the microcontroller. The software environment that has been selected to compile and implement the code is the Code Composer Studio Integrated Development Environment (IDE).

5.11.1 Standby and Operational Mode

The microcontroller program will operate in two modes: standby and operational. This was a feature requested by the research group under Dr. Abdolvand. The standby mode will limit the program loop to read relevant resonator values and output the calculated resistance and corresponding temperature to the LCD screen. The operational mode will receive the resonator voltage and current values, perform the necessary calculations to determine the appropriate resistance and temperature to display to the user, and adjust the temperature for the next iteration by incrementing or decrementing the voltage. This will allow the user to control whether the device is operating to stabilize the temperature or to simply display the state of the resonator in terms of resistance and temperature to the user. In addition, it will conserve power without sacrificing the time to power down and startup the device again. This will contribute to the low-power quality of the application when the program is in standby mode.

5.11.2 Resistance Calculation

The purpose of this task is to calculate the value of the resistance from voltage and current measurements that are passed in from the resonator. The microcontroller will take in the floating inputs from the resonator using the pins on the microcontroller. The microcontroller will then calculate the resistance using the following equation:

$$R = \frac{V}{I}$$

The resistance will need to be continuously processed due to the voltage and current being oscillated until the desired temperature is achieved. Thus, the resistance will need to be updated on a periodic basis whenever an updated voltage and current is passed from the resonator. The functionality of a timer interrupt will allow for the constant check of whether the voltage and current values have been modified. When the values have been changed, the microcontroller will proceed with the temperature conversion, as detailed in the section below. Otherwise, the microcontroller will wait until they have been updated. The microcontroller will output the value of the resistance to be searched through the lookup table as explained in the section below.

5.11.3 Temperature Translation

The purpose of this task is to determine the corresponding temperature from the resistance calculated in the previous function. The microcontroller will take the calculated resistance and search for the analogous temperature through use of a lookup table. An existing table conveying the relationship between the resistance and temperature will be implemented as a lookup table. This will contribute to the efficiency and accuracy of the application which are two desired objectives being considered throughout the project.

5.11.3.1 Lookup Table

A lookup table (LUT) is a table that returns an output for any given input. If applicable, LUTs determine a value faster than actually calculating the value through an algorithm. However, the disadvantage of using a lookup table is its memory usage which is dependent on the size of the table.

An LUT will be implemented within this program to determine the corresponding temperature based on a predetermined chart relating the given resistance to a temperature. The LUT will be defined using the static, const, and __flash keywords. The static keyword will be used to prevent the program from reallocating space and initializing the table each time the temperature needs to be looked up. The const keyword will enable the table to be read-only which is desired for this application. The __flash keyword will be used to ensure that the table is stored in the flash memory rather than the RAM. The reason behind this is due to the fact that RAM in microcontrollers is often insufficient.

The LUT will be implemented using sorted arrays since the temperature has a relational correlation to the resistance. There will be a total of two sorted arrays, one for the predetermined resistance values and the other for the predetermined temperature values. These will be associated by the index in the array. A for-loop will be used to sort through the resistance array and determine the corresponding index of the given resistance. This index will then be used to determine the corresponding temperature which will be returned to the main program.

5.11.4 User Display

The purpose of this task is to display the temperature and resistance values to the user in an easy-to-read format. The selected display will be of size 16x2. Thus, the LCD will be able to display 16 characters on each line with a total of two lines. Functions will be created and used to format the characters and symbols on the LCD. These functions will include clearing the display, writing to the second line on the display and printing a string to the display. This will allow the LCD to properly display the periodically refreshing values. The desired output on the LCD will be in a similar format as shown below.

Temperature=
Resistance=

To accommodate for the potential of adding other values to the user display and for the sake of testing, the alternative option of the display is to show only one name and value pair on the LCD at a time and use one of the buttons featured on the microcontroller to toggle between values. The flexibility of the selected LCD proves that it is a good choice for the application. It supports the potential for modifications and additional features as the project progresses.

This alternative display is shown below.

Temperature=

toggle

Resistance=

5.11.5 Voltage Control

The purpose of this task is to calculate a voltage to pass back into the resonator for the resonator to remain at a constant temperature. A PID controller will be programming into the microcontroller. Then, the microcontroller will feed the desired values into a PID controller to appropriately maintain the voltage. The PID controller will perform the calculation and output the new voltage to a Voltage to Current Converter. This current will then be passed into the resonator, and the next iteration will begin.

5.11.5.1 PID Controller

A proportional-integral-derivative (PID) controller is a control loop feedback algorithm that is often used for applications where continuous modulated control is necessary. PID controllers are a common solution to controlling a specific variable. This mechanism is implemented on the microcontroller using a macro and its advantages include the yield of simple and precise calculations.

This project calls for control of a system, specifically the temperature stability of a resonator. The desired values will be passed into the PID controller in the microcontroller. The PID controller will in turn plug those values into the formula programmed into the controller and output the result. This result will be used to determine the voltage and current passed back into the resonator.

5.11.5.2 Feedback Control Loop

A feedback loop is a process used in the design of a control system. The methodology is as such: the system output is assessed, and the system is reconfigured based on the desired output response. This can be used in the engineering of control systems where engineers are tasked with improving the performance of existing systems.

The implementation often follows the steps of processing information passed from a measurement device to a controller which then evaluates the input and calculates a correcting output that is passed back into the process. A basic control system consists of the following elements: relevant variable measurements, a desired range for the controlled variable, a control algorithm, and a control element. With respect to this project, the relevant variable measurements are temperature, voltage, and current. The desired range of the temperature is between 90°C and 100°C. The algorithm is the calculation for the input voltage, and the control element is that voltage.

5.12 Integrated Development Environment

Code Composer Studio is an integrated development environment (IDE) that was designed to support Texas Instruments' family of microcontrollers and embedded processors. Code Computer Studio provides a multitude of tools to develop and debug embedded applications. In addition, it is comprised of a compiler, source code editor, debugger, and project build environment. The tools and interfaces are user-friendly but also familiar to the members of this team which makes it a perfect choice for our application. Essentially, Code Composer Studio achieves the valuable aspects of Eclipse with advanced embedded debug capabilities.

5.13 Programming Language

The languages available in Code Composer Studio are C, C++, and assembly. However, the language of choice for these types of applications is often the C programming language since C++ has limited support. C encompasses built-in and user-defined types, data structures, and open control flow which make it the more productive and reliable language with respect to assembly. High-level languages also often produce efficient code.

In addition, the architectures of modern processors are intended for compilers rather than assembly code and online resources lean towards the C language approach. The software lead, along with the rest of the team, is fluent in the C programming language. For these reasons, the selected language for the project is C.

5.14 Algorithm Overview

The main tasks of the program will be implemented in four different code segments. The tasks are divided as follows: resistance calculation, temperature translation, user display and voltage control. All constants and variables will need to be defined for each objective. In addition, the initial values for the inputs and outputs along with any external libraries will need to be initialized at the top of the program. Global variables will be used for the resistance, temperature, current, voltage and toggle state values because they will need to be used between functions. Since the program implements these functions sequentially, there is no concern for these variables to be overwritten and cause a bug. In addition, the lookup table will be initiated through the initialization of two arrays of predetermined resistance and temperature values. This will also act as a global variable so that it is not instantiated for every iteration, when the relevant function is called. Below, each function is defined by its procedure to show the processor compilation sequence and program logic.

5.14.1 Resistance Calculation

The resistance calculation function begins the software component of this project, and its objective is to calculate the resistance of the resonator from the passed in voltage and current values of the resonator. This function would take in the resistance, current and voltage values as its parameters. Further, this function would be *void* and not return a value due to the use of global variables. The microcontroller will read in the relevant values using its I/O pins and store them into their corresponding global variables. The resistance would be calculated using simple arithmetic for the equation below.

$$R = \frac{V}{I}$$

Then, the resistance would be stored in its respective placeholder variable. Next, the function would return to the main program.

The purpose of this function is to periodically read in the voltage and current values of the resonator to calculate the resonator's resistance. This will enable the computation of the corresponding temperature and realize the capabilities of the microcontroller to interact with the resonator. The resistance calculation function's flow chart is shown in Figure 5.16.

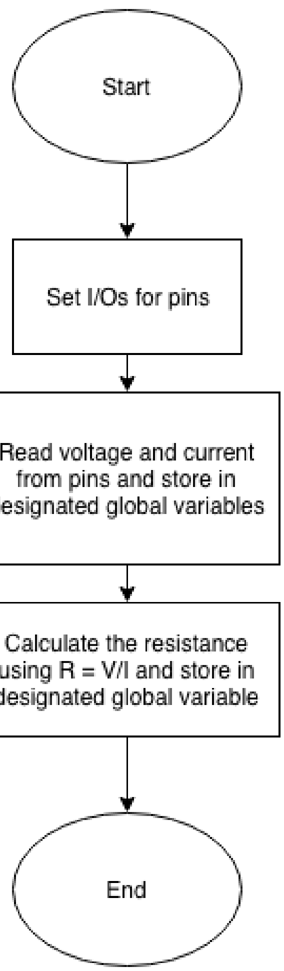


Figure 5.16: Resistance calculation flow chart

5.14.2 Temperature Translation

The temperature translation function will follow the resistance calculation function. Its objective is to convert the resistance into its corresponding temperature. This function would take in the resistance as a parameter. Similarly, this function would be *void* and not return a value due to the use of global variables. The microcontroller will iterate through a preexisting array which represents a lookup table between the resistances and temperatures. The temperature would be calculated by comparing the current value of the resistance to the resistance values within the look up table. Then, the index of the matching resistance value would be used to determine the desired temperature. This temperature would be stored in its respective placeholder variable. Next, the function would return to the main program.

The purpose of this function is to convert the calculated resistance to its corresponding temperature using a lookup table. This will allow for the temperature to be displayed to the user. In addition, the temperature is a main component that is used to improve the system over time. The temperature translation function's flow chart is shown in Figure 5.17.

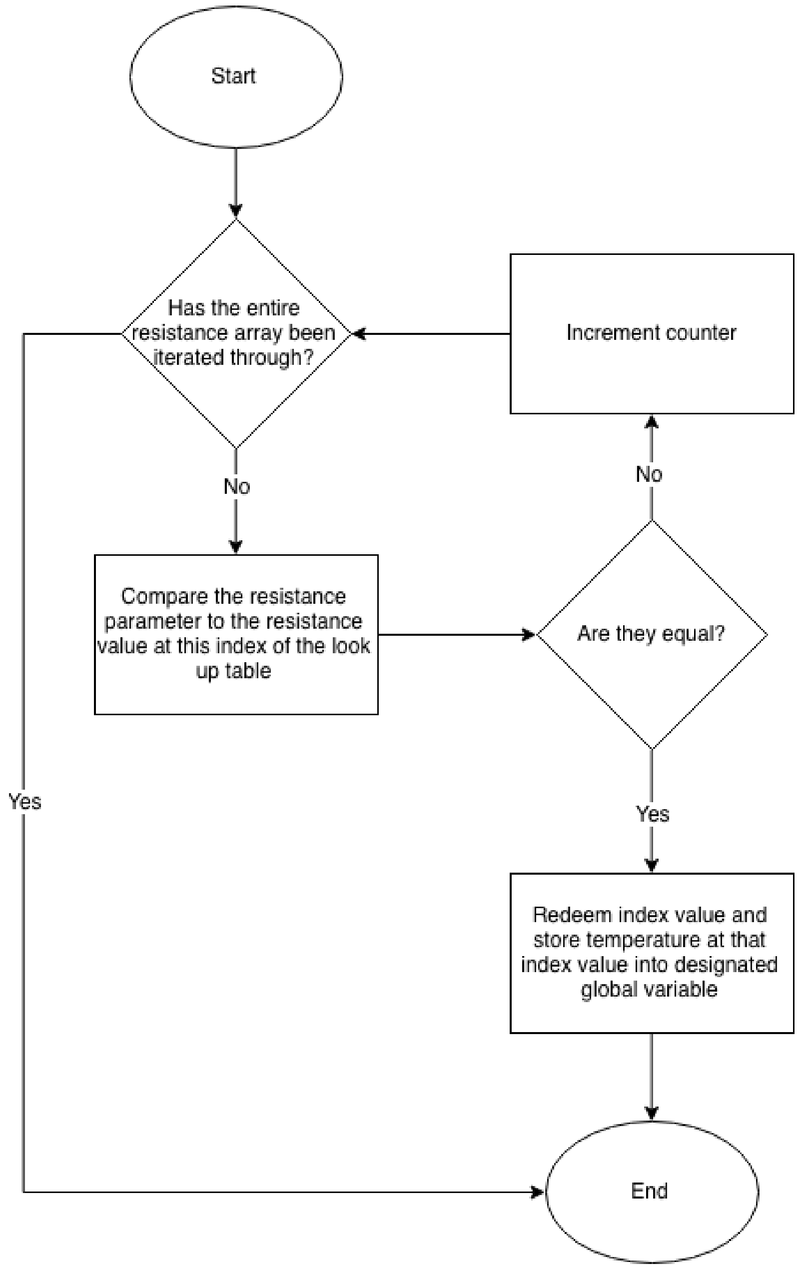


Figure 5.17: Temperature translation flow chart

5.14.3 User Display

The user display functions will follow the temperature translation function. Their objectives are to project the resistance and temperature values to the user using the LCD screen. The first function will be responsible for setting up the initial output and printing the resistance. It would take in the temperature as a parameter. Similarly, this function would be *void* and not return a value since it does not perform any type of calculation. The microcontroller will print the word temperature followed by the current value of the resistance. Next, the function would return to the main program.

The second function would be responsible for adjusting the display if the toggle switch is used. It would take in the resistance, temperature and toggle state as parameters. Depending on that state of the toggle, the microcontroller will then print the word resistance or temperature followed by the current value of the resistance or temperature, respectively. Next, the function would return to the main program.

The purpose of this function is to display and periodically update the temperature and resistance values to the user on the LCD. This will serve to communicate the functionality and operation of the program in real-time to the user. In addition, it will assist in the testing phases for the application. The second user display function's flow chart is shown in Figure 5.18.

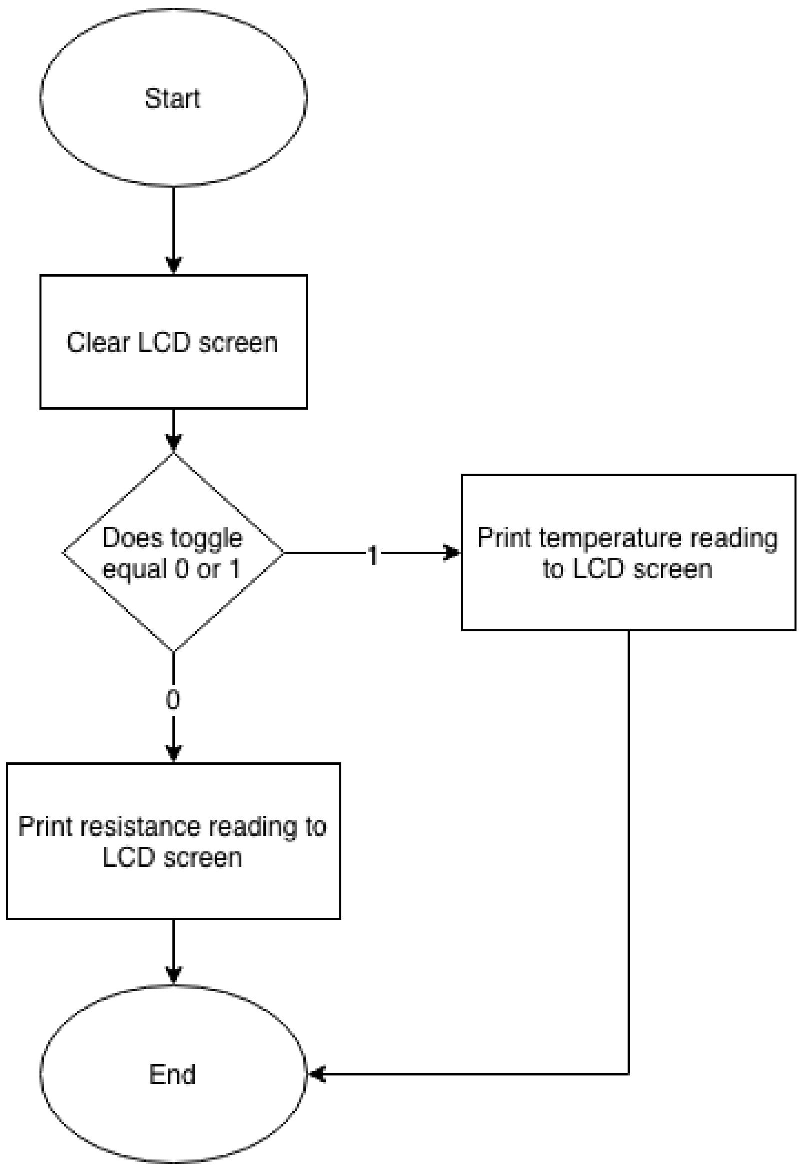


Figure 5.18: User display flow chart

5.14.4 Voltage Control

The voltage control function completes each iteration of the algorithm. Its objective is to determine the corresponding voltage dependent on the current voltage and resistance values. This function would take in the voltage and resistance as parameters. Similarly, this function would be *void* and not return a value as it will pass the required values to the appropriate device within the function. The microcontroller will serve as the liaison between the PID controller and the voltage to current converter. The microcontroller will pass the voltage and resistance values into the PID controller in order to get the desired new voltage. Once the microcontroller receives the voltage value, it will pass it to the voltage to current converter. The function will then return to the main program.

The purpose of this function is to determine how to adjust the voltage input to the resonator to achieve stability and output that value to the resonator. This serves as one of the main features of the application because it achieves the main objective of determination of a stable temperature for the oscillator. The voltage control function's flow chart is shown in Figure 5.19.

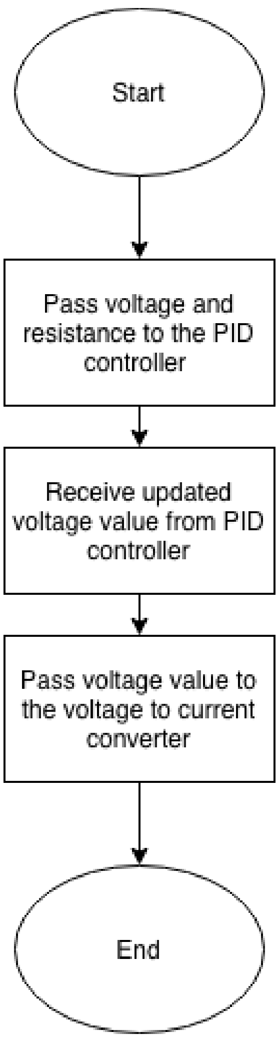


Figure 5.19: Voltage control flow chart

5.14.5 Coded Flow Chart

The complete algorithm flow chart for a single iteration is shown in Figure 5.20.

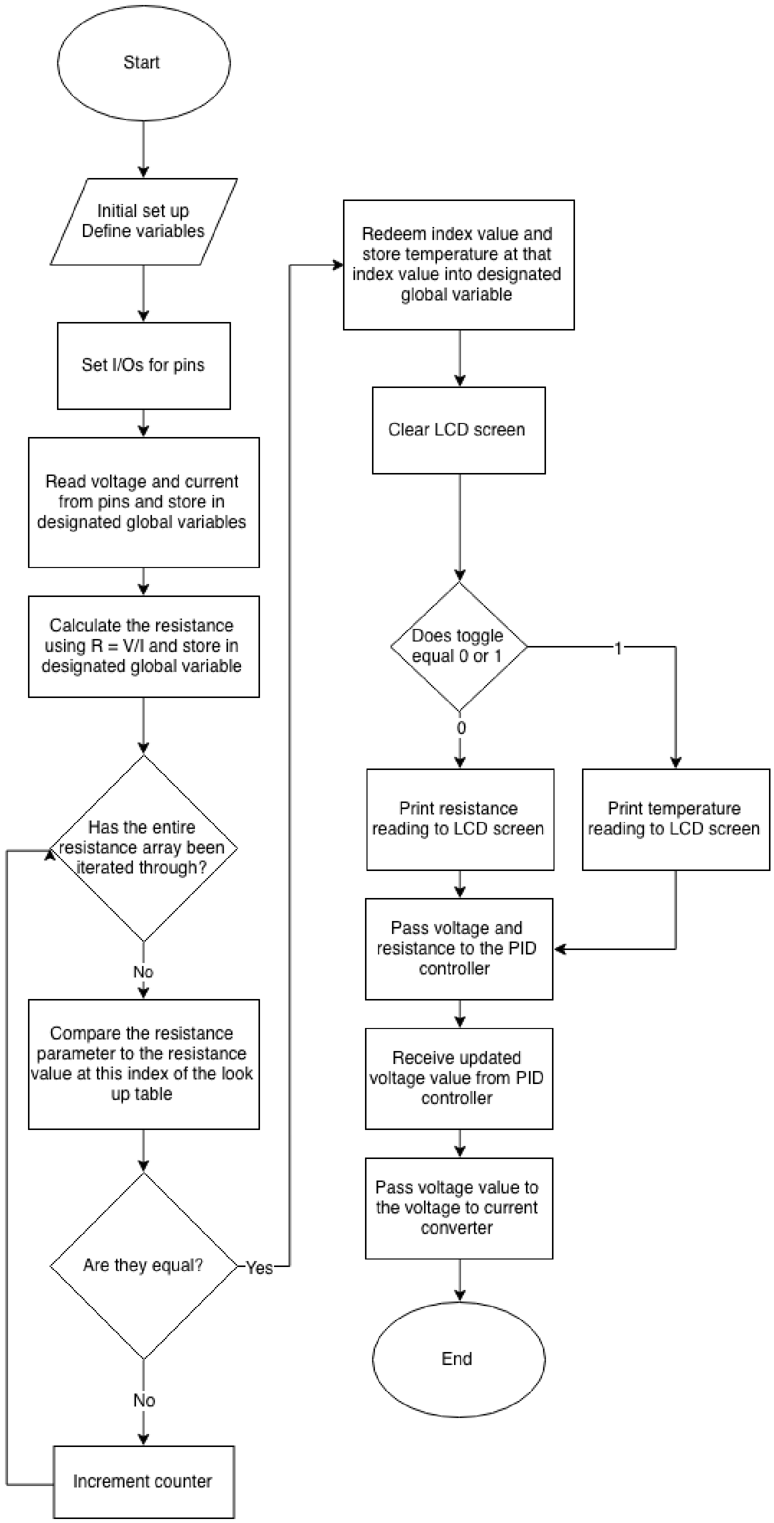


Figure 5.20: Complete algorithm flow chart

5.14.6 Coded Flow Chart Logic

The logic for the software in the microcontroller unit is shown in the software flowchart in Figure 5.20. In the chart, it is evident that the first priority when the microcontroller is switched on is to read the resonator voltage and current from the pins and store that information into designated variables. The current state of the resonator is significant as it used to determine whether the voltage and current values need to be adjusted. These values will be input as an analog signal, so they will need to be measured and converted to digital values.

Once the current values are received, the microcontroller will perform a simple calculation to determine thee resistance using Ohm's Law. Next, the program will traverse through an array that links the resistance values to their corresponding temperature values. The calculated resistance will be compared to each resistance in the array iteratively until a match is found. They index of the corresponding resistance in the table will be utilized as the index of the corresponding temperature.

Once the value of the temperature is determined, the program will seek to print this value to the screen. The program will utilize a toggle button to display relevant values to the user. For now, the relevant values are the resistance and the temperature. Since there are two values, the toggle will operate on a binary system where a value of 0 will correspond to the resistance while a value of 1 will correspond to the temperature.

Once the relevant values are output to the LCD screen, the program will determine whether the state of the resonator in terms of voltage and current need to be modified. The voltage and calculated resistance values will be passed into the PID controller. The PID controller take in the voltage and resistance and perform a calculation to determine the updated voltage value. This value will be passed back to the microcontroller, which will then redirect the voltage value to the voltage to current converter. While the new voltage and current values are passed back to the resonator, the microcontroller will wait for the state of the resonator to be passed into it again. The process will continue until the temperature reaches a stable value.

5.14.7 LCD Testing

The testing of the LCD screen is essential for enabling the user to view the resistance and temperature adjustments throughout the program execution. The resistance and temperature calculations will be performed by the microcontroller. In order to confirm whether the LCD screen is properly connected to the microcontroller and fully functional, a test message will be programmed to the LCD. The LCD should be able to display the word "Testing" using the serial write function in order to verify its connectivity and functionality.

5.15 Additional Software Features

There are additional features that have been under consideration for additions to the project. These components would be able to be implemented through additional functions and code programmed to the microcontroller. These capabilities would not affect the satisfaction of the basic requirements for the application but act as additions to benefit the application user. Consequently, these additions should not be expected in the final design of this project as they will be implemented if the team decides to expand the functionality of the application and if time permits.

5.16 Potential Obstacles and Sources of Error

There are a number of potential obstacles and sources of error that may have a negative effect on the project design, construction, or usage. Strategies on how to minimize or eliminate these factors should be considered.

6. <u>Project Construction</u>

This section analyzes how the project will be implemented and assembled after the initial design is completed.

6.1 Bill of Materials

The following table lists the parts to be used for the printed circuit board for this project.

Table 6.1: Bill of Materials

Name	**Part Number**	**Manufacturer**	**Number in Design**
Microcontroller	MSP432G2ET	Texas Instruments	1
Instrumentation Amplifier	INA217	Texas Instruments	2
Voltage Reference	LM4050	Texas Instruments	1
Op-Amp	OP828	Texas Instruments	1
Step-down Power Module	LMZM23600V3SILR	Texas Instruments	1
Step-down Power Convertor	TPS62173DSGR	Texas Instruments	1
0 TCR Resistor	TBD	TBD	1
NPN Transistor	TBD	TBD	1
Display	TC1602A-09T	Adafruit	1
Resistors	Various	Varies	Various
Capacitors	Various	Varies	Various
Digital Potentiometer	TPL0501	Texas Instruments	1

6.2 Resonator Testing

This section elaborates on the testing specific to the resonator and is based on work from one of the group member's undergraduate thesis work.

6.2.1 Setup

To characterize the initial response of the resonator to current and temperature stimuli, the portion of a wafer containing the resonator was glued to a portion of a silicon wafer using thermal paste. The thermal paste allows the heat from the chamber chuck to transfer to the die.

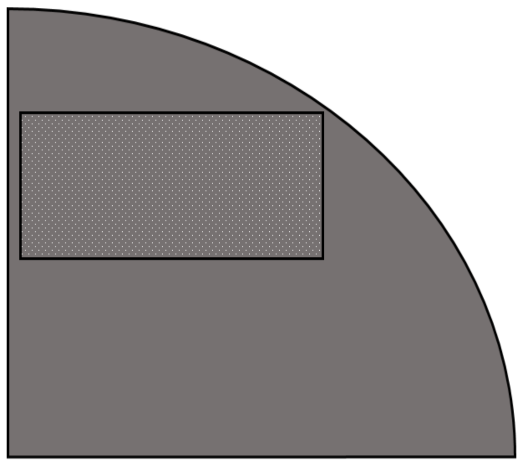


Figure 6.1: Resonator sample adhered to portion of silicon wafer [39]

Then, a portion of a copper breadboard had pins soldered to it and was also glued to a portion of a silicon wafer using an adhesive. The pins were soldered on in order to provide easier access for reading the voltage and/or current associated with the resonator under certain conditions. Without these pins, connecting a multimeter or measurement device to read the voltage across or current through the resonator would be cumbersome.

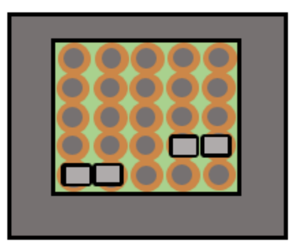


Figure 6.2: Breadboard with soldered pins on silicon [39]

This portion of the silicon wafer was also glued to the same wafer as the die. They were placed close together so that wire bonding could be done as easily as possible. The resonator was usually bonded to the copper pads closest to the edge of the breadboard; as the wire bonding was completed many times over, it became more and more apparent that the angle at which the two bonds were completed was important. For instance, a bond completed as shown below on the left is not good because the second one is at an angle compared to the first bond. In contrast, on the right, the second bond is exactly 90° north of the first bond. When the bonds are not made at the proper angle, stresses on the first bond are often greater than the force adhering the bond to the copper breadboard, and the first bond will lift off of the substrate. Hence, the second bond must be 90° north of the first, even if this means rotating the setup to make this feasible.

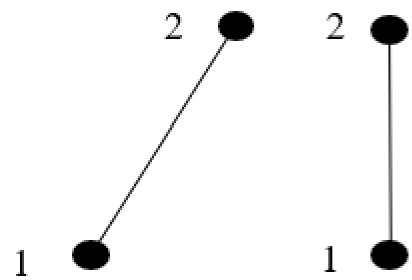


Figure 6.3: Example of bad (left) and good (right) first and second bond location

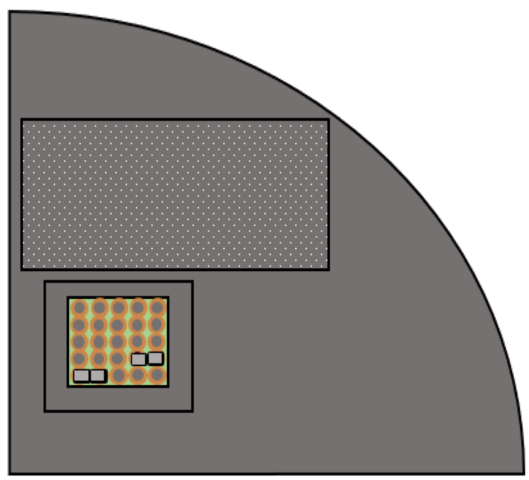


Figure 6.4: Wafer portion with resonator sample and breadboard [39]

The above figure shows the final testing setup just prior to the wire bonding between the die and the breadboard. The following image depicts the (from left to right) the breadboard with soldered pins, sample of the wafer containing the resonators, and the wire bonder needle. This is before the resonator has been bonded to the breadboard.

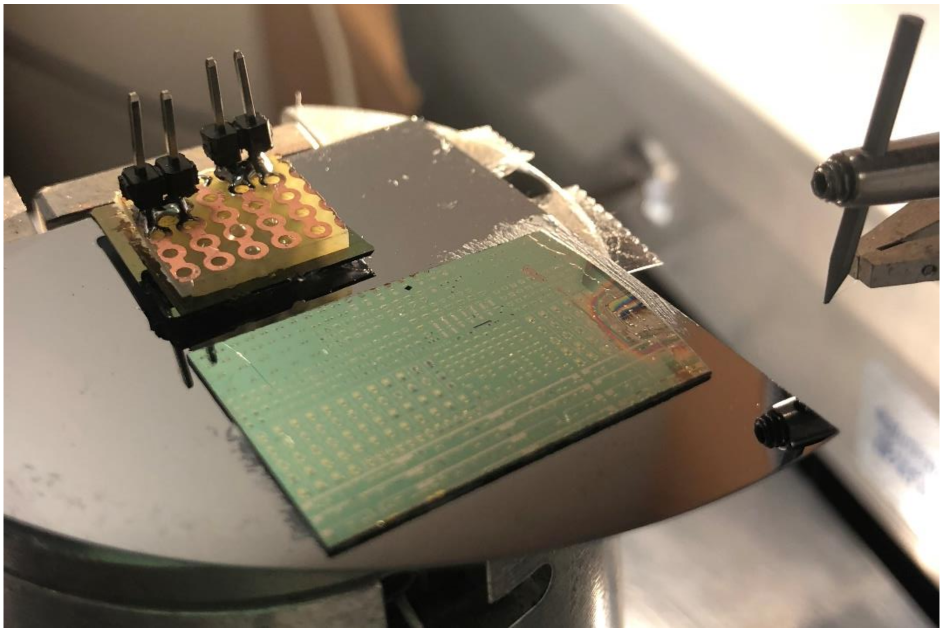


Figure 6.5: Actual setup of resonator testing [39]

Figure 6.6 depicts the final testing setup after wire bonding is complete. Then, an ohmmeter or voltage source can be attached to connector pins to obtain a better understanding of the electrical characteristics of the resonator.

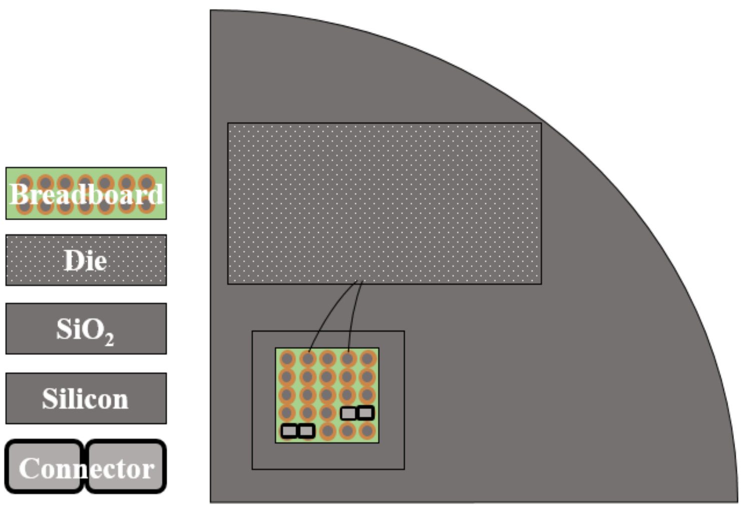


Figure 6.6: Final testing setup [39]

6.2.2 Wire Bonding

The breadboard rows connected to each pin were each connected to a side of the resonator using aluminum wire from an ultrasonic wire bonder. Aluminum wire was chosen because of its low cost and the low temperature at which it can be used. The bonder used was the West-Bond 7400A. This is an ultrasonic wire bonder that is used at room temperature. It was used for this experiment to make wedge bonds. Essentially, there is a bit of wire that protrudes from the needle and is melted into a sphere using ultrasonic energy. The first ball is deposited in such a way that the aluminum wire does not break; thus, the needle can be moved to another place for the second ball to be deposited. After this, the wire in the needle snaps, and the user is left with a wire bonded in two places on the substrate [43].

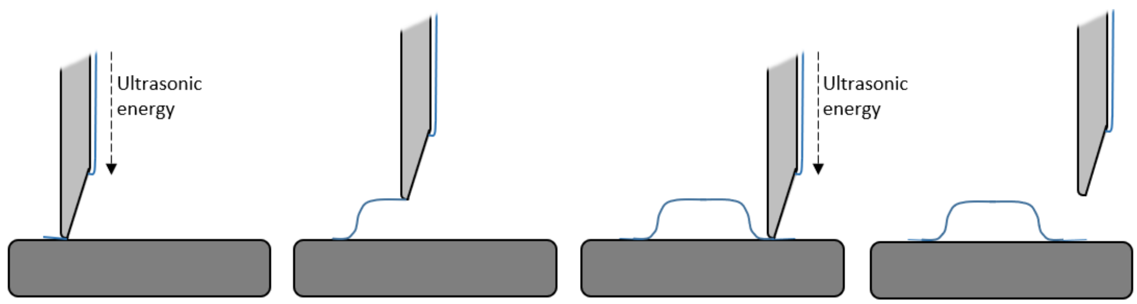


Figure 7: Sample of wire bonding activity [39]

For both depositions, the wire bonder is activated by pressing the needle into the substrate. The user-controlled variables on the machine are time and power, and the user also has control over the speed and force with which the needle is moved and placed on the substrate.

Using this wire bonder takes skill and experience. There are a number of aspects of operation that have to be done in a particular manner, or else the bonds will not hold. Descriptions, pictures, and instructions are no substitute for experience on this machine. To find optimal bonding conditions, it is recommended that the user practice on another substrate until familiarity with the machine is gained.

6.3 Passive Temperature Testing

To complete passive temperature and resistance characterization as a method to determine what range of resistance should be expected and how high temperatures affect the device, the resonator was placed in a vacuum chamber made by Janis Research Company Inc.

Inside of the chamber, wires to connect the device so that measurements could be made outside of the chamber were soldered to the pins on the breadboard. On the outside, wires were soldered to pins and then soldered to alligator clips to reduce connector resistance. Then, the other ends of the alligator clips were connected to a multimeter. The pressure was reduced to approximately 1-5mTorrs; this range of values is due to the inability of the vacuum chamber to reach lower pressures if it had not recently been used.

The chuck that the overall piece of silicon was placed on is connected to a heater that can be controlled from outside the chamber. The chuck temperature was set to 30°C, 40°C, 50°C, 60°C, 70°C, 80°C, 90°C, 100°C, and 110°C, each time using a PID controller built into the system.

6.3.1 Challenges and Optimization

Since this type of characterization of the resonator's resistance and temperature is new for this type of device in the Dynamic Microsystems Lab, there were certainly some adjustments needed to the initial TCR testing. Some of the difficulties encountered and how they were overcome are described in the following sections.

6.3.1.1 Wire Bonds Over 60°C

An initial challenge face by the passive temperature characterization was the instability of the wire bonds over chuck temperatures of 50°C. The first two trials of placing the sample under vacuum resulted in a measurement of infinite resistance (meaning the wire bonds had come undone) when the heater was set to 60°C. This was not acceptable since the resonator sample would eventually need to be kept at a constant temperature of 100°C.

Since for these trials the adhesive used to bond the breadboard and silicon dioxide and silicon was hot glue, it was hypothesized that the elevated temperature was causing a settling and shifting of the breadboard and jarring the wire bonds loose. Thus, Loctite plastic epoxy was used since it was rated for temperatures of up to 150°C. This was a very successful fix and allowed subsequent tests with the chuck set to 110°C and no broken bonds.

6.3.1.2 Variety in Resistance Measurements

Another issue that arose was the difficulty of obtaining consistent resistance measurements. Because of the complexity of the test setup and challenges of correctly wire bonding, the initial resistance measurements varied by several hundred ohms. However, this was able to be improved as testing progressed.

6.4 Characterization Results

Characterization of the TCR is incomplete at this time but will be completed during the next semester following the sample undergoing rapid thermal annealing to provide more Ohmic contacts with the resonator.

6.5 Project Testing and Prototype Construction

The purpose of the prototyping stage is to create a testable model that meets project requirements and will allow for replication or improvements from. This is crucial in the design process because it will allow the current design to be evaluated and further prioritized. Furthermore, it will clarify the production process and identify costs and issues. These tests will be performed on the both the hardware and software components of this project to ensure full functionality and addressment of any faults.

6.5.1 Resistor Testing

Initially, the final printed circuit board will not be tested on a resonator; it will be tested on a resistor. The two pads for wire bonding the resonator will have a resistor soldered to them. Since resistors do change value with temperature, the resistor's datasheet can be referred to for an examination of the effectiveness of the circuit. However, the resistor should not exhibit resonance, so testing the frequency stability will be more difficult in this stage. This step will mostly test the voltage and current and their regulation. A resonator will not be used in this step because one of the requirements of the circuit is that it be failsafe. However, this cannot be ascertained until the overall operation of the PCB has been tested. Thus, a resistor will be used in place of the resonator for initial testing.

6.5.2 Resonator Testing

When the circuit has been deemed failsafe and properly functional, a resonator will be wire bonded to the connection pads. Then, operation and performance of the resonator can be examined both in terms of temperature and resistance. A bench supply will be used to power the circuit. The resistance and temperature should be shown to the user on the display. Some of the testing for the resonator is based on a process developed by one of the team members for her thesis for the Honors in the Major program at UCF [39].

6.5.3 Demonstration

The circuit board is designed to be powered using a bench supply and thus can be demonstrated in any indoor setting with access to a wall plug to power the bunch supply. Prior to the demonstration, a resonator or resistor will need to be connected to the pads for resistance measurements. This can be done through wire bonding for a resonator and soldering for a resistor. The power supply will need to be set to 10V and activated. The user will need to set the desired temperature for the resonator. The output for user edification should be displayed on the LCD screen. Once the desired temperature and setpoint have been reached, the circuit should operate at that temperature and resistance with little deviation. This can be verified by user observation.

6.5.4 Software Testing

The evaluation of the software is critical for verifying the correct performance of the application. This software component of this system will need to be able to power on, receive voltage and current inputs and perform calculations and conversions based on the input. Furthermore, the processor will need to be able to properly present the resistance and temperature measurements to the LCD screen, and the microcontroller will need to be able to produce a new voltage value based on the given inputs to send back to the resonator. Also, it would be advantageous if the microcontroller had the capability to display error messages to the LCD screen once common errors are determined.

Objective: The objective of this test is to verify that the software component of this project is satisfying requirements and functioning as programmed.

Environment: The chosen environment for testing and prototyping is the designated Senior Design Laboratory in the Engineering II Building, Room 456 at UCF. Personal computers with Code Composer Studio will be used to modify the code while testing.

Procedure: The following steps will be implemented in order to test the functionality of the software component of this project.

1. Turn on the processor and anticipate for the LCD screen to properly display the test message. The display should read the word “Testing”.
2. Verify that the display clears itself and configures the resistance to be displayed on the first line of the LCD screen.
3. The display should now read the measurement phrase “Resistance = […]”
4. Compare the manually measured resistance with the resistance displayed on the LCD screen.

5. Convert this resistance to the corresponding temperature using the given lookup table and toggle the reading on the LCD screen by clicking the button on the microcontroller.
6. Verify that the display clears itself and configures the temperature to be displayed on the first line of the LCD screen.
7. The display should now read the measurement phrase “Temperature = […]”
8. Compare the manually determined temperature with the temperature displayed on the LCD screen.
9. The algorithm will be deemed successful if the resistance and temperatures values fall within the appropriate level of tolerance. Otherwise, the test will need to be rerun and there will be possibility for modification of the current code.

Figure 6.8 shows the initial stages of testing with a device from the MSP430 family with an attached LCD screen.

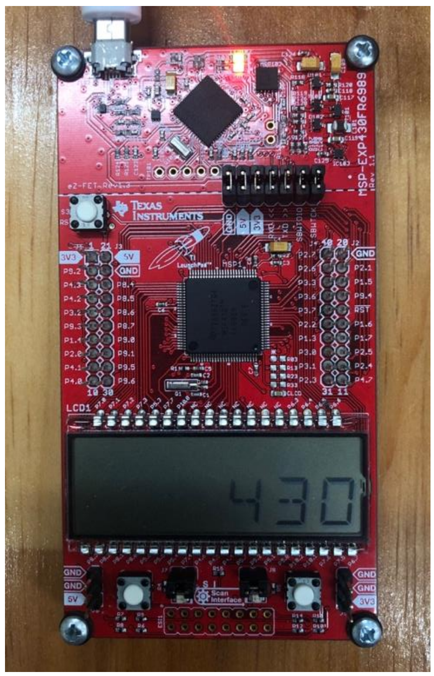


Figure 6.8: MSP430 testing

6.5.5 Operational-Amplifier Circuit Testing

Within the project schematic design, there are two operational-amplifier sections that are used to amplify or decrease voltage to ensure that it is within the input voltage range of the microcontroller. The voltage measured across the resistor is expected to be 0.5V or less, so a non-inverting amplifier circuit will be used to amplify the signal before it is passed to the microcontroller. The voltage measured across the resonator is expected to be 6V or less, so a voltage divider circuit will be used to decrease the signal before it is passed to the microcontroller. The circuits are designed to ensure that the maximum voltage passed to the microcontroller is 3V. First, the non-inverting amplifier circuit will be designed.

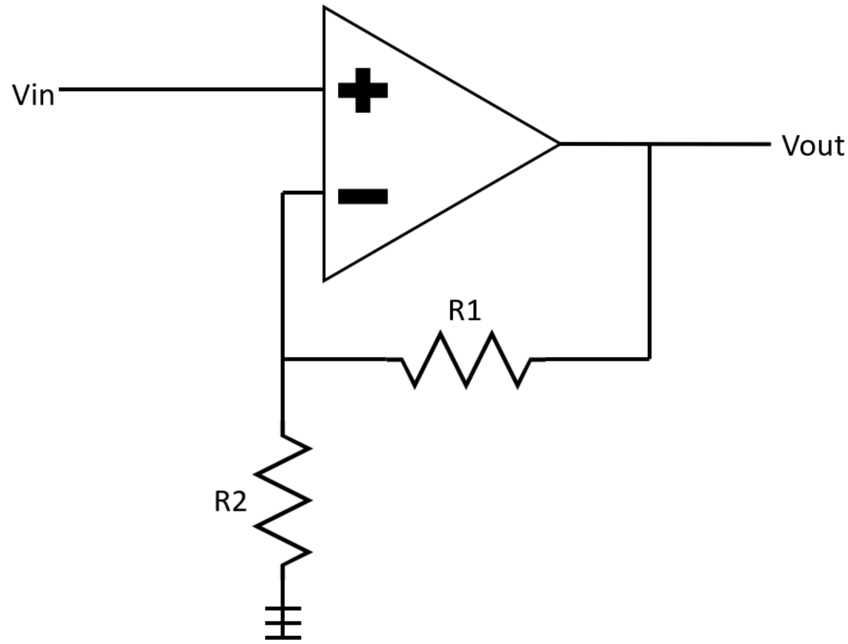


Figure 6.9: Non-inverting amplifier circuit schematic

Since the maximum voltage measured across the resistor is expected to be less than 0.5V but the desired maximum voltage is 3V, the gain of the circuit should be 6. The transfer function of the non-inverting amplifier circuit schematic shown in Figure 6.9 is set equal to the desired gain (6):

$$\frac{Vout}{Vin} = 1 + \frac{R1}{R2} = 6$$

It is calculated that resistor R1 should be 5 times larger than the resistor R2. Within the project design, R2 is chosen to be 1kΩ and R1 is calculated to be 5kΩ to achieve the desired gain. For breadboard testing, the TL084 operational amplifier is used because it is large enough for breadboard testing (the operational amplifier used within this project design is too small to test on a breadboard). A DC power supply will be used to provide 10V for the operational amplifier’s positive voltage supply and -10V for the negative voltage supply. The function generator will be used to provide the input voltage. The resistors used during testing had a tolerance of 5% which is more than the ones that will be used within the project design. This circuit is then built on a breadboard for testing as shown in Figure 6.10.

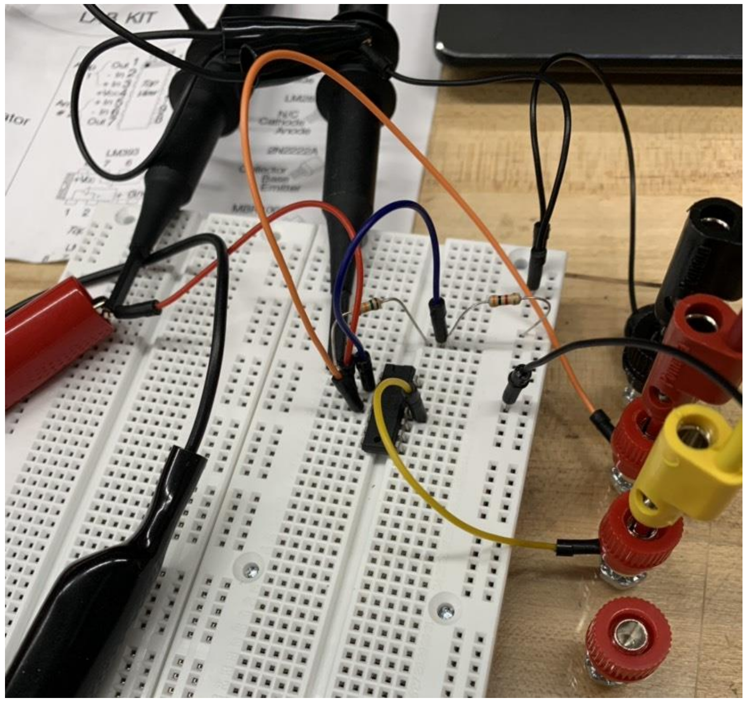


Figure 6.10: Non-inverting amplifier circuit

To test that this design works for a range of voltage values, the input voltage (Vin) is set to a low frequency (200Hz) sine wave with a peak-to-peak voltage (V_{pp}) of 500mV using a function generator. The oscilloscope is used to measure the input voltage waveform (dark blue) and output voltage waveform (light blue) as shown in Figure 6.11.

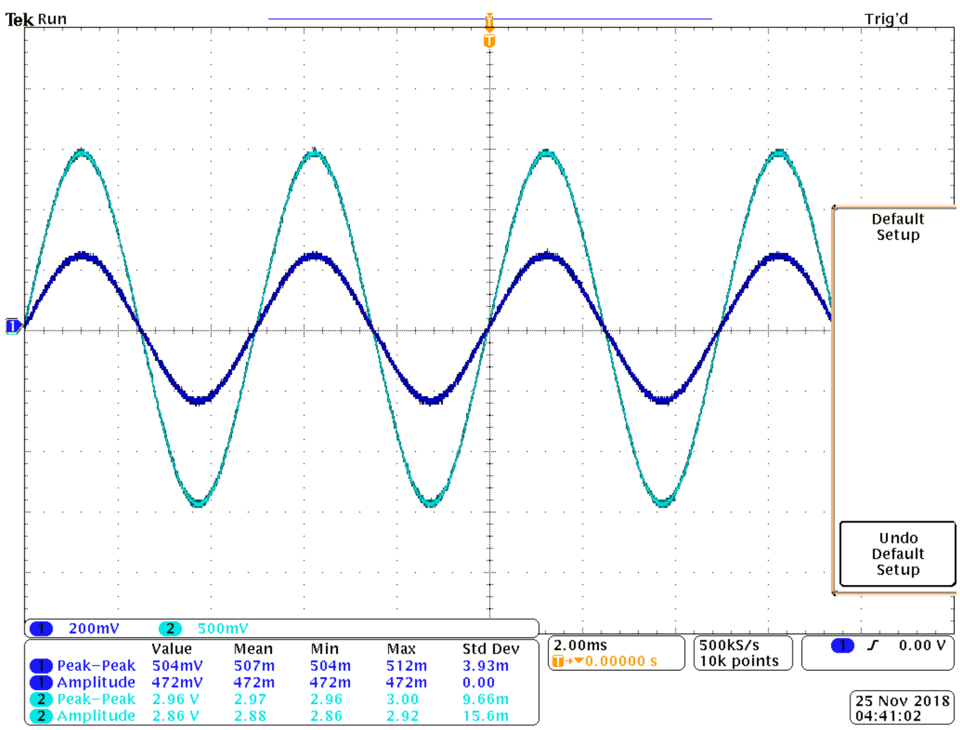


Figure 6.11: Non-inverting amplifier circuit voltage output and input waveforms

During breadboard testing, it is found that the input voltage was 504 mV_{pp} and the output voltage was 2.96 V_{pp} as shown in Figure 6.11. That is a total gain of 5.873 compared to the desired gain of 6. That means there is only a 2.1167% error, which is very good considering the resistors used for testing have a 5% tolerance. Next, the voltage divider circuit will be designed.

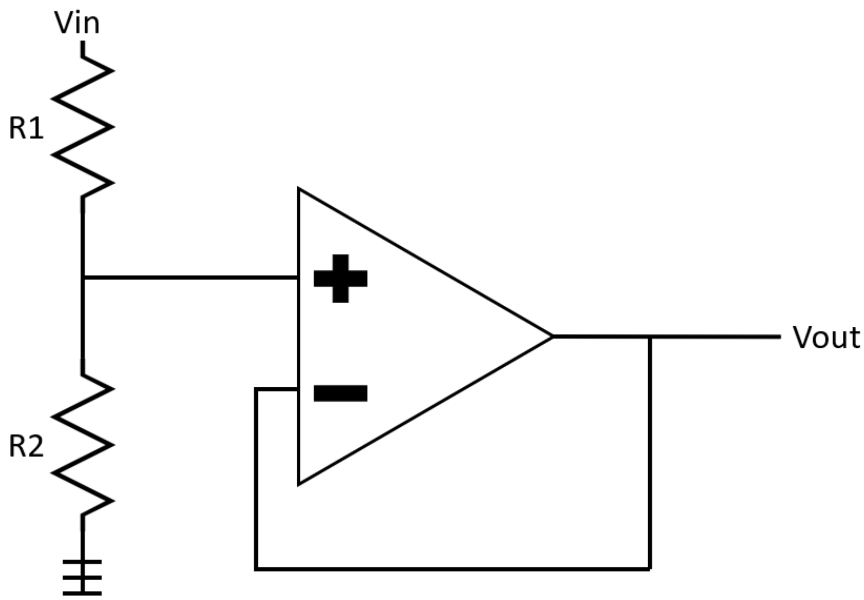


Figure 6.12: Voltage divider circuit schematic

Since the maximum voltage measured across the resonator is expected to be 6V but the desired maximum voltage is 3V, the gain of the circuit should be 0.5. The transfer function of the voltage divider amplifier circuit schematic shown in Figure 6.12 is set equal to the desired gain (0.5):

$$\frac{Vout}{Vin} = \frac{R2}{R1 + R2} = 0.5$$

It is calculated that resistor R1 should be equal to resistor R2. Within the project design, resistors R1 and R2 are chosen to be 1kΩ to achieve the desired gain. For breadboard testing, the TL084 operational amplifier is used because it is large enough for breadboard testing (the operational amplifier used within this project design is too small to test on a breadboard). A DC power supply will be used to provide 10V for the operational amplifier's positive voltage supply and -10V for the negative voltage supply. The function generator will be used to provide the input voltage. The resistors used during testing had a tolerance of 5% which is more than the ones that will be used within the project design. This circuit is then built on a breadboard for testing as shown in Figure 6.13.

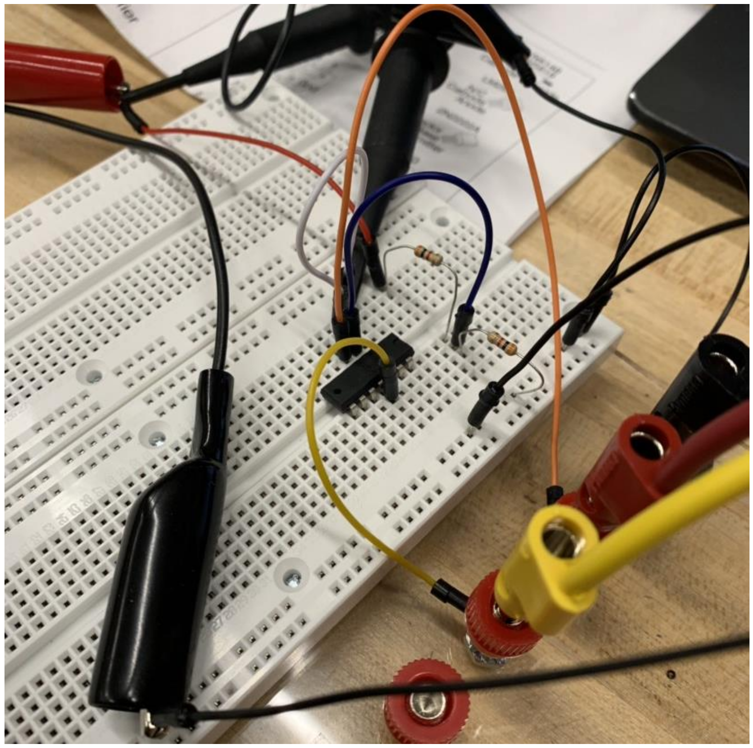


Figure 6.13: Voltage divider circuit

To test that this design works for a range of voltage values, the input voltage (Vin) is set to a low frequency (200Hz) sine wave with a peak-to-peak voltage of 1V using a function generator. The oscilloscope is used to measure the input voltage waveform (dark blue) and output voltage waveform (light blue) as shown in Figure 6.14.

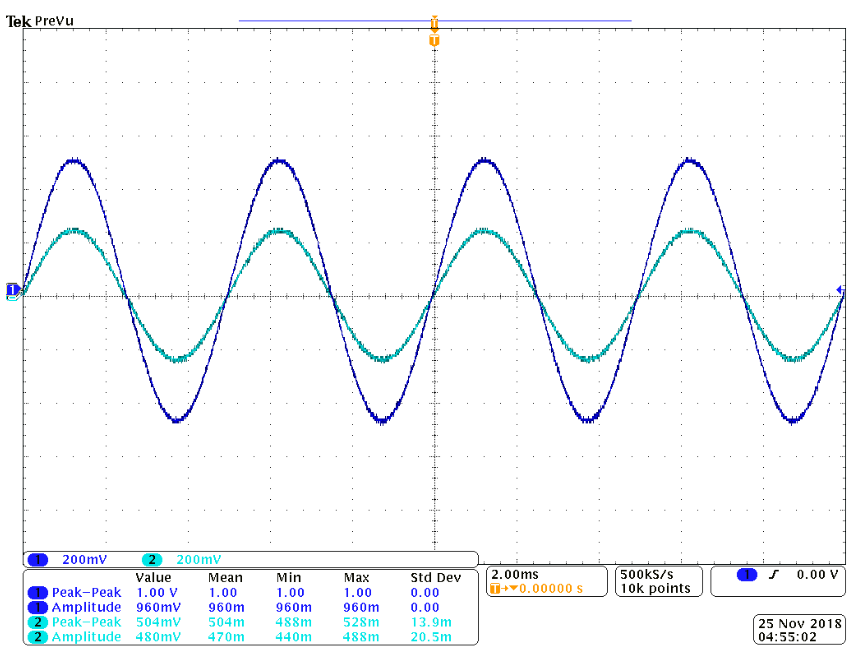


Figure 6.14: Voltage divider circuit voltage output and input waveforms

During breadboard testing, it is found that the input voltage was 1 Vpp and the output voltage was 504 mVpp as shown in Figure 6.11. That is a total gain of 0.504 compared to the desired gain of 0.5. That means there is only a 0.8% error which is very good considering the resistors used for testing have a 5% tolerance.

6.5.6 Hardware Component Testing

Once the prototype board has been ordered and received, the project will be tested using test points on the physical board. Issues with the design will be investigated and tweaked so that future PCB revisions meet design requirements. The project sponsor has indicated that breadboard testing is impractical because of small component size and would prefer for the team to complete iterations of PCB design and testing to produce a finished product.

Currently, the PCB design has been largely completed and only needs some minor adjustments prior to being ordered.

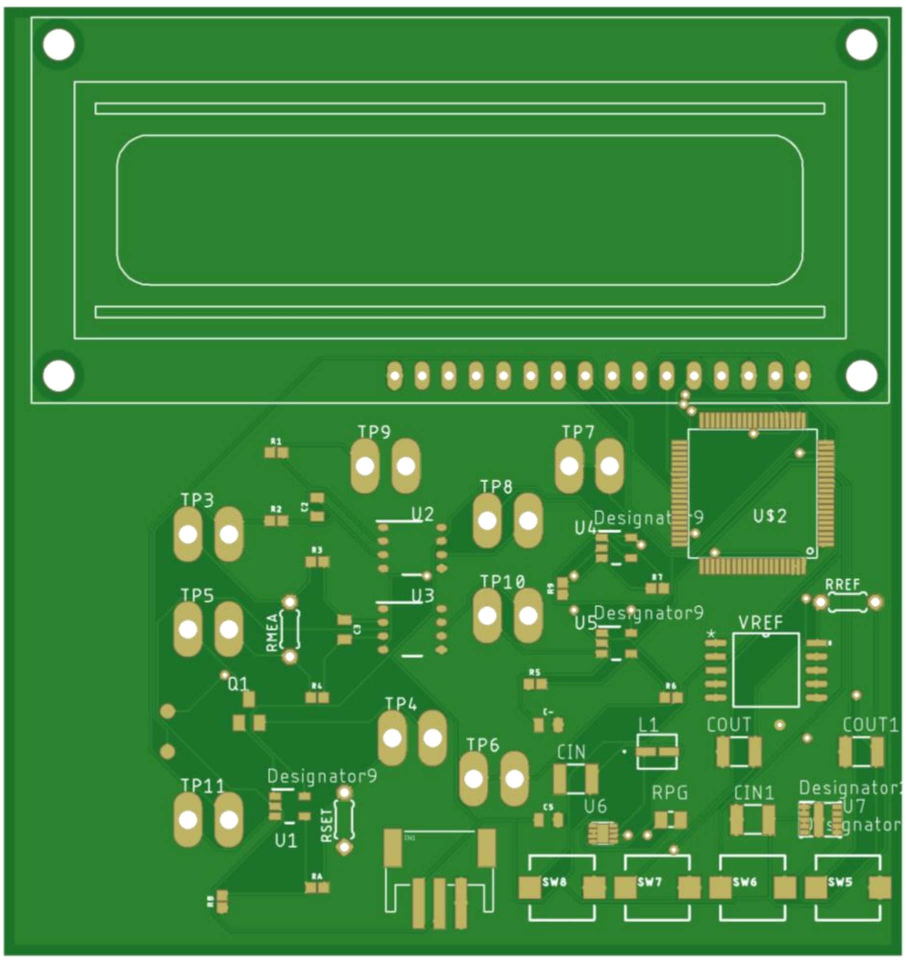


Figure 6.15: PCB manufacturing layout

There is a total of nine sets of test points on the board that will allow each portion of the circuit to be tested. Jumper wires will need to be soldered to the test points for the circuit to be tested. This will allow for sequential testing of each portion of the circuit.

For initial testing, a resistor will be soldered onto the copper connection points on the left side of the circuit board. Extra space has been created on the left side on the PCB. This is so that the portion of the wafer containing the resonator samples can be adhered to the board for a physically stable setup. The LCD here is at the top of the board, with the switches and power input on the bottom right side.

6.6 Project Operation

This section describes user operation of the finished product and offers details about troubleshooting and mistakes to avoid when using this circuit board. There is also a section about tips and tricks of wire bonding for these devices on the machine found in the Dynamic Microsystems Lab at UCF.

6.6.1 Wire Bonding Tips

Before practicing on the actual resonator sample, be sure to practice with another device first. Particularly for manual wire bonders, there will be a learning curve as the proper power and time are set and as the machine is correctly threaded and operated. The wire bonder used for testing this device has to have the wire at exactly the right place between the clamps. This could be adjusted by increasing or decreasing the needle height. Ensure that the power supply and the machine are one the same bond number (1 or 2) each time.

If the bonds are not sticking, ensure that the sample's substrate is well adhered to whatever stage is being used. If using tape or glue, try applying a new layer. If the substrate is not well adhered to the stage, the vibrations from the ultrasonic bonding process will cause the substrate to vibrate and not be as stable of a medium for bonding.

Power optimization is also very crucial, as too little or too much power can also cause the bonds to not adhere to the sample or the connector. Often, this range can also be very narrow; sometimes, changing the power by five can mean the difference between a strong and weak bond. Thus, trial and error and research must be carried out to find the best power and time settings for that particular substrate.

For the West-Bond 7400A used for these experiments, if the needle cannot be threaded because the wire keeps curling, the needle is potentially jammed. Its threading hole can be examined for a jam under a microscope. If there is indeed a piece of wire wedged into the needle and blocking it from being threaded, the needle can either be soaked in acetone to unclog it or must simply be replaced.

7. Personnel and Administrative Content

This section discusses all personnel contributing to this project, financing, and any other administrative content related to this project.

7.1 Project Responsibilities

The project roles and responsibilities of each member are detailed in this section. The group consists of three members: two electrical engineers and one computer engineer. The main priorities consist of schematic design, PCB design and software design and are assigned as explained below.

Megan Driggers will take responsibility for the PID controller design, PCB design, and setup of the power supplies. She has prior experience working to design a PCB using AutoCAD Eagle and Texas Instrument's WEBENCH power designer software in the Junior Design class at UCF. She also has experience examining component datasheets and power requirements from the Electronics 2 class and lab taught at UCF which will give an advantage in choosing the best-suited components. Megan has also taken several courses involving power and systems including fundamentals of power systems, linear control systems, and digital control systems. This background knowledge will translate directly to her tasks.

Michaela Pain will be responsible for choosing the microcontroller and its accompanying accessories to achieve the user-related requirements outlined by Dr. Abdolvand's research group. This will include defining the appropriate inputs and outputs for the microcontroller and programming the microcontroller to calculate the necessary values based on the received input. Michaela Pain has experience with microcontroller programming from the Digital Systems and Embedded Systems classes offered at UCF. She will be responsible for any additional software features that are incorporated into the project throughout the product development lifecycle. As a computer engineering major, she has the most software experience from classes and internships. Hence, she is the best team member to focus on the microcontroller and its affiliated information.

Heather Hofstee will design the schematic in LTSpiceXVII and AutoCAD Eagle. In addition, she will also serve as the project lead. She has done research with Dr. Abdolvand and his group for the past year, so she is the most familiar with project and its background. She also sees Dr. Abdolvand regularly and is able to ask him questions related to the project and relay the information to the team. Her background knowledge of the project and its objective allows her to be able to design the project schematic with full insight while her experiences with the people involved ensure that the project keeps moving forward in the right direction.

In addition, all team members will be responsible for testing their designated designs to achieve a reliable and trusted product. It is encouraged for members to reach out and communicate regarding any roadblocks encountered during the design and test phases of the project, especially if the issue is cross-disciplinary. Members will be held accountable for providing documentation regarding their specific roles to the project. The objective is to have each member contribute an equivalent amount of documentation towards the project to promote a fair working environment. This team dynamic will encourage prompt and quality deliverables.

7.1.1 PCB Design Process Flowchart

Figure 7.1 shows the PCB design process flowchart which breakdowns the work into discrete milestones and demonstrates the order in which activities need to be completed. In addition, the chart displays the statuses of each phase.

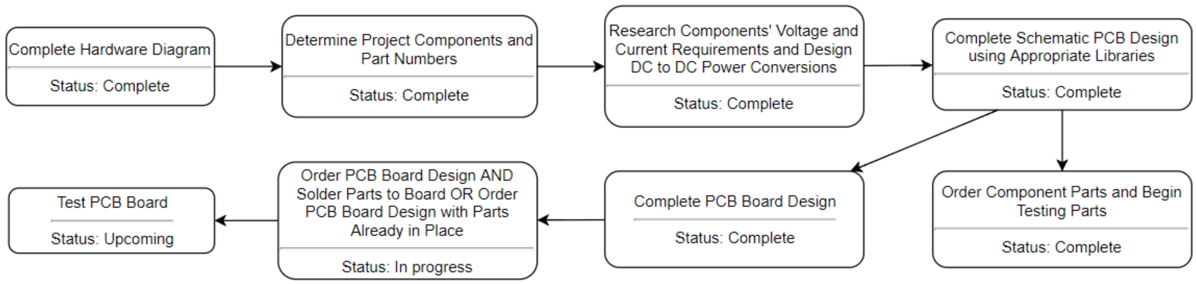


Figure 7.1: PCB design process flowchart

7.1.2 Software Flowchart

Figure 7.2 shows the software flow diagram which details the software logic to be implemented to achieve the project objectives. In addition, it communicates the intended functionality of the microcontroller.

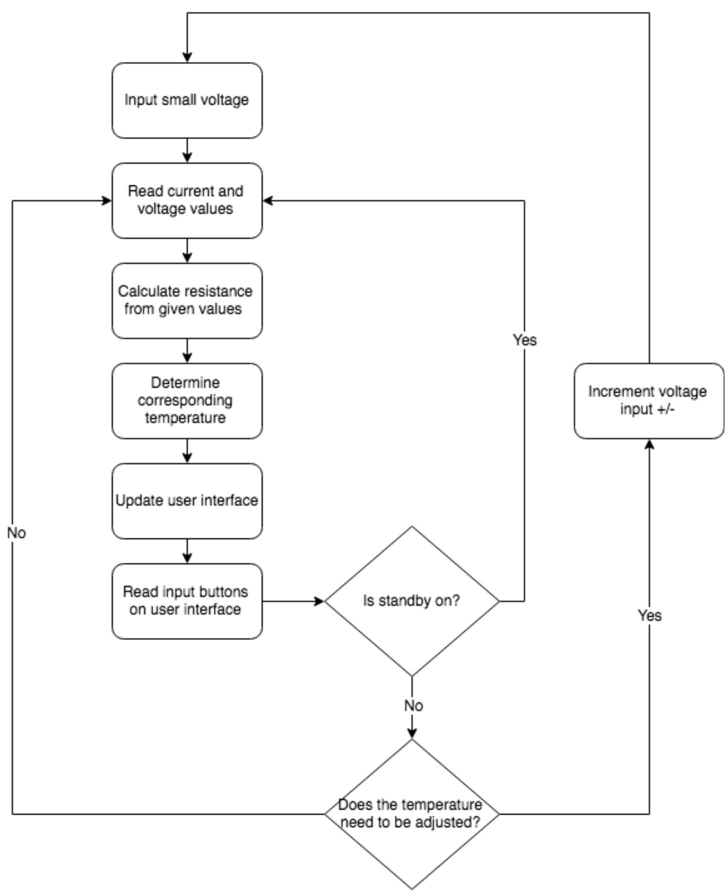


Figure 7.2: Software flow diagram

7.2 Financing

The financing for this project will be provided by the project sponsor, Dr. Reza Abdolvand. The projected overall budget for this project is estimated to be $500. Although, the objective is to determine the most cost-effective solution for the project. After further research and compilation of the potential parts, it is evident that approximately half of the budget can be allotted to equipment such as the microcontroller and display and the PCB boards. Remaining funds will be reserved for the order of miscellaneous parts due to unpremeditated decisions and changes regarding the design.

Table 7.1: Project Finances

Part	Manufacturer	Part Number	Quantity	Estimated Cost
Microcontroller	Texas Instruments	MSP430	1	$9.99
PCB Boards	Element 14	N/A	3	$50
Display	Adafruit	TC1602A-09T	1	$9.95
Miscellaneous parts for resonator	N/A	N/A	N/A	$50
Miscellaneous parts for DC to DC conversions and power supply	N/A	N/A	N/A	$25
Miscellaneous Parts for measuring voltage	N/A	N/A	N/A	$30
Shipping Costs				$50
Estimated Total Cost: $224.94				

The budget chart illustrated above shows the specific parts that will be ordered for intended use in the project. It is important to note that the table above will be updated throughout the design and prototyping phases, and the components listed may not be used in the final prototype depending on the testing results. The goal is to remain below budget throughout the duration of the project in order to simulate real-world experience of project management and budget planning.

7.3 Project Schedule

This section gives descriptive project goals for both semesters to ensure the project is completed in a timely manner.

Table 7.2: Fall 2018 Project Schedule

Week	Dates	Task(s)
1	8/20-8/26	Register as group, begin team discussion
2	8/27-9/2	Attend boot camp, start working on initial summary, schedule meeting with Dr. Abdolvand
3	9/3-9/9	Meet with Dr. Abdolvand; over half completion of initial summary
4	9/10-9/16	Complete 10-page Initial Project Summary
5	9/17-9/23	Meet to discuss details of Initial Project Summary
6	9/24-9/30	Update Initial Project Summary and resubmit
7	10/1-10/7	Complete 15 out of 45 pages for design document
8	10/8-10/14	Complete 20 out of 45 pages for design document
9	10/15-10/21	Complete 30 out of 45 pages for design document
10	10/22-10/28	Refine design document and begin selecting components
11	10/29-11/5	45-page design document due, finish schematic, order some components for testing
12	11/6-11/11	Complete 60 out of 75 pages
13	11/12-11/18	75-page design document due
14	11/19-11/25	Complete 80 out of 90 pages for design document
15	11/26-12/3	Finish 90 out of 90 pages for design document and submit
16	12/3-12/9	Final exams; final 90-page design document due

Table 7.3: Spring 2019 Project Schedule

Week	Dates	Task(s)
1	1/7-1/13	Reconvene group, plan prototyping, schedule meeting with Dr. Abdolvand
2	1/14-1/20	Meeting with Dr. Abdolvand, begin to order parts
3	1/21-1/27	Begin prototyping
4	1/28-2/3	Assemble prototype, begin testing
5	2/4-2/10	Continue testing, consider redesign
6	2/11-2/17	Redesign if necessary and reorder parts, schedule meeting with Dr. Abdolvand
7	2/18-2/24	Meeting with Dr. Abdolvand, reassemble prototype, begin testing
8	2/25-3/3	Continue testing, hone working prototype
9	3/4-3/10	Continue testing, hone working prototype
10	3/11-3/17	Finalize working prototype, assign roles regarding final document
11	3/18-3/25	Work on Final Documentation
12	4/1-4/7	Complete Final Documentation
13	4/8-4/14	Prepare for Final Presentation, Peer Reviews
14	4/15-4/21	Final Presentation (Exact Date TBD)
15	4/22-4/28	Final Exams

8. Conclusion

To complete this first phase of the project, the design and documentation of the design has been finished. Relevant constraints and standards have been examined, noted, and applied. Currently, all parts and their values have been chosen. However, resonator TCR characterization testing is still currently on hold due to the sample needing annealing for forming more Ohmic contacts. Thus, the circuit component values are estimates that will need to be recalculated later. For now, though, the values chosen for the resistors and voltage input range should be sufficient for initial testing of the project.

Because many of the components are a few millimeters in size and none of the project members have significant experience with soldering, there will not be breadboard testing done for this project at the request of the sponsor. Dr. Abdolvand would prefer for the team to order different iterations of the PCB, gaining experience with what works and what needs to be improved. The PCB design is essentially complete and only needs minor improvements before the first revision can be ordered. The first PCB ordered will have numerous test points to allow for analysis of each phase of the circuit.

The design for the hardware has been completed thus far. However, with different PCB iterations, the parts and configurations will need to be refined and reordered. Each new PCB ordered will have a different routing configuration, something that will need to be altered each time. The microcontroller has been wired according to current performance criteria; however, some pins may need to be reconfigured later.

Correspondingly, some basic decisions for the software have been made. Things like clock speed have been chosen already. But, there are many bits that will need to be set when the device is programmed. The algorithms for the programming have been designed and building the software will be a major portion of the design moving forward. The PID controller for the input voltage to the VCCS has been designed and will be implemented using programming during the operational testing of the circuit board.

Using a TPoS resonator from Dr. Abdolvand's lab to achieve ppb accuracy with this circuit is a unique design problem that requires a precise and accurate design. Thus, while the initial design has been completed, there is still need for testing and refinement that will explored as the project moves forward.

9. Appendices

This section is used to reiterate some common acronyms used throughout the document, list the references, and show the permissions requested to use images from other sources.

9.1 Nomenclature

AC: Alternating current

BOM: Bill of materials

CPU: Central processing unit

DC: Direct current

DCO: Digitally controlled oscillator

ECE: Electrical and computer engineering

IDE: Integrated development environment

JST: Japan solderless terminal

LCD: Liquid crystal display

LUT: Lookup table

MCU: Microcontroller unit

MEMS: Microelectromechanical systems

MSP: Mixed signal processor

OCXO: Oven-controlled crystal oscillator

PCB: Printed circuit board

PID: Proportional-integral-derivative

PPB: Parts per billion

RISC: Reduced instruction set computer

RoHS: Restriction of hazardous substances

TCF: Temperature coefficient of frequency

TCR: Temperature coefficient of resistance

TPoS: Thin-film piezoelectric-on-substrate

UCF: University of Central Florida

VCCS: Voltage-controlled current source

V_{pp}: Peak-to-peak voltage

9.2 References

[1] R. Abdolvand et al, "Micromachined Resonators: A Review," 2016.

[2] K. Khamoushi, "TEMPERATURE COEFFICIENT OF RESONANT FREQUENCY (TCF)," [Online]. Available: http://www.students.tut.fi/~khamousk/TCF.htm.

[3] J. Charmet, R. Daly, J. Thiruvenkatanath, J. Woodhouse and A. Seshial, "Observations of modal interaction in lateral bulk acoustic resonators," in *Applied Physics Letters*, 2014.

[4] S. Shahraini, R. Abdolvand and H. Fatemi, "Temperature Coefficient of Frequency in Silicon-Based Cross-Sectional Quasi Lamé Mode Resonators," in *IEEE International Frequency Control Symposium*, Olympic Valley, 2018.

[5] Texas Instruments , "MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller datasheet (Rev. J)," May 2013. [Online]. Available: http://www.ti.com/lit/ds/symlink/msp430g2553.pdf. [Accessed 26 October 2018].

[6] PC Magazine, "Definition of: transmissive LCD," [Online].

[7] Lumex Incorporated, "LCM-X01604DXX," 9 October 1998. [Online].

[8] Electronics Assembly, "LCD-Modul 2x8 - 5.01mm," October 2016. [Online].

[9] Adafruit, "Data Sheet TC1602A-09T SpecV00," 4 June 2009. [Online]. Available: https://cdn-shop.adafruit.com/product-files/181/p181.pdf. [Accessed 26 October 2018].

[10] Microtips Technology Inc. , "LIQUID CRYSTAL DISPLAY MODULE MODEL: NMTC-S20200BMNHSGW-12," 6 September 2017. [Online].

[11] "LCD-20x4Y: 20x4 Liquid Crystal Display with HD44780 driver," 2014. [Online].

[12] Newhaven Display International, "NHD-0216K1Z-FL-YBW Character Liquid Crystal Display Module," 26 March 2018. [Online].

[13] Renesas, "ISL28533, ISL28534, ISL28535, ISL28633, ISL28634, ISL28635," 13 January 2017. [Online].

[14] Analog Devices, "High Performance, Low Power, Rail-to-Rail Precision Instrumentation Amplifier," 2015. [Online].

[15] Analog Devices, "Low Noise, Low Gain Drift, G = 2000," 2012. [Online].

[16] Texas Instruments, "INA828 50-µV Offset, 7-nV/√Hz Noise, Low-Power, Precision Instrumentation Amplifier," January 2018. [Online].

[17] Texas Instruments, "INA12x Precision, Low-Power Instrumentation Amplifiers," January 2018. [Online].

[18] Texas Instruments, "Low Noise, Low-Distortion Instrumentation Amplifier Replacement for SSM2017 datasheet (Rev. C)," November 2015. [Online]. Available: http://www.ti.com/lit/ds/symlink/ina217.pdf. [Accessed 26 October 2018].

[19] Texas Instruments, "LM4040-N/-Q1 Precision Micropower Shunt Voltage Reference," June 2016. [Online].

[20] Texas Instruments, "LM4050-N/-Q1 Precision Micropower Shunt Voltage Reference," 2015.

[21] Analog Devices, "LT1236 Precision Reference," 1995. [Online].

[22] Analog Devices, "LT1021 Precision Reference," 1995. [Online].

[23] Diodes Incorporated, "ZHT431 ADJUSTABLE PRECISION ZENER SHUNT REGULATOR," April 2018. [Online].

[24] ON Semiconductor, "TL431A, B Series, NCV431A, B Series, SCV431A Programmable Precision References," October 2016. [Online].

[25] Bourns, "Potentiometers LINEARITY TECHNICAL NOTE," 2017. [Online].

[26] Bourns, "3549 - 10-Turn Precision Potentiometer," October 2014. [Online].

[27] Bourns, "3590 - Precision Potentiometer," November 2017. [Online].

[28] Vishay Spectrol, "7/8" (22.2 mm) Multi Turn Wirewound Potentiometer," 17 May 2018. [Online].

[29] ETI Systems, "MW22B," 24 February 1989. [Online].

[30] Bourns, "3386 - 3/8 ” Square Trimpot® Trimming Potentiometer," May 2016. [Online].

[31] CTS Electrocomponents, "Series 284 Data Sheet Precision Potentiometer," 22 March 2013. [Online].

[32] Agilent, "Agilent E3631A Triple Output DC Power Supply," July 2013. [Online]. Available: https://edisciplinas.usp.br/pluginfile.php/2405092/mod_resource/content/1/Agilent_E3631%20Power%20Supply_Users_Guide%20%284%29.pdf. [Accessed 26 October 2018].

[33] Department of Defense, *Environmental Engineering Considerations and Laboratory Tests,* G ed., 2008.

[34] National Institute of Standards and Technology, "Compliance FAQs: RoHS," Standards Information Center, 19 December 2017. [Online].

[35] International Electrotechnical Commission, "International Standard IEC 60950-1," December 2005. [Online]. Available: https://webstore.iec.ch/preview/info_iec60950-1%7Bed2.0%7Den_d.pdf.

[36] I. C. Society, IEEE Standard System C Language Reference, New York: The Institute of Electrical and Electronics Engineers, Inc., 2012.

[37] IEEE, ISO/IEC/IEEE 29199 International Standard, New York: The Institute of Electrical and Electronics Engineers, Inc., 2015.

[38] ISO/IEC, International Standard ISO/IEC 9899, New York, 2005.

[39] H. Hofstee, "Analog Oven-Control Circuit for a Thin-Film Piezoelectric-On-Substrate Microelectromechanical Systems Oscillator," Orlando, 2018.

[40] Texas Instruments, "LMZM23600 36-V, 0.5-A Step-Down DC/DC Power Module in 3.8-mm × 3-mm Package," April 2018. [Online]. Available: http://www.ti.com/lit/ds/symlink/lmzm23600.pdf?HQS=TI-null-null-mousermode-df-pf-null-wwe&DCM=yes&ref_url=https%3A%2F%2Fwww.mouser.com%2F.

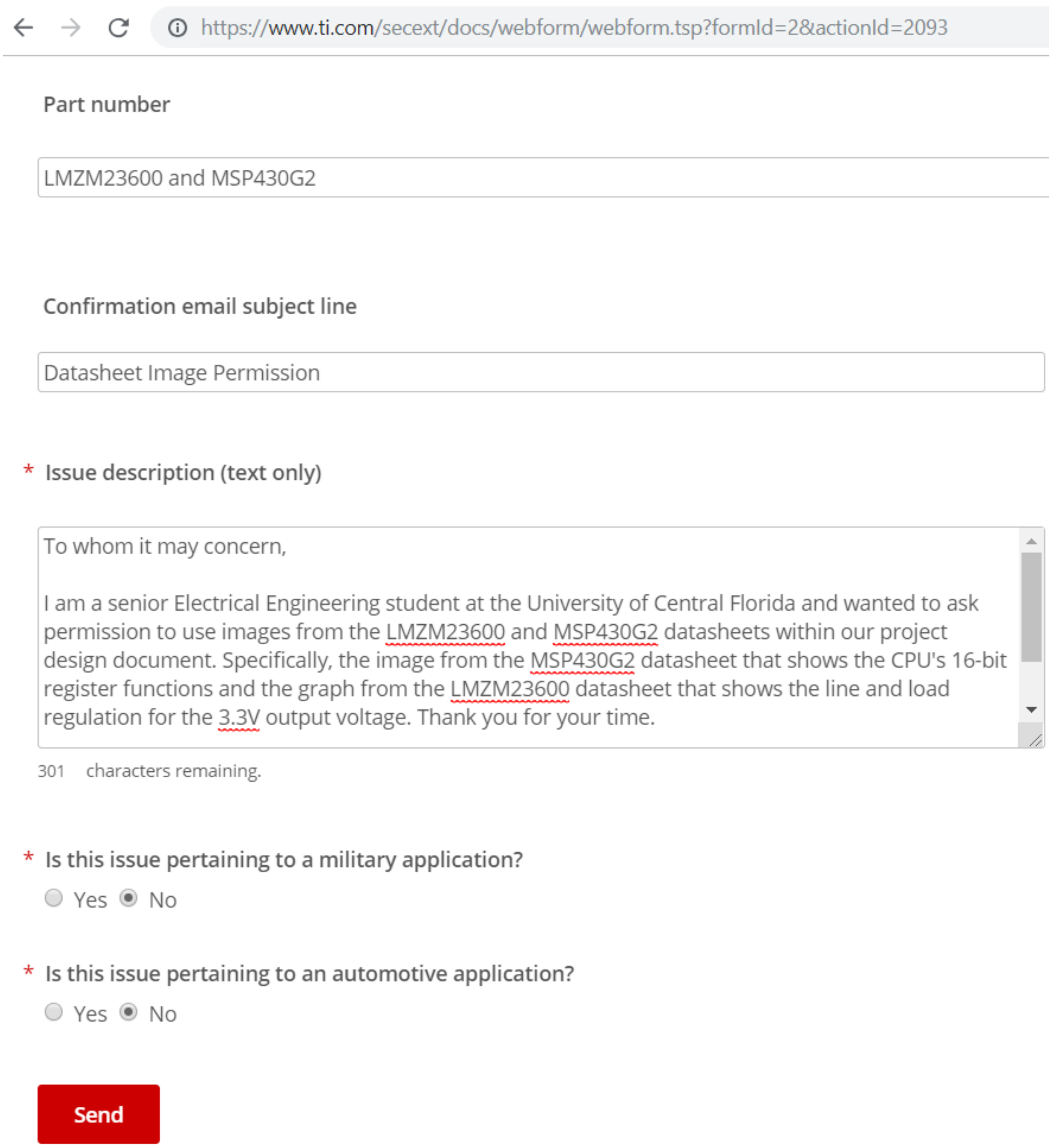
[41] Texas Instruments, "TPS6217x 3-V to 17-V, 0.5-A Step-Down Converters with DCS-Control™," May 2017. [Online]. Available: http://www.ti.com/lit/ds/symlink/tps62173.pdf.

[42] Texas Instruments, "OPA828 Low-Offset, Low-Drift, Low-Noise, 50-MHz, 36-V JFET-Input Op Amp datasheet," September 2018. [Online]. Available: http://www.ti.com/lit/ds/symlink/opa828.pdf.

[43] T. Gleditch, H. Hvims, R. Rörgren and J. Vähäkangas, "Chapter A: Wire Bonding," 2000. [Online].

9.3 Permissions

This section shows images of the permission requests to use manufacturer's images in this report.



https://www.ti.com/secext/docs/webform/webform.tsp?formId=2&actionId=2093

Part number

LMZM23600 and MSP430G2

Confirmation email subject line

Datasheet Image Permission

* Issue description (text only)

To whom it may concern,

I am a senior Electrical Engineering student at the University of Central Florida and wanted to ask permission to use images from the LMZM23600 and MSP430G2 datasheets within our project design document. Specifically, the image from the MSP430G2 datasheet that shows the CPU's 16-bit register functions and the graph from the LMZM23600 datasheet that shows the line and load regulation for the 3.3V output voltage. Thank you for your time.

301 characters remaining.

* Is this issue pertaining to a military application?

Yes No

* Is this issue pertaining to an automotive application?

Yes No

Send

-----Original Message-----
From: support@ti.com
Sent: 11/30/2018 03:15:16 PM
To: mmeganddriggers@aol.com
Subject: RE: Datasheet Image Permission

Dear Megan,

Thank you for contacting Texas Instruments. Your request has been received and Service Request number 1-4613040534 has been assigned to your inquiry.

Those datasheets that are publicly available are intended to use as a reference and resources. So, you are okay to use those images if it helps you to complete your project documentation.

Kind regards,

Julie Ann
Texas Instruments
TI Customer Support
Tel: +49 8161 80 2121 (International line)
00800 275 83927 (Free line, activated NOT in all countries)
00800 ASK TEXAS (Free line, activated NOT in all countries)
Product Information Center: http://www.ti.com/supportform

https://www.adafruit.com/contact_us

Please log in to provide us with more information.

Full Name: Megan Driggers *

Email Address: mdriggers@knights.ucf.edu *

Confirm E-Mail: mdriggers@knights.ucf.edu *

Category: Educator Inquiries

Educator's FAQ Page: https://www.adafruit.com/educators

Adafruit offers volume discounts and special offers to educators purchasing Adafruit and Arduino products. Please contact us if you're an educator and plan to place a large order for your students/workshops. The discount can be up to an additional 10% off plus quantity discounts depending on the purchase! Email us, we have team dedicated to working with you!

We work with universities, book stores, we accept purchase orders and for large orders we can also create custom packs depending on your needs. We offer quantity discounts as well as specific discounts for educators. Please contact us to learn about our special packs, discounts and more for workshops, classes and students!

the TC1602A-09T datasheet within our project design document.
Specifically, the image from the image showing the LCD connections

Send Message

Re: [[EDUCATOR INQUIRY]]

AI Adafruit Industries <support@adafruit.com>
Today, 11:00 AM
Megan Driggers

totally OK, please do!

On Thu, Nov 29, 2018 at 8:50 PM Megan Driggers <support@adafruit.com> wrote:

> contactname : Megan Driggers
> email address : mdriggers@knights.ucf.edu
> Message : To whom it may concern,
>
> I am a senior Electrical Engineering student at the University of Central Florida and wanted to ask permission to use an image from the TC1602A-09T datasheet within our project design document. Specifically, the image from the image showing the LCD connections to the MCU found at:
> https://cdn-shop.adafruit.com/product-files/181/p181.pdf
> Thank you for your time.