Smart Recloser

A Major Qualifying Project Report

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

in Electrical and Computer Engineering

Submitted by:

Salvador Antoniou ____________________

Thomas Cucinotta ____________________

James Silvia ____________________

Date: April 28, 2011

Approved by:

____

Professor Alexander E. Emanuel - Advisor

ABSTRACT

The goal of this project was to develop an intelligent recloser system that is able to differentiate between a temporary and a persistent fault, and then take proper action. The intended use of this system is to increase reliability and energy quality. The system was realized by using a non-invasive method to monitor the soundness of the line. Through signal processing, this information is converted into logic that a microprocessor can use to control the recloser.

ACKNOWLEDGEMENTS

First and foremost, we would like to express our gratitude to Professor Alexander E. Emanuel for this eternal patience, guidance, and support throughout the project. We would also like to thank Professors Stephen Bitar and James O'Rourke for their troubleshooting assistance. Next, we would like to thank Tom Angelotti and Patrick Morrison of the ECE Shop for providing us with the equipment needed to complete this project.

Last but not least, we would like to thank the ECE department at WPI for providing us with valuable knowledge throughout the course of our undergraduate studies.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

I. INTRODUCTION

At the turn of the 20th century and into the modern age, electricity has played a crucial role in the development of societies around the world. As it continues to proliferate, it has become imperative that power distribution be as reliable and safe as possible. Of the many modern tools that exist in aiding this goal, the type focused on in this project is the recloser.

The primary method for power distribution in the United States and Europe involves the use of a power grid that uses large metal conductors elevated at or above tree lines. As much of the grid is openly exposed to the elements, it is subject to extreme weather conditions, animals, trees, and humans. One major problem that arises from this interaction is a fault. A fault is an irregularity in the distribution system; typically a large surge in energy that can lead to unexpected behavior on a subscriber's end. An example of such a situation involves a lightning strike on a line. This can cause a surge in power being delivered to a manufacturing plant involving the use of heavy machinery. The sudden increase in power can cause the machinery to behave irregularly and pose a fatal threat to workers. Thus, reclosers are a remote tool implemented in the power distribution system to counteract these irregularities and protect the system by acting as a circuit breaker and disconnecting the afflicted power line.

Many of the reclosers in the nation's aging power distribution system are of the autorecloser type. As the name indicates, they are automatic and function by attempting to reconnect a line periodically a set number of times after a fault has been detected. Once the number of tries is exceeded, they usually require manual labor to reset. The limitation of this implementation lies in that a fault is not always temporary, some are persistent.

Thus, this paper focuses on the development of a "smart" recloser that is able to tell the difference between a temporary or persistent fault. Then once the fault clears, it reconnects the line, otherwise it will stay disconnected. This is accomplished by using a transducer to monitor the power line and then processing the signal to determine whether a fault is occurring or not. If a fault has been detected, the system switches the power line into a high impedance mode where a large resistor is connected in series with the line to limit the current flow. While this is done, the transient of the line is monitored in real time. If the transient returns to a normal level, then the system will know that the fault has cleared and restore normal operation, otherwise, it will continue to monitor the transient in the high impedance mode.

II. BACKGROUND

The electrical distribution system powering today's society is an intricate array of devices that are tasked with maximizing power quality and uptime for the consumer. One major issue that distributions systems face is the occurrence of a fault. These can be detrimental to the consumer and supplier of the system. In order to protect against faults, devices called reclosers are implemented along the feeder lines in order to act as circuit breakers of the transmission line.

2.1 - TYPICAL POWER DISTRIBUTION SYSTEM

Electrical energy is generated by means including burning fossil fuels such as oil, gas and coal as well as renewable energy sources. Once generated, the electricity is delivered through a distribution system. Power distribution systems are comprised of generators, substations and multiple feeders. Figure 1 depicts a typical power distribution system from generation to branch systems downstream.

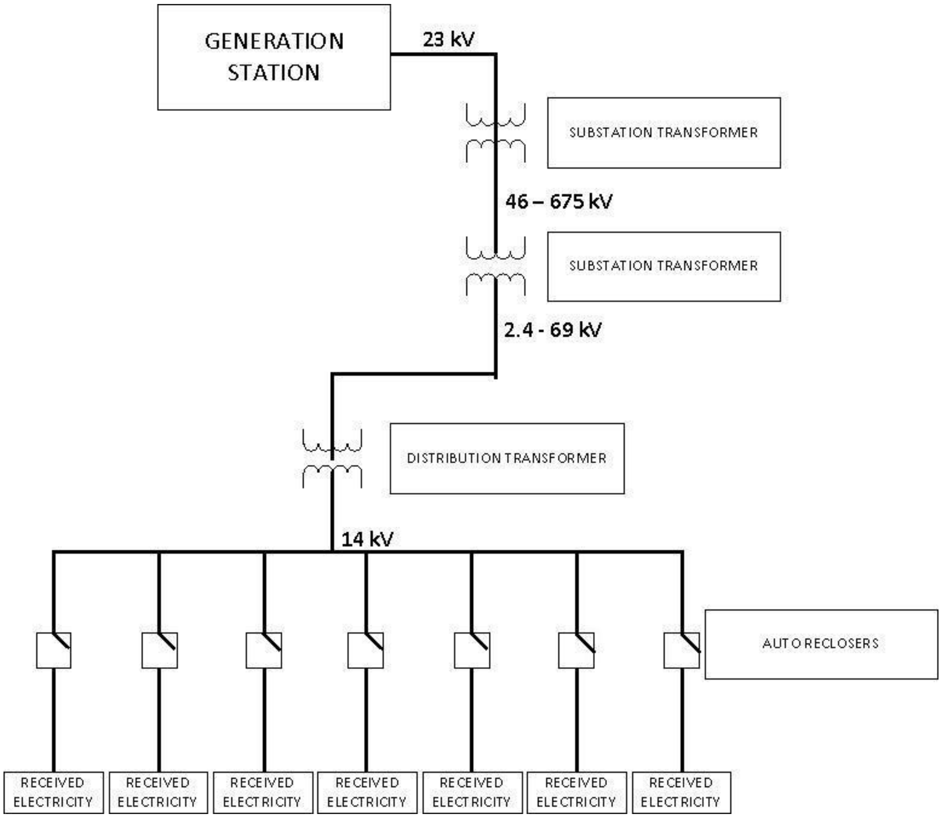


FIGURE 1 - POWER DISTRIBUTION SYSTEM

The generators located at utility plants are the first step in the power distribution system, and usually output between 11 and 23 kV. [1] The typical characteristics of generated electricity are sinusoidal waves with a frequency of 60 Hz, AC Voltage. Following the generation of electrical energy the voltage is stepped up, by the use of a transformer in a substation. This is located at the utility plant, for transmission. Typical voltages of 46 kV, 69 kV, 115 kV, 138 kV, 161 kV, and 230 kV are used in this process. [1] The voltage is stepped up to a much greater magnitudes because this magnitude decays with distance. Other substations are employed in the distribution system to alter the voltage levels to most efficiently transmit the voltage along the varying distances.

These transmitted voltages are then stepped down by transformers located at the beginning of the feeder lines. Feeder lines distribute electricity directly to various subscribers such as factories, manufacturers, as well as other industrial complexes. The primary distribution system voltages vary between 2.4 kV and 69 kV, depending on the standards and needs of the power lines. [1] Distribution transformers are placed along distribution lines, to further step the voltage down to 14kV, as well as intermediate voltages for shorter distance transmission. [1] The distribution transformers can be seen along power lines and on telephone/power line poles. An image of an overhead transformer can be seen in Figure 2.

FIGURE 2 - OVERHEAD TRANSFORMER [2]

Distribution systems can effectively provide electricity to its subscribers by these means. However, this does not occur without various issues. One issue that is present in distribution systems is the occurrence of faults.

2.2 - FAULTS

A fault is a large surge of current along the transmission line, and it can damage many aspects of the distribution system. There are many different types of faults that can appear along power distribution lines at any time during their normal operation. The common faults include transient, persistent, and asymmetric.

TRANSIENT FAULT

Transient faults occur by shorting a feeder line with the ground or with another feeder line. Transient faults are common in overhead power lines, because they are subjected to the environment in which they are erected. Transient faults are commonly caused by trees, animals, lightning, and a conductor clash. [1]

PERSISTENT FAULT

Unlike the transient fault, a persistent fault will not disappear when the power is disconnected. Power lines that are often underground are at risk for a persistent fault, but also have limited risks of faults from nature such as lightning or trees. [1] The reason that persistent faults are most common among underground power lines is because these power lines are encased. Since there is an encasing around the power lines it is very rare that a transient fault will occur, and more likely that the type of fault that is present is a persistent fault.

ASYMMETRIC FAULT

An unbalanced or asymmetric fault differs in its effect on each of the three phase lines. Common asymmetric faults include phase to ground, phase to phase, double phase to ground, etc. [1]

- Phase to ground: a short circuit between one phase line and one ground
- Phase to phase: a short circuit between lines, through physical contact or air ionization
- Double phase to ground: two lines could come into contact with the ground and caused a short circuit

2.3 AUTO-RECLOSER

An auto-recloser is a type of circuit breaker that commonly has a mechanical device or other mechanism that automatically closes the circuit breaker after being tripped by a fault. An auto-recloser would make a few attempts to reclose the circuit (typically three) to reenergize the line. If the fault is persistent and were to remain along the line, after the several attempts to reclose the circuit, the auto-recloser would remain open until a technician can manually close the circuit.

Auto-reclosers are made in both single-phase and three-phase and can also operate by means of oil, or a vacuum, along with mechanical devices that physically disconnect the transmission line. They are rated for voltages between 2.4 kV and 38 kV, load currents between 10 and 1200 A, and fault currents up to 16 kA. [2] & [3] Shown below is an image of what an auto-recloser looks like that can commonly be seen on overhead power distribution lines.

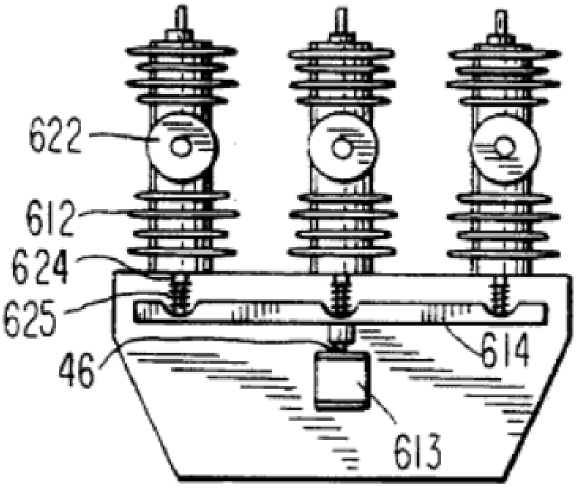
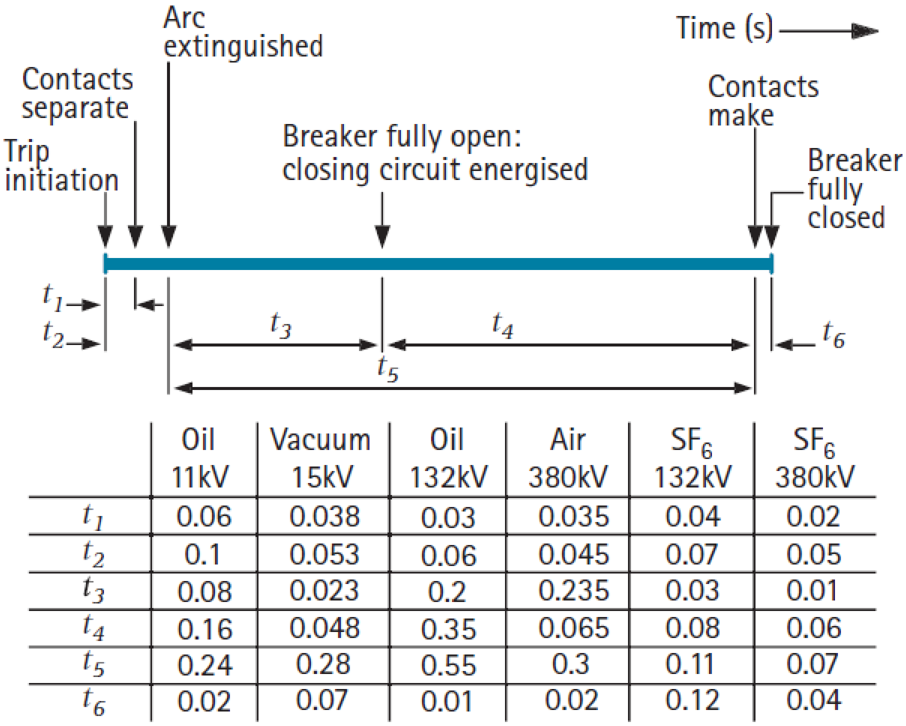


FIGURE 3: AUTO-RECLOSER IMAGE [3]

The auto-recloser's operating time can be broken up in to several different sections beginning at the initial time (t_0) of a fault. They include the contacts of the recloser separating, the arc being extinguished; time the circuit breaker is open, the circuit being re-energized, and the closing of the contacts. The figure below breaks up the different recloser actions, and describes how long they each take for different types or reclosers.



	Oil 11kV	Vacuum 15kV	Oil 132kV	Air 380kV	SF_6 132kV	SF_6 380kV
t_1	0.06	0.038	0.03	0.035	0.04	0.02
t_2	0.1	0.053	0.06	0.045	0.07	0.05
t_3	0.08	0.023	0.2	0.235	0.03	0.01
t_4	0.16	0.048	0.35	0.065	0.08	0.06
t_5	0.24	0.28	0.55	0.3	0.11	0.07
t_6	0.02	0.07	0.01	0.02	0.12	0.04

FIGURE 4: AUTO-RECLOSER REACTION TIME [3]

The time needed to break the circuit following a fault to extinguishing the arc (t_1-t_2) can take between 0.07 s and 0.16 s. The pre-arranged time for waiting to see if the fault is clear varies, but the time to energize and reclose the circuit fully (t_4-t_6) can take between 0.17 s and 0.91 s. At the end of this report, these times will be compared to the prototype.

III. PROBLEM STATEMENT

When reclosers are operating by mechanical means there is no way to determine if the attempt to reconnect the circuit is occurring after the fault has cleared or not. The power distribution system shown earlier in Figure 1 is broken down to display the focus of the MQP, as seen below:

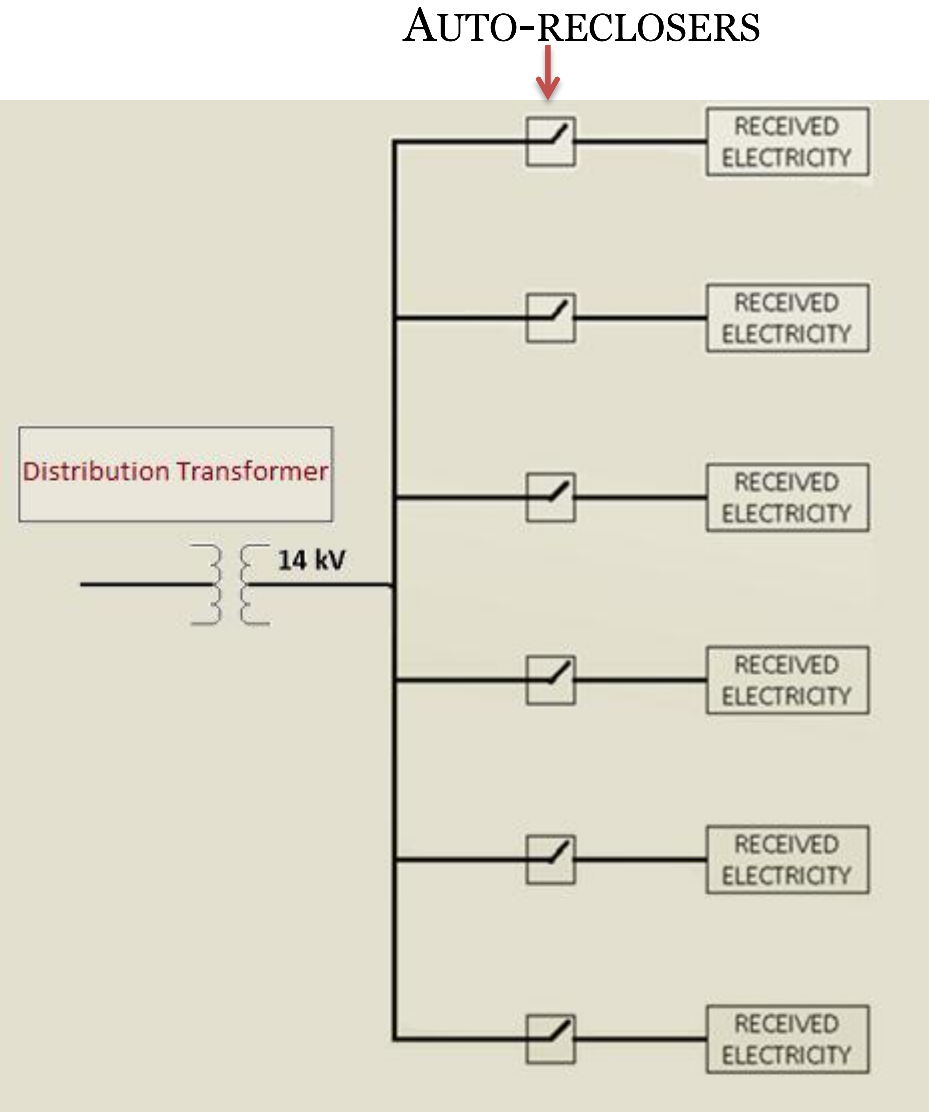


FIGURE 5: DISTRIBUTION SYSTEM- MQP FOCUS

The major issue of attempting to reclose a feeder line prior to the fault clearing the line can be seen by examining Figure 6, and Figure 7.

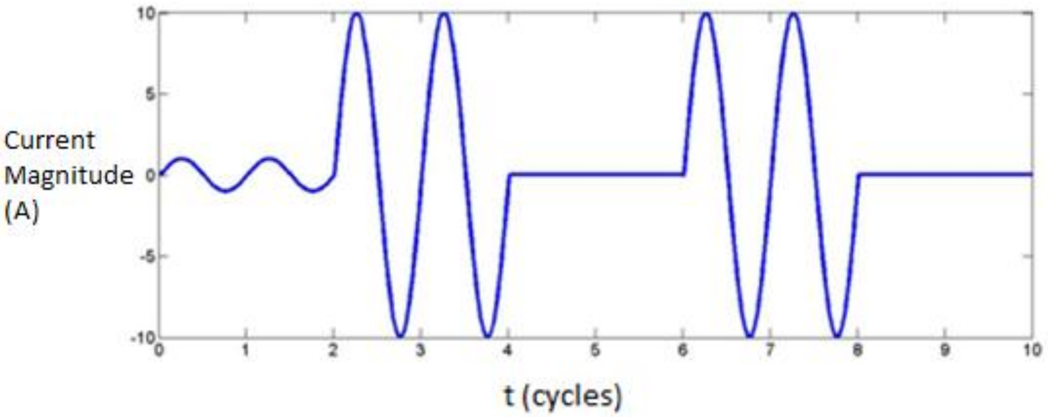


FIGURE 6: CURRENT

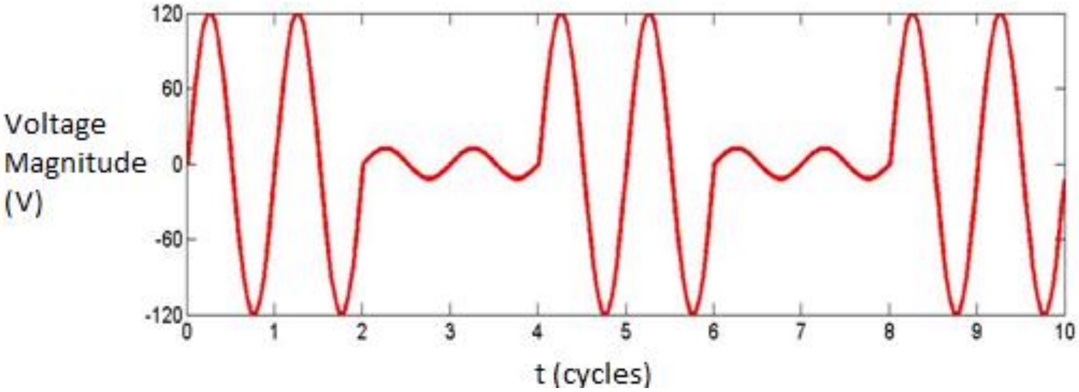


FIGURE 7: VOLTAGE

Assuming a fault was to occur on the first feeder line in Figure 5, this would cause the auto-recloser to open in order to restore normal operation to the remaining feeder lines. Shown above is the current and voltage during a fault situation along a power distribution line. Figure 6 represents the current along the feeder line during a fault, and Figure 6 represents the voltage along the other feeder lines that do not have a fault.

The first two cycles, labeled 0-2 along the t axis, represents the current and voltage during normal operation. The voltage along the other feeder lines is under normal conditions until the fault appears along the first feeder line, the start of the second cycle on Figure 6. At the time of the fault, the voltage shrinks along all of the other feeder lines. When the auto-recloser recognizes that a fault has occurred along the line, it opens to restore normal operation to the remaining feeder lines, shown at the start of the 4th cycle. However, normal auto reclosers attempt to close after a set amount of time, without knowing if the fault is present or clear. If the fault is persistent, it causes a repetitive shrinkage and growth of voltage on the other feeder lines. This shrinkage and growth of the voltage can be very harmful. This is because typically factories or machineries have equipment that is designed with the intent of operating a specific voltage. With a changing voltage as shown in Figure 7, property damage could occur to this equipment, or the necessary function of this equipment may be rendered useless.

OUR GOAL

The goal of this MQP is to develop a “smart” recloser prototype that is capable of detecting a fault while also determining if the fault has cleared from the power distribution line. Using this information, the recloser will be able to determine the appropriate time to reclose the transmission line, thus protecting the other feeder lines from inconsistent voltage transmission. This will improve the quality of electricity delivered to subscribers, and also reduce the amount of manual labor hours required to maintain the current auto reclosers.

III. THE SMART RECLOSER – OVERVIEW

The following section provides an overview of the Smart Recloser system. The full system block diagram is introduced with a brief summary of the several major functional blocks. The theoretical behavior of the system is then described so that the reader has an understanding of its proposed operation.

3.1 BLOCK DIAGRAM OVERVIEW

The Smart Recloser device consists of the testing apparatus and four major functional blocks; the Transducer, Circuit 1, Circuit 2, and the MSP430 Block. The block diagram can be seen in Figure 8.

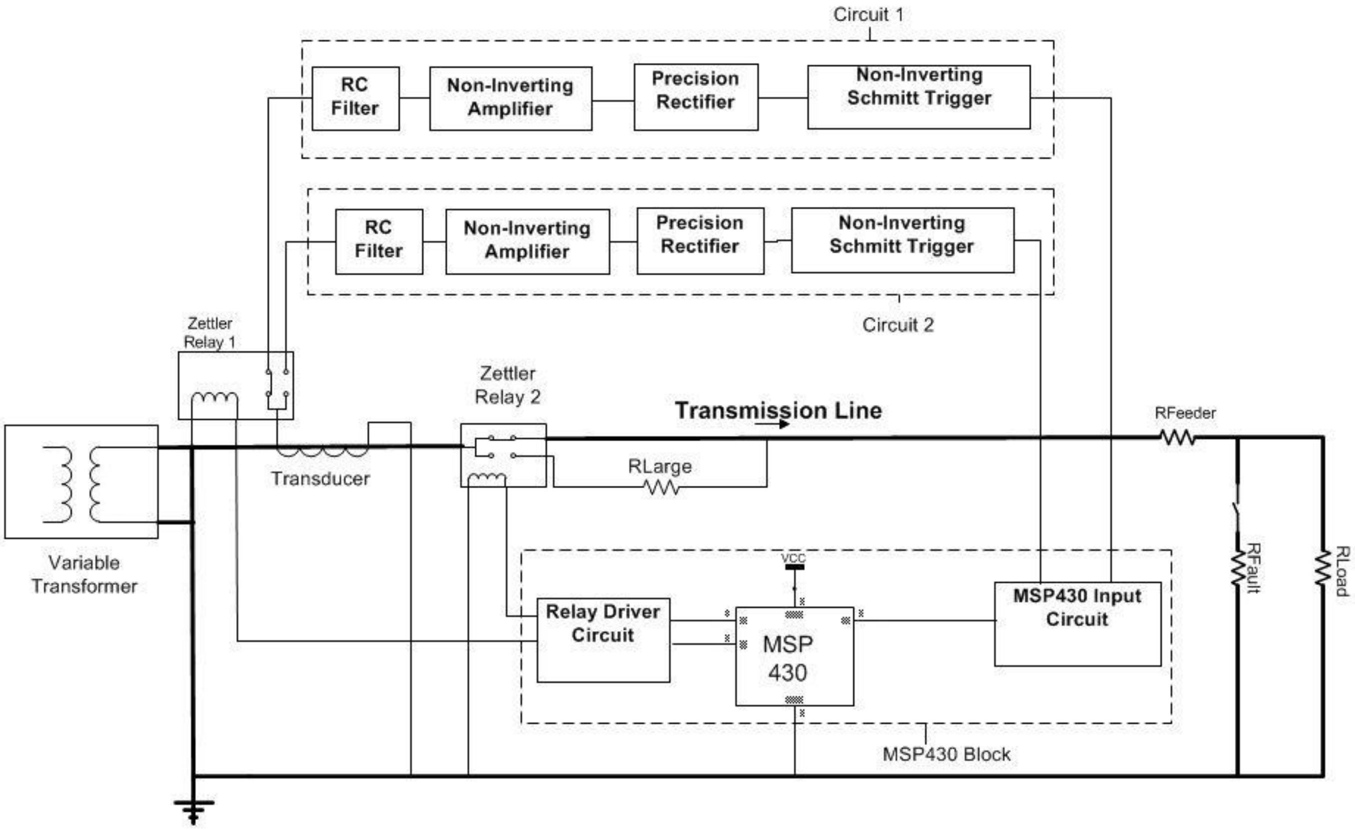


FIGURE 8 - SMART RECLOSER BLOCK DIAGRAM

TESTING APPARATUS

For designing purpose, a testing apparatus was created to simulate faults at lower voltage and current levels when compared to a real electrical distribution system. The testing apparatus allows for the design and implementation of a scale model solution. Repeatable, consistent behavior that was portable was a key factor in its design. The testing apparatus is comprised of a variable transformer, a power relay, and a power resistor array partitioned into three parts: R_{Large}, R_{Load}, R_{Fault}.

TRANSDUCER

The transducer represents the first functional block of the design. Its purpose is to monitor the transmission line and provide a signal corresponding to its behavior. The main advantage of utilizing this method for monitoring the transmission line is that the transducer provides a means of non-invasive information gathering. For the transducer to function, it must be within close proximity to the transmission line. If this condition is met, the device will produce an output that varies based on the operation of the line.

CIRCUIT 1

The next major block is Circuit 1. The purpose of this block is to process the signal provided by the transducer and to produce a binary output that indicates whether or not a fault has occurred. When no fault is detected, Circuit 1 produces an output of “logic low.” Conversely, when a fault occurs, it produces a “logic high.” This circuit is only active during normal operation or pre-fault conditions. The reason for this is that when a fault occurs, the system engages a high impedance mode. The change in mode causes a large change in the transducer output, and thus the threshold voltages needed to monitor the fault. Therefore, Circuit 1 is specifically calibrated for the threshold voltages during normal operation. It is comprised of a filter, non-inverting amplifier, precision rectifier, and Schmitt Trigger.

CIRCUIT 2

Since the system has two modes of operation, a second signal processing circuit was implemented. Circuit 2 is specifically calibrated to operate during fault conditions and high impedance mode. It contains the same basic topologies as Circuit 1, however these sub-blocks are designed around different threshold voltages. It maintains the same purpose as its sister circuit with the stipulation that it provides a binary output based on whether or not the fault has cleared. It continues to output a logic high for as long as the fault persists, and then a logic low if the fault clears.

MSP430 BLOCK

The MSP430 Block provides the system with a control system that interprets the outputs of the signal processing circuits and triggers actions corresponding to each of two output cases. The block is comprised of three sub-blocks; an Input Circuit, the MSP430 itself, and a Relay Driver circuit. The input circuit scales down the outputs of Circuit 1 and 2 to within the limits of the MSP430 and merges them into a single input since only one is active at a given time. Next, this output is processed by the MSP430 whose operation will be discussed in much greater detail in the corresponding section. Lastly, the Relay Driver circuit was implemented to operate the two power relays in the system.

3.2 SYSTEM OPERATION

The Smart Recloser system has two modes of operation, normal and high impedance mode. When in normal mode, Relay 1 is defaulted to connecting Circuit 1. The transducer's output is then processed through Circuit 1. While conditions are normal, Circuit 1 will continuously output a logic low. When a fault occurs, it will respond by outputting "logic high". This output is then scaled down and fed into the MSP430. Upon detecting "logic high", the MSP430 outputs two signals. The first signal is to trigger Relay 2 into a high impedance mode where R_{Large} will be connected in series with the main line to limit the current. The second signal, delayed by approximately 1 second, will trigger Relay 1 to switch to Circuit 2 to monitor the fault under high impedance mode.

Under this condition, the system has entered its second mode of operation, high impedance mode. Circuit 2 will output “logic high” as long as the fault persists and keep the high impedance mode engaged. If the fault is temporary and clears, Circuit 2 will then output “logic low” to the MSP430. The MSP430 will read the signal and wait 10 seconds to make certain that the output from Circuit 2 is consistent. If at the end of these cycles it is still “logic low”, then the MSP430 will output two signals again. The first will reengage Circuit 1, and the second will reconnect the transmission line.

3.3 SYSTEM SIMULATION

Basic models for a simplified autorecloser and smart recloser were developed in PSPICE to observe their characteristics. First, a basic autorecloser model was created. Figure 9 was the model used to simulate this device.

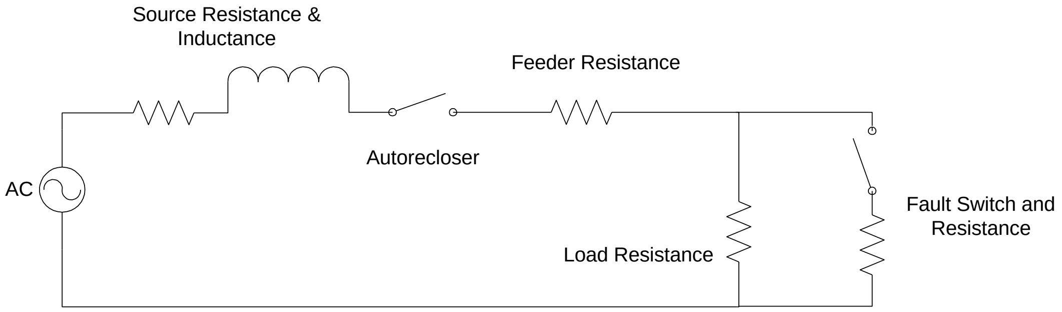


FIGURE 9 - AUTORECLOSER PSPICE MODEL

The source resistance and inductance are characteristics of a transformer which is symbolized by the AC source. A voltage controlled switch is used to simulate the autorecloser reclosing periodically when a fault is triggered. The feeder resistance simulates the non-ideal resistance in a transmission line.

The load resistance is connected for normal operation and the fault resistance is connected by a switch in parallel to simulate an increased draw in current from a fault.

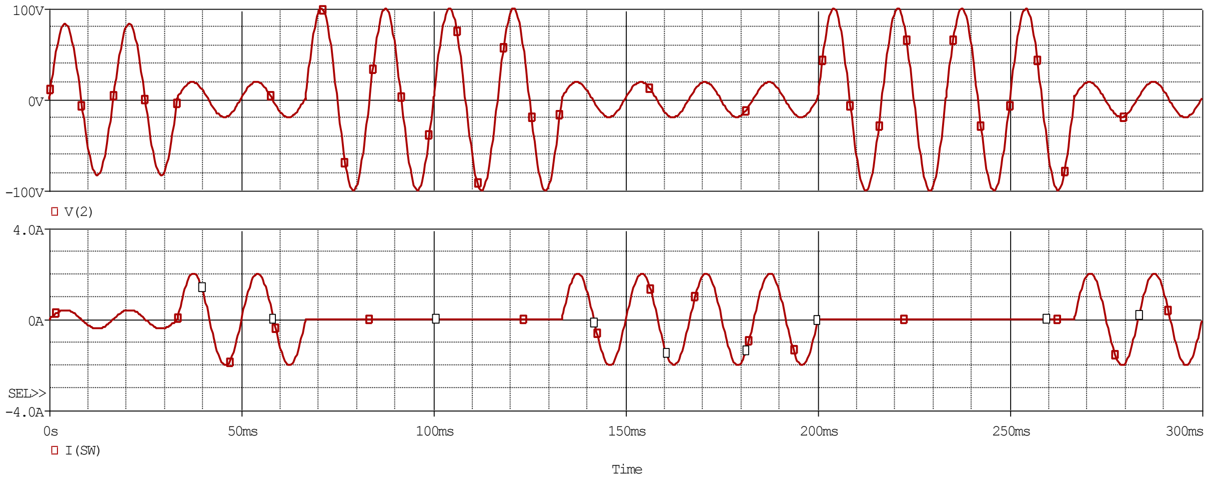


FIGURE 10 - AUTORECLOSER BEHAVIOR

Figure 10 contains the characteristics obtained from the autorecloser simulated with a persistent fault. For the first two cycles, the system is running normally, however at 66ms, a fault is triggered. A sag in the voltage is seen while there is a corresponding surge in current. Over the next two cycles, the autorecloser trips restoring the voltage and cutting the current. Periodically, it recloses to check if the fault has dissipated causing the voltage to sag since it is a persistent fault.

The system is next outfitted with a simplified model of a smart recloser in Figure 11. A high impedance is connected in parallel with the switch to simulate the high impedance mode of the device. The recloser switch is now different from the first version in that it has a different set of operating conditions. There are now two sets of threshold voltages that govern its operation; one for when the fault is detected, and one for high impedance mode.

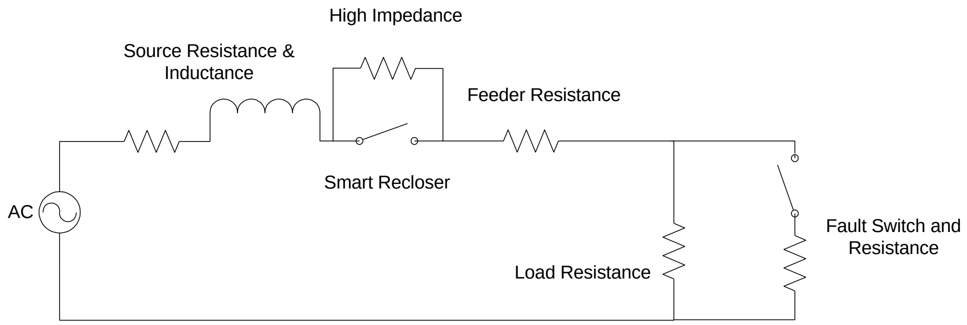


FIGURE 11 - SMART RECLOSER PSPICE MODEL

Figure 12 contains the behavior of the system. When the fault is triggered, the system responds by putting the system in high impedance mode limiting the current and preserving the peak voltage on the line. For the duration of the fault, the line is monitored with limited current, until the fault clears. When this condition is met, the system reconnects the line and reverts to normal operation.

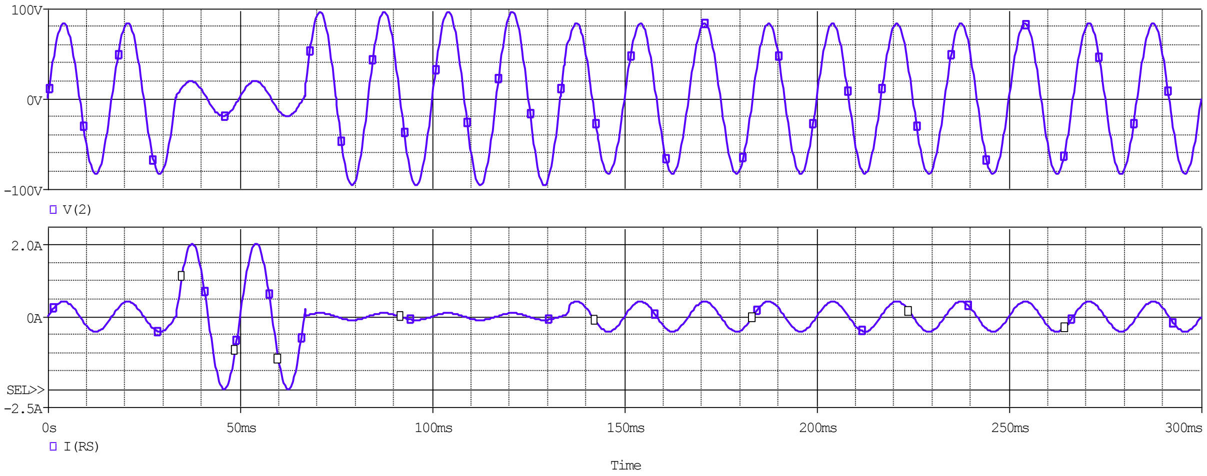


FIGURE 12 - SMART RECLOSER BEHAVIOR

IV. THE SMART RECLOSER – SYSTEM DESIGN

The following details the design process for the Smart Recloser device. The major functional blocks summarized in the previous section will be dissected into their individual elements and their implementations are discussed. To conclude, a performance summary of each of these elements is provided with collected data from the prototype.

4.1 TESTING APPARATUS

In order to begin designing the Smart Recloser system, the specific parameters of a power distribution network needed to be determined so that normal baseline and fault behavior could be established. Since it is not practical to work at the high voltage and current values used in a real power system, a downsized testing apparatus was constructed, pictured in Figure 13.

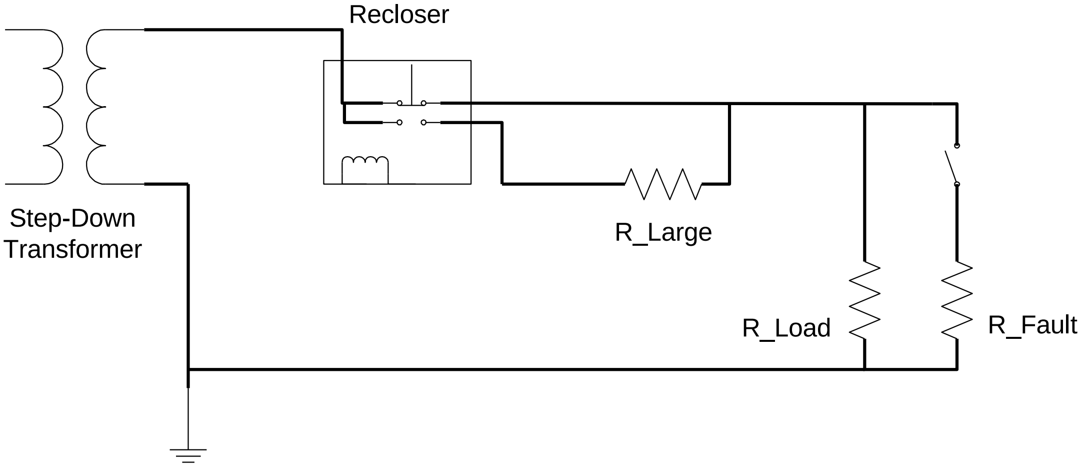


FIGURE 13 - TESTING APPARATUS DIAGRAM

The model is comprised of five main elements: a transformer, a recloser, a high impedance, a load, and a fault. The substation transformer of a real power system was substituted by a small step down transformer that can be plugged into a wall outlet. The recloser was substituted by an electromagnetic relay that will switch between a normal mode and high impedance mode when triggered by the MSP430. A power resistor array was used to implement the final three elements R_{Large}, R_{Load}, and R_{Fault}. R_{Load} was used to draw current and simulate normal operation for the system. A fault was simulated by connecting R_{Fault} in parallel via a switch to lower the resistance and increase current. Lastly, R_{Large} was multiple power resistors in series to limit the current when high impedance mode was engaged.

With this model, a set of operational constraints was then developed based on available products and budget constraints. The testing apparatus was being designed to be portable and compatible with 120V wall outlets. The transformer must be able to supply a minimum of 100V and a current of at least 2.5A for a fault. Normal power line operation would be characterized as 0.5A through the primary load resistor. Since a fault was being set as 2.5A the power resistors used for the faults would need to be able to dissipate up to at most 250 watts for up to a second if all the current were flowing through one resistor. The electromagnetic relay would need to be able to handle disconnecting a line at 100V, 2.5A to engage the high impedance mode. Lastly, while faults normally occur unexpectedly in real distribution systems, the testing apparatus must have an expected method for triggering a fault. Therefore, a fault switch is needed to connect the power resistors in parallel for this purpose. The switch must be rated for at least 100V at 2.5A as well.

TRANSFORMER

Given the constraints of the transformer, a review of current products yielded a very small list of cost effective solutions. Ultimately, a 130V, 3A variable transformer was selected because this would be able to supply the desired 100V at 2.5A for a fault. Table 1 is a summary of the product specifications.

Part Number: SC-3M Variable Transformer	
Input	110V AC
Output	0 - 130V AC
Max Current	3A, fused
Dimensions	5.35'' x 4'' x 5.5''
Weight	6.5 lbs
Source	allelectronics.com

TABLE 1 - VARIABLE TRANSFORMER CHARACTERISTICS

It should be noted that there was a slight performance issue with this product. The voltage at the output was typically 5V higher than the supposed voltage on the dial.

ELECTROMAGNETIC RELAY

For the recloser, a Zettler AZ 735 electromagnetic relay was used. Rated for 120V AC at 15A, this meets the requirement of 100V at 2.5A. The main reason for using these was that they were in abundant supply at the WPI ECE Shop and were available free of charge. Table 2 contains the important specifications. It should be noted that this model is no longer in circulation and its replacement with similar characteristics is the Zettler AZ 755.

Model: Zettler AZ 735	
Max Switched Current	20 A
Max Switched Voltage	380V AC
Dielectric Strength	5000 Vrms
Coil Operation Time	8 ms
Coil Release Time	5 ms
Coil Resistance @ 9V DC	155Ω +/-10%
Minimum Coil Trip Voltage (Measured)	7.2V DC

TABLE 2 - ZETTLER RELAY SPECIFICATIONS

The coil operation times and resistance will become important when calculating the response time of the entire system and the design of the relay driver circuit. For additional specifications, the Data Sheet can be found in Appendix F.

POWER RESISTOR ARRAY

The power resistor array was comprised of five adjustable power resistors. They are manufactured by Huntington Electric Inc. and have a maximum resistance of 200Ω with a power rating of 225 Watts. Figure 14 is an image of the layout used for the prototype. Set 1, marked in blue represents the load resistor. Set 2, marked in red represent the fault resistors connected in parallel via a switch. Lastly, set 3, marked in green, represent R_{Large} for high impedance mode. Table 3 contains a summary of specifications.

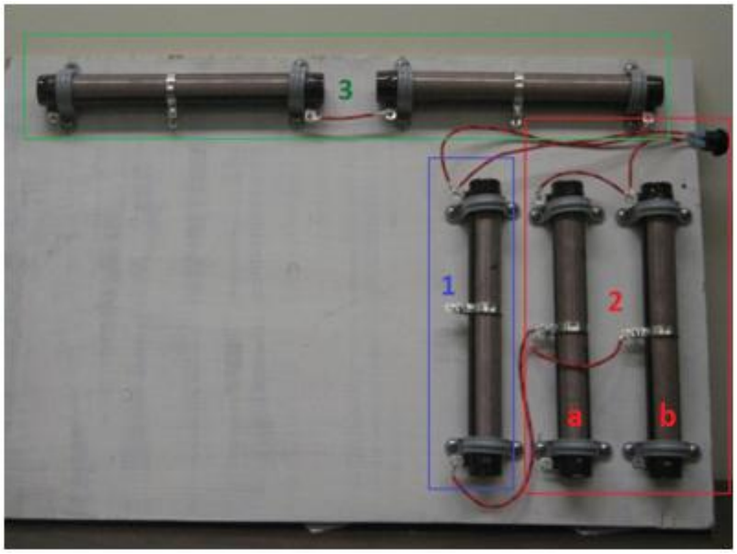


FIGURE 14 - POWER RESISTOR LAYOUT

Part Number: AVT200-200-ND	
Resistance	200 Ω
Tolerance	+/- 10%
Power	225 Watts
Dimensions	1.13'' Dia x 10.5'' L
Source	Digikey.com

TABLE 3 - POWER RESISTOR SPECIFICATIONS

Once the array was built, the values of each of the resistances were measured to verify that they were set appropriately so that the prototype would behave as expected. Table 4 contains a summary of measured values. The Data Sheet for the product is in Appendix F.

Measured Resistance Values	
R_{Load} (Blue)	195.63 Ω
$R_{a\ Fault}$ (Red)	50.31 Ω
$R_{b\ Fault}$ (Red)	50.32 Ω
R_{Fault} (Red + Blue)	40.07 Ω
$R_{High\ Impedance}$ (Green)	389.65 Ω

TABLE 4 - POWER RESISTOR ARRAY MEASURED VALUES

Using Ohm's Law and these measurements, the predicted currents flowing in various conditions are summarized in Table 5. The actual current measurements are included as well for comparison.

Resistor Configuration Currents		
Resistor Configuration	Theoretical Current (100V Applied)	Actual Current (100.39V Applied)
Normal Operation (Blue)	0.511 A	0.49 A
Fault Mode (Red + Blue)	2.495 A	2.47 A
Fault in High Imp (Red + Blue + Green)	0.233 A	0.21 A
Cleared Fault High Imp (Blue + Green)	0.171 A	0.15 A

TABLE 5 - POWER RESISTOR ARRAY PERFORMANCE

In summary, while all of the actual currents were a bit lower than their expected values, they are very close to their theoretical values and sufficient for the purposes of this design.

FAULT SWITCH

The last component used in the test apparatus was a switch to set off the fault. The following switch manufactured by Cherry was used. Table 6 summarizes its pertinent specifications.

Part Number: CH945-ND	
Contact Rating	10 A
Voltage	125 V AC
Electrical Life	10,000 cycles
Source	Digikey.com

TABLE 6 - SWITCH SPECIFICATIONS

4.2 TRANSDUCER

The device needs a means of determining if a fault is present on the transmission line or not, by means of a non-invasive method. A transducer was employed to perform this function. The transducer will take advantage of the fact that the transmission lines create an electromagnetic field. By coupling the transmission line with another coil the transducer can monitor the current on the transmission line in a non-invasive manner.

When a current (I) is passing through the transmission line magnetic fields are produced. The strength of the magnetic field (H) can be determined using the Equation below, where d is equal to the diameter of the transmission line:

$$\mathrm{H}= I/(2\pi d) \tag{1}$$

By multiplying the strength of the magnetic field (H) by the permeability of air (μ_o), the magnetic flux density (B) can be determined. This is measured in Teslas. This equation can be seen below:

$$B=\mu_0 H \tag{2}$$

Since a magnetic flux density is present using a pickup coil positioned perpendicular to the magnetic flux a voltage will be induced upon the coil that is proportional to the current on the transmission line. By monitoring this voltage a fault can easily be determined.

FINAL PRODUCT

The design of this transducer began with a 2" diameter wooden core. This was very important to our design because typically coils like this are made with iron cores, and this causes hysteresis to occur on the signal. This will disrupt our signal and cause it to be inaccurate. The core dimensions can be seen in Figure 15. Values are in millimeters.

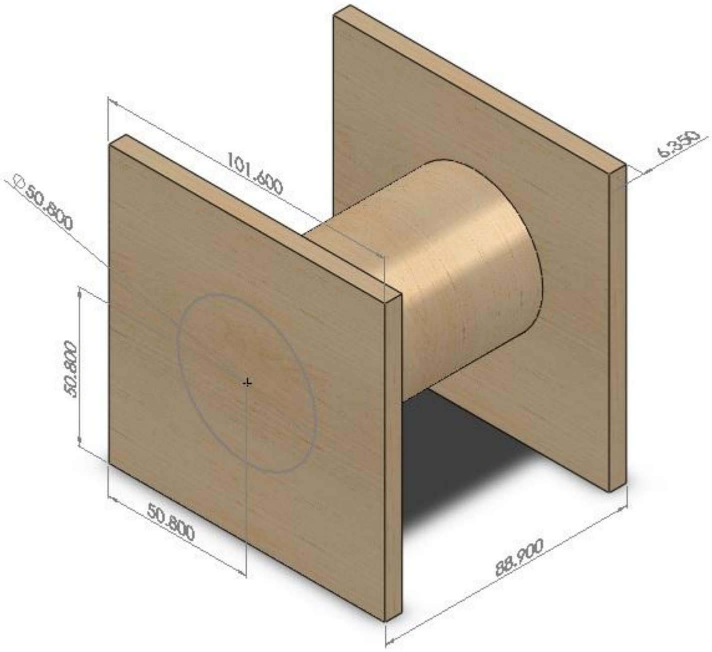


FIGURE 15: WODDEN CORE DIMENSIONS (MM)

Next the wooden core was wrapped with 400 turns of 12 gauge wire, as shown by the blue selection in the Figure below. This was than coupled with 10 turns of 8 gauge wire, which acts as our transmission line. This is shown in Figure 16. This coupling allows for a signal to be non-invasively picked up from the transmission line.

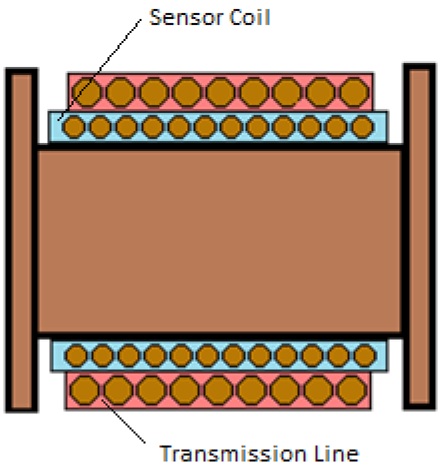


FIGURE 16: WINDING COUPLING

A picture of the actual device can be seen in Figure 17. The blue insulated wire on the right represents the transmission line, while the black insulated wire on the left represents the sensor coil.

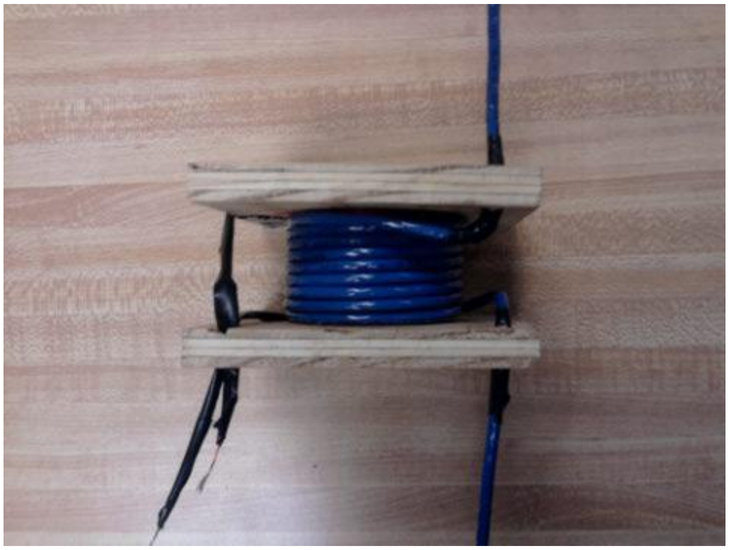


FIGURE 17: TRANSDUCER

PERFORMANCE DATA

The results that can be obtain using our transducer prior to and during a fault can be seen below in Figure 18 and Figure 19 respectively. The voltage level picked up by the transducer during normal operations is much smaller than the voltage level detected during a fault. This difference is why two different signal processing circuits must be used in the design.

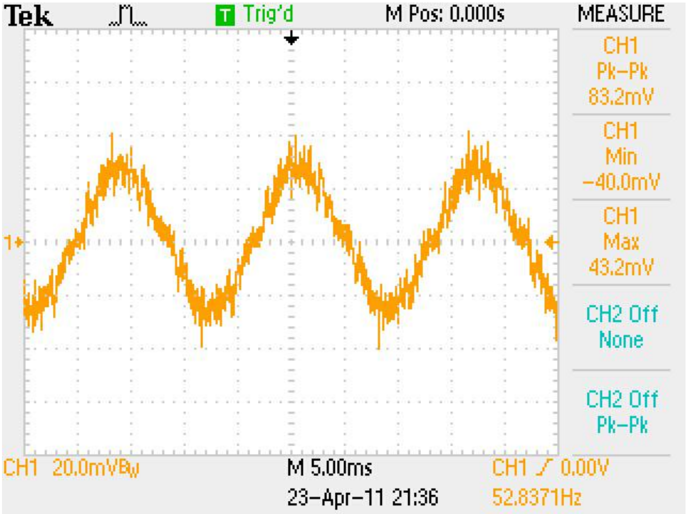


FIGURE 18 - SAMPLE TRANSDUCER OUTPUT – NORMAL OPERATION

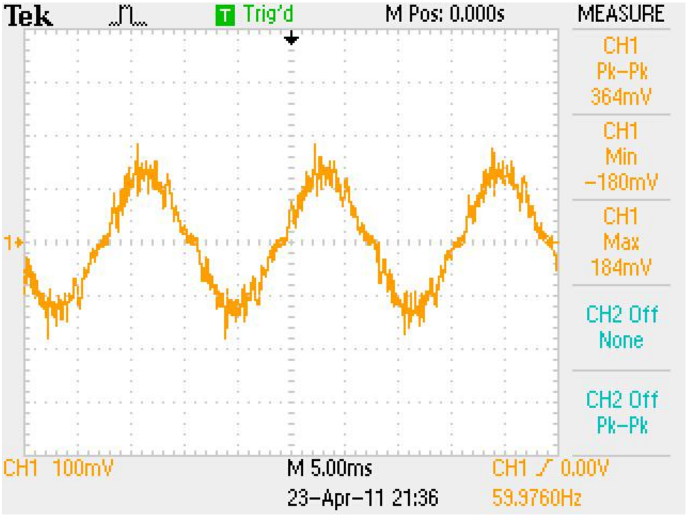


FIGURE 19 - SAMPLE TRANSDUCER OUTPUT - FAULT

4.3 TRANSDUCER SIGNAL PROCESSING (CIRCUIT 1 & 2)

The next major functional block involves processing the output of the Transducer through Circuits 1 and 2. The objective of this block is to extract the useful information from the signal and convert it into a binary output that is easily compatible with the MSP430. First, the signal is filtered to remove distortion, and then amplified to a larger magnitude to facilitate processing. Next, a precision rectifier is used to take the peak value of the sinusoid and provide a dynamic DC value. Lastly, this DC value is sent into a non-inverting Schmitt Trigger with two separate tripping values, one for “logic high” and one for “logic low.” The Schmitt Trigger then swings to either the positive or negative rail based on its input. As previously mentioned, the system has two modes of operation. Two separate signal processing circuits are used for these modes because they require different threshold voltages.

RC FILTER

The first circuit block of Circuit 1 and 2 is a low-pass RC Filter. This passive filter topology was selected over an active filter for three reasons. The first is that it does not consume any power in its operation. The second is that it is likely to cost much less than an active filter because it uses no integrated components. Lastly, this topology is very simple and easy to implement.

The first aspect of the design taken into consideration is the filter’s cutoff frequency. Since most power transmission systems function at 60 Hz, this would be the dominant signal being picked up by the transducer. Therefore to clear the noise, the filter’s cutoff frequency would need to be around 60 Hz. The cutoff frequency of a filter is recognized as f_{3dB} and described by equation (3). This is the frequency at which the input signal is attenuated by half of its power, or approximately 0.707 times the value of the peak input voltage for sinusoidal waveforms. [4]

$$f_c = \frac{1}{2\pi RC} \tag{3}$$

DESIGN CONSIDERATIONS

Capacitor values tend to come in fewer denominations than resistors. Because of this, a 1 μF capacitor was used as a constant in the design. Solving equation (3), it is found that a resistor value of 2652 Ω is needed to obtain a cutoff frequency of 60 Hz. Using this as a reference when considering standard values, a 2k Ω was selected yielding a cutoff frequency of 79 Hz.

As a final note, the use of this filter will cause a 45° lag from input to output at the cutoff frequency because of the time it takes for the capacitor's plates to charge. At 79 Hz, this results in a 1.58 ms system response delay.

FILTER SCHEMATIC

The following schematic in Figure 20 depicts the finalized design for the filter.

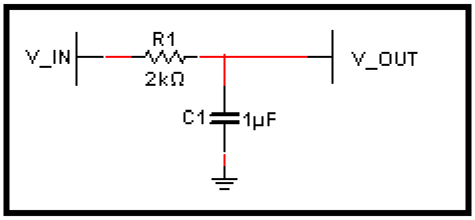


FIGURE 20 - LOW-PASS FILTER SCHEMATIC

- R1 is a 5% tolerance, 1/4 watt 2k Ω resistor.
- C1 is a 1 μF electrolytic capactor.

The error associated with using the 5% tolerance resistor is trivial such that it will have no impact on the performance of the rest of the system. The effects of the tolerance can cause the cutoff frequency to vary from 75.8 Hz to 83.8 Hz which is permissible because the important frequency of 60 Hz is not at risk of being filtered out.

PERFORMANCE DATA

The designed filter was constructed in lab and its performance was tested. First, the cutoff frequency was verified. A test sinusoid with amplitude 5V with frequency varying from 1 to 10k Hz was used. Figure 21 contains the filter's frequency response. The table of data can be found in Appendix C.

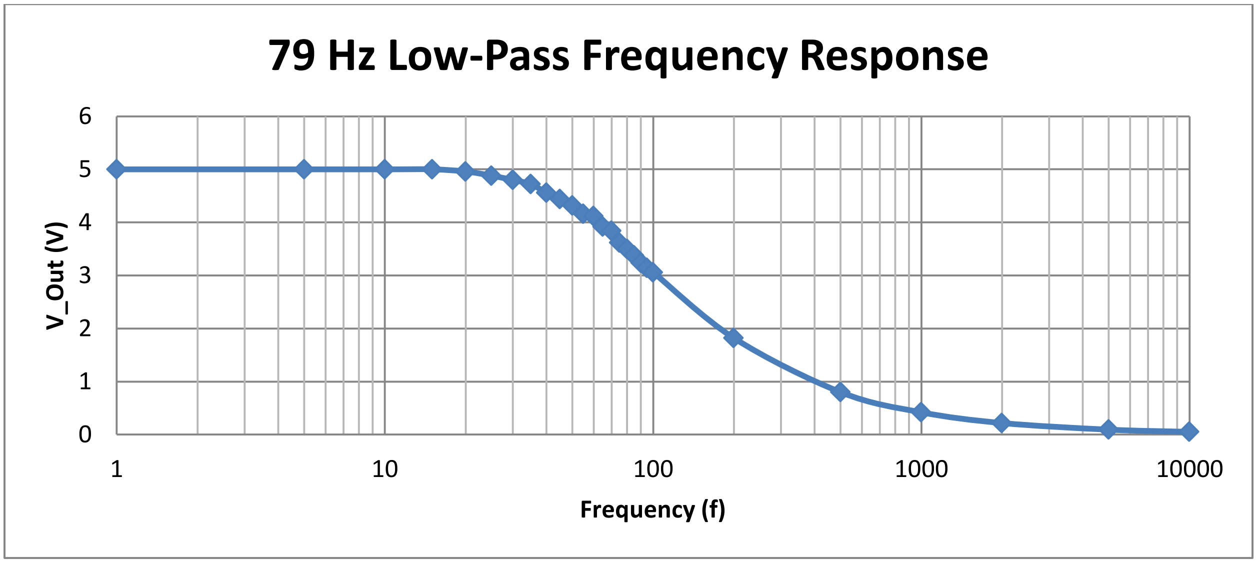


FIGURE 21- LOW-PASS FILTER FREQUENCY RESPONSE

With the frequency response verified, two filters were integrated into the system. Figure 22 is an oscillogram collected from the system from Circuit 1 under normal operation. Figure 23 verifies that a phase delay has occurred in the filtering process and provides an approximation.

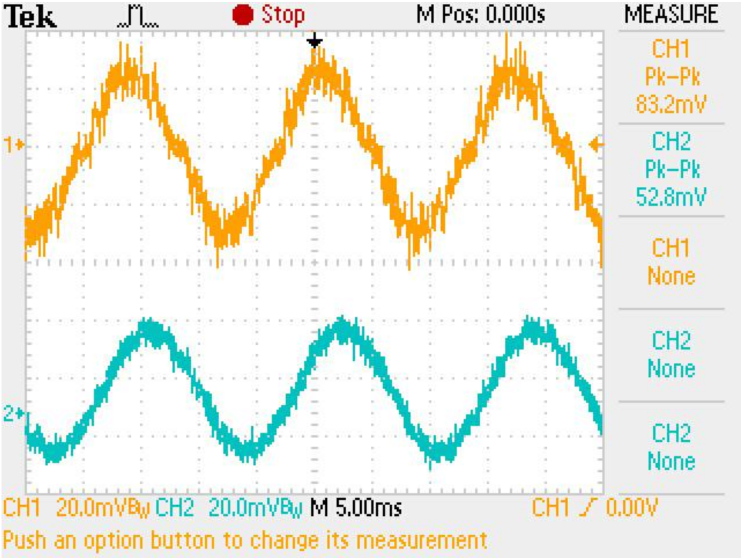


FIGURE 22 - FILTERED SIGNAL CIRCUIT 1 - NO FAULT

The waveform in yellow is the output of the transducer, while the waveform in blue is the filter output. The attenuation factor was found to be slightly greater than the theoretical value of 0.707. In this particular case it attenuated the signal down to 0.634 of its original value.

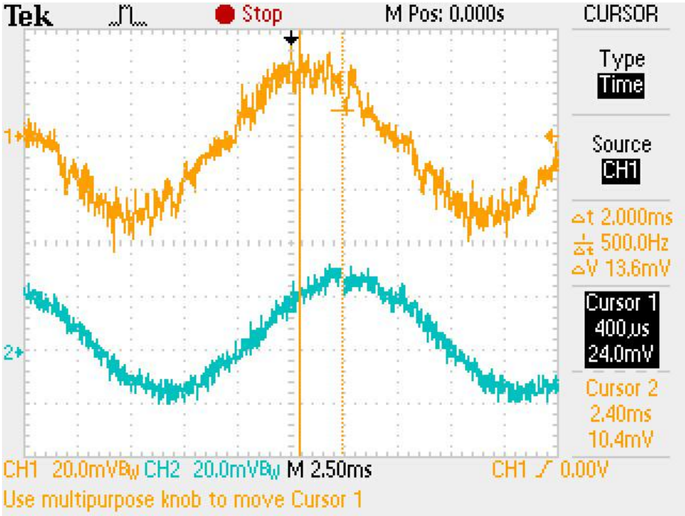


FIGURE 23 -SIGNAL PHASE DELAY

While not an exact measurement, a phase delay of approximately 2ms can be extrapolated. This relates reasonably to the theoretical delay of 1.58 ms for the filter design.

NON-INVERTING AMPLIFIER

The signal provided by the transducer was of a small magnitude and filtering it attenuated it even further. In order to process it more accurately, amplifying the signal was necessary. A non-inverting amplifier was used for this purpose. The topology was selected for its high input impedance in order to keep the input voltage stable, as well as easily adjustable gain for calibration purposes. The transfer function of this circuit is described in equation (4).

$$\mathrm{V_{Out}} = \mathrm{V_{In}}\left(1 + \frac{\mathrm{R_2}}{\mathrm{R_1}}\right) \tag{4}$$

DESIGN CONSIDERATIONS

The core of the design revolves around the selected operational amplifier. The following constraints were considered in the selection:

1. Rail voltage ranging from -15V to 15V
2. Low frequency input signal – 60 Hz
3. Small input signal minimum value of ~ 26 mV

In order to raise the input signal to a reasonable level of a few volts, the configured gains of the op-amps would need to be on the order of up to a few hundred. Considering the low frequency application, the op-amps slew rate and gain bandwidth product could be on the lower end of the performance spectrum and still meet the demands of the design.

Given this, the LM348 (Quad LM741 Chip) chipset was selected for this design. They are very widely available and low cost. The important LM348 parameters are summarized in Table 7.

LM348 Op-Amp Parameters	
Gain Bandwidth	1 MHz
Supply Min	8V
Supply Max	36V
Slew Rate	0.5V/µS

TABLE 7 – LM348 OP-AMP PARAMETERS [5]

NON-INVERTING AMPLIFIER SCHEMATIC

The following schematic in Figure 24 depicts the finalized design for the two non-inverting amplifiers.

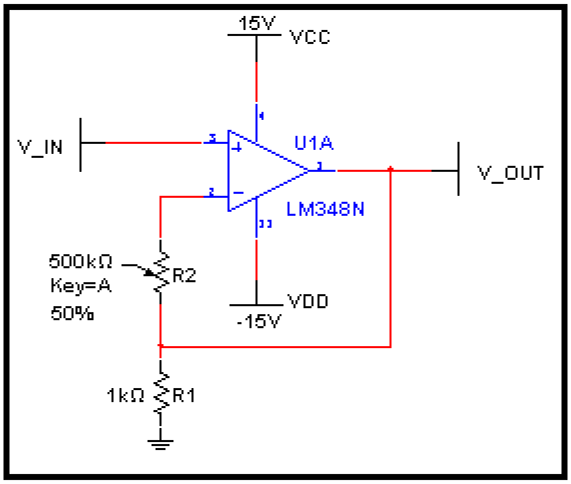


FIGURE 24 - NON-INVERTING AMP SCHEMATIC

- U1A is an LM741 Op-Amp contained within the LM348 Quad 741 Chipset.
- R1 is a 5% tolerance, 1/4 watt 1k Ω resistor.
- R2 is a potentiometer with an ideal range of 1 Ω to 500k Ω.

In the current build of the prototype, R2 is a 5% tolerance 1/4 watt 51k Ω resistor yielding a theoretical gain of 52 for Circuit 1, and a 5% tolerance 1/4 watt 300k Ω resistor yielding a theoretical gain of 301 for Circuit 2.

PERFORMANCE DATA

The design was integrated into the system and its performance checked. Figure 25 contains the performance of Circuit 1 under normal conditions. Figure 24 and Figure 25 contain the performance of Circuit 2 under normal and fault conditions respectively.

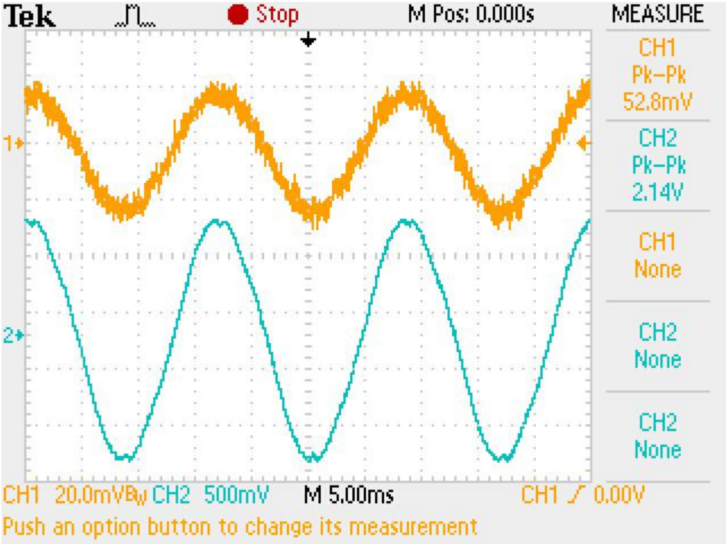


FIGURE 25 - AMPLIFIER PERFORMANCE CIRCUIT 1 NORMAL

The yellow waveform is the filter output, while the blue waveform is the amplifier output. The calculated gain of this implementation comes to 40.53. This deviates substantially from the theoretical gain of 52 for Circuit 1. This substantial decrease in expected gain is also seen in Circuit 2's amplifier. The cause of this anomalous behavior could not be determined and it is seen in Circuit 2 that the gain tends to improve with a larger input signal. Further research into this behavior is needed.

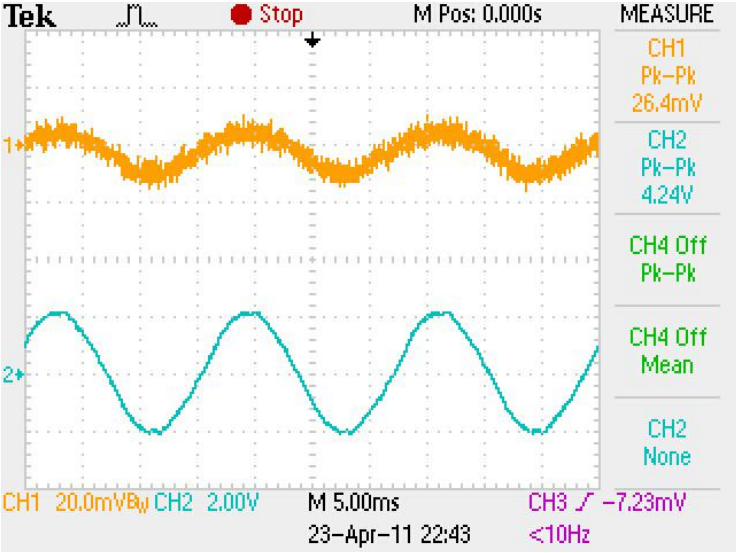


FIGURE 26 - AMPLIFIER PERFORMANCE CIRCUIT 2 - HIGH IMPEDANCE - CLEARED FAULT

The calculated gain for this implementation under the specified conditions is 160.6. This deviates by a large factor from the expected gain of 301 for Circuit 2.

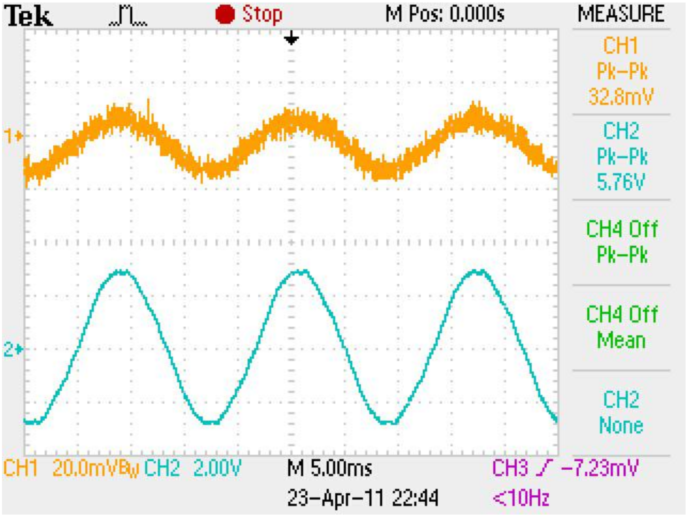


FIGURE 27 - AMPLIFIER PERFORMANCE CIRCUIT 2 – HIGH IMPEDANCE – FAULT

The calculated gain under the fault condition with a larger input signal is 175.6. This evidence may indicate that as the signal input increases, so does the gain of the circuit. The proceeding circuit blocks were designed and compensated with this anomalous behavior in mind.

PRECISION RECTIFIER

The next step in signal processing involved a method for extracting the peak value of the amplified signal. During normal operation, the peak of the input signal falls into a normal range of values centered at about 2.14V for Circuit 1, and 4.24V for Circuit 2 as seen in Figure 25 and Figure 26. However, during a fault, these values will spike because the increased current through the main line will cause an increase in induced voltage.

To capture this information a precision rectifier was used. This method of peak detection was used over more conventional peak detectors because it does not require a manual or automated reset once a peak has been detected. The precision rectifier will output a dynamic DC value based on the magnitude of the input.

DESIGN CONSIDERATIONS

Research into the topology was done and Figure 28 in the following section contains the implemented circuit for the system. [6] Considering the low frequency application of 60Hz, it was determined through data sheet inspection and testing that the LM348 Op-Amp would be sufficient to use in the design. Due to the sensitivity of the circuit, the tolerances of R2 and R3 were selected to be 1% in order to keep the output as stable and precise as possible.

Another major design consideration is the ripple voltage at the output of the rectifier. It is desired to minimize this for the final stage of processing in the Schmitt Trigger; however there are some degrees of freedom due to the Schmitt Trigger’s hysteresis. The Schmitt Trigger has two different trip thresholds that can be designed for, so a moderate ripple can be compensated for. The initial approach taken for this part was to minimize the ripple voltage to approximately 5% of the peak value. The general expression for the ripple voltage for a full-wave rectifier can be seen in (5)

$$\mathrm{V_r} = \frac{\mathrm{I}}{\mathrm{2fC}} \tag{5}$$

Taking Circuit 1’s output as an example, 5% of the peak value of 2.14V gives a target ripple of 0.107V. Because the current was unknown, testing of available capacitors was done, and a 1000 uF capacitor was found to sufficiently meet this target. The performance of this selected capacitor can be seen in Figure 30. Ultimately, the same valued capacitor was used for Circuit 2.

PRECISION RECTIFIER SCHEMATIC

The following schematic in Figure 14 depicts the finalized design selected for the two precision rectifiers.

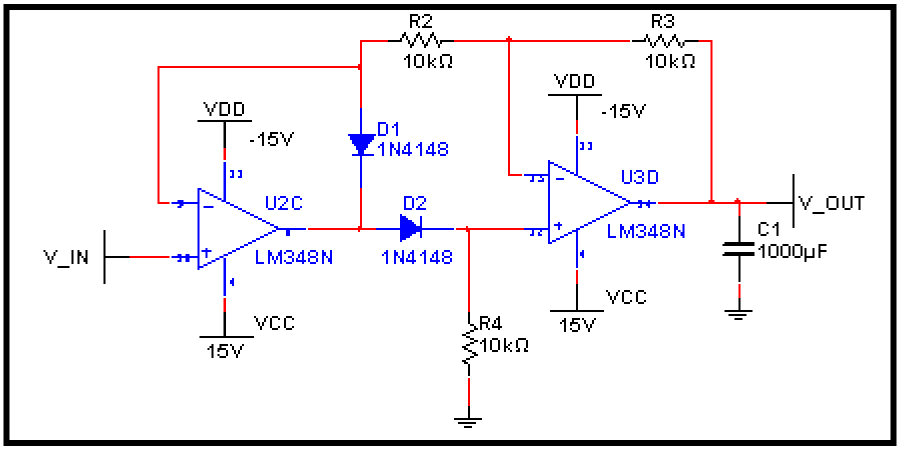


FIGURE 28 - PRECISION RECTIFIER SCHEMATIC

- U2C and U3D are part of the LM348 Quad 741 Chipset.
- R4 is a 5% tolerance, 1/4 watt 10k Ω resistors.
- R2 and R3 are 1% tolerance, 1/4 watt 10k Ω resistors.
- D1 and D2 are 1N4148 diodes.
- C1 is a 1000 uF electrolytic capacitor.

PERFORMANCE DATA

The testing of this circuit produced some slightly anomalous, yet consistent performance data. Figure 29 contains the amplifier voltage pictured in blue and the rectified voltage in purple for Circuit 1.

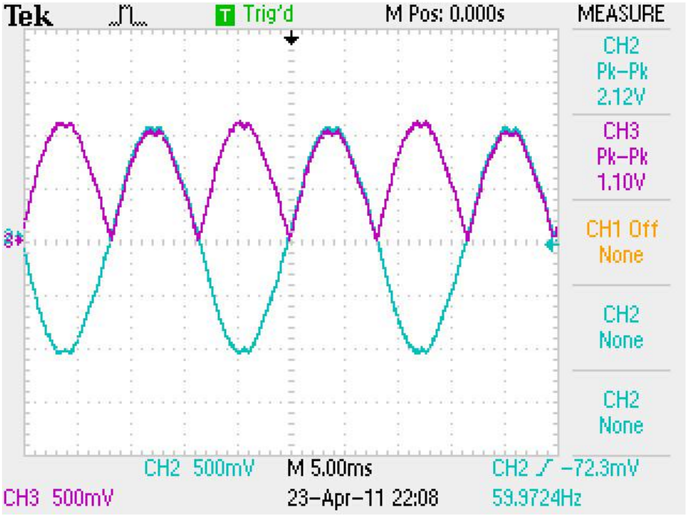


FIGURE 29 - FULL WAVE RECT - NORMAL OPERATION – CIRCUIT 1

The peak to peak value of the rectified voltage is approximately half of the original waveform, which is expected. There is some slight distortion at the peak of the waveform that could not be cleared.

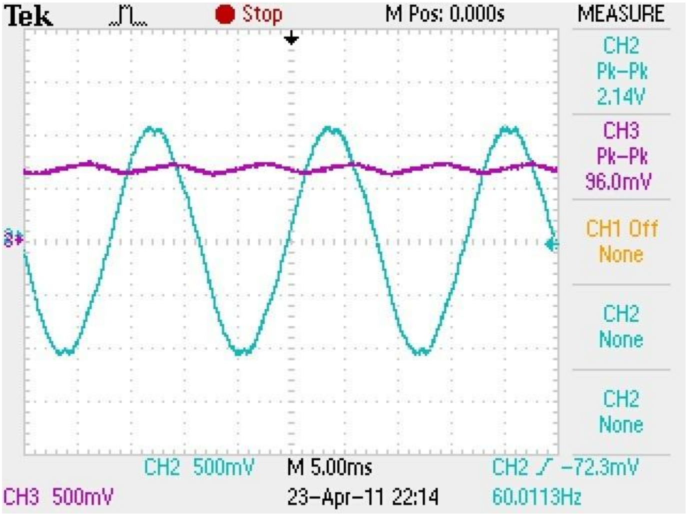


FIGURE 30 - NORMAL OPERATION - SMOOTHING CAP - CIRCUIT 1

Figure 30 contains the behavior of the circuit when the 1000 uF capacitor was added. The anomalous behavior consists of the fact that the peak value is not being taken. The reason for this could not be determined; however the behavior was consistent in all cases and was compensated for in the Schmitt Trigger design.

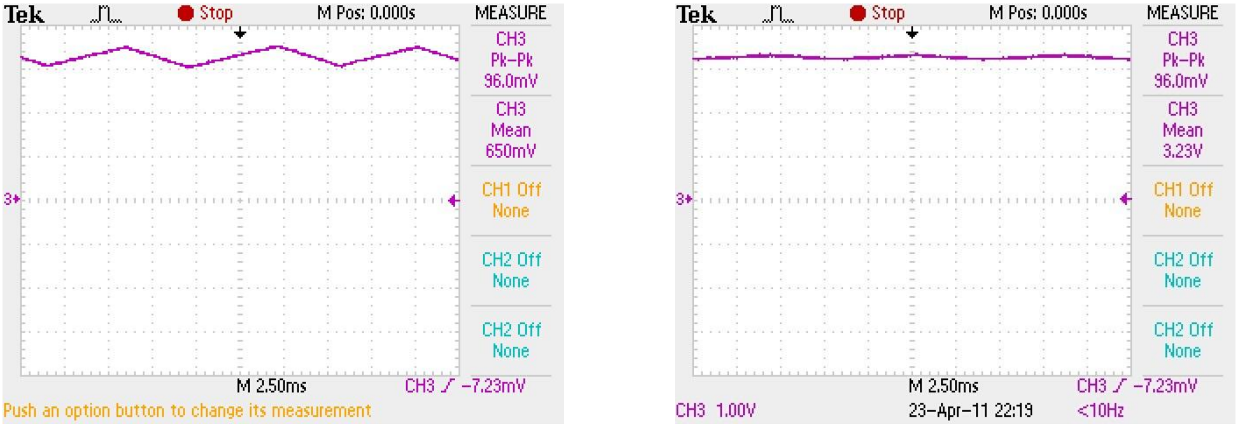

(A) (B)

FIGURE 31 - (A) CIRCUIT 1 - NO FAULT (B) CIRCUIT 1 – FAULT

The smoothed voltage was inspected more closely and its mean values extracted for the design of the Schmitt Trigger. The values of 650 mV and 3.23V, seen in Figure 31, would become the lower and upper bounds by which the Schmitt Trigger would have to swing to produce a “logic low” or “logic high.”

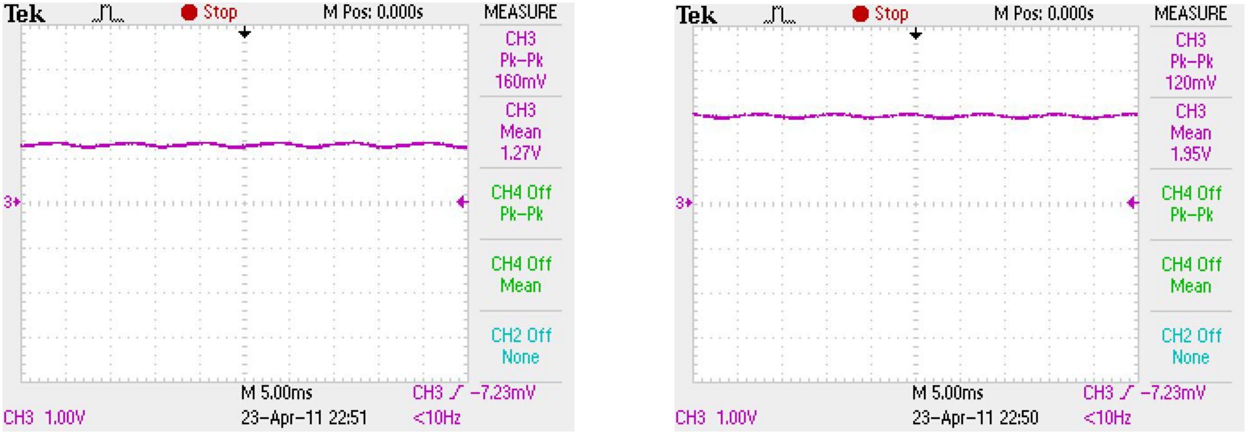

(C) (D)

FIGURE 32 - (C) CIRCUIT 2 - NO FAULT (D) CIRCUIT 2 – FAULT

The same process was done for Circuit 2. The threshold values of 1.27V and 1.95V were collected.

NON-INVERTING SCHMITT TRIGGER

The final circuit in the signal processing functional block is the Non-Inverting Schmitt Trigger. This topology was implemented for two reasons. The first is that it can produce a binary output that is easily compatible with the MSP430. The second is that through a characteristic called hysteresis, the Schmitt Trigger can be designed to have two separate threshold voltages that produce these binary outputs.

DESIGN CONSIDERATIONS

The Schmitt Trigger functions as a comparator by taking in an input at its non-inverting input and comparing it a reference voltage at the inverting input. If the input voltage exceeds the value of the reference voltage, the Schmitt Trigger's output will swing to the positive rail. Conversely, if the input is lower than the reference, then the output will swing to the negative rail. For reference, the Schmitt Trigger schematic can be seen in Figure 33 in the following section.

The characteristic of hysteresis emerges from the positive feedback configuration of R2 as it adds a fraction of the output voltage back into the input. This creates two separate threshold values centered on the reference voltage. The buffer around the reference voltage is characterized by equation (6).

$$\mathrm{V_{TL}} = \mathrm{V_{ref}} - \frac{\mathrm{R_2}}{\mathrm{R_1}};\ \mathrm{V_{TH}} = \mathrm{V_{ref}} + \frac{\mathrm{R_2}}{\mathrm{R_1}} \tag{6}$$

Where V_{TL} is the lower threshold to swing to the negative rail, and V_{TH} is the upper threshold to swing to the positive rail.

Depending on the proportion of the feedback resistor to the input resistor, the band around the reference voltage can be set. An applet was used to facilitate the design process. [7] It was able to solve for the nearest standard resistor values and reference voltage when given the rail and desired threshold voltages as inputs. Hand calculations were then done to verify the accuracy of these calculations and to make sure they met the theoretical constraints of the system. Voltage dividers also had to be designed to provide the closest reference voltages possible. These results are summarized in Table 8 in the upcoming "Performance Data" section.

NON-INVERTING SCHMITT TRIGGER SCHEMATICS

The following schematics in Figure 33 and Figure 34 depict the finalized design selected for the two Schmitt Triggers.

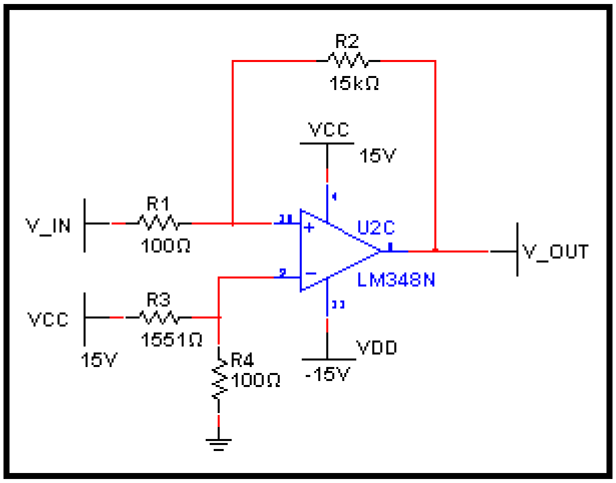


FIGURE 33 - CIRCUIT 1 SCHMITT TRIGGER SCHEMATIC

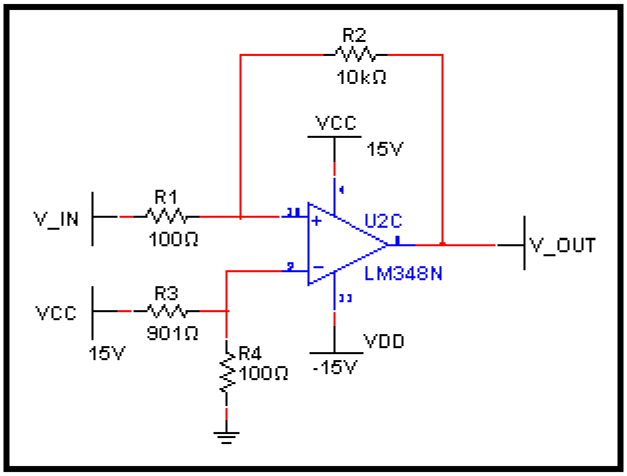


FIGURE 34 - CIRCUIT 2 SCHMITT TRIGGER SCHEMATIC

- All resistors shown are 5% tolerance, 1/4 watt.

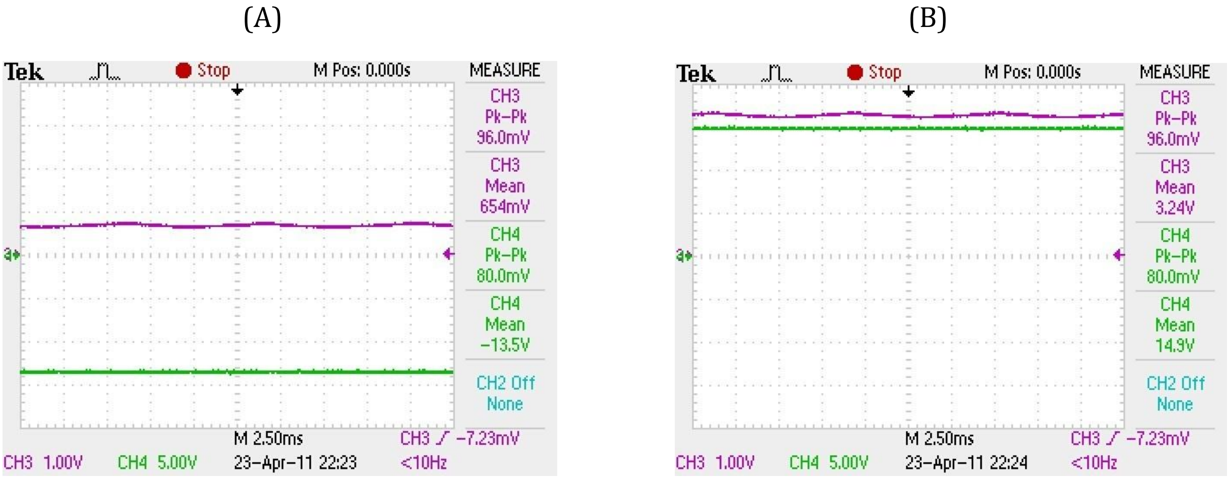
PERFORMANCE DATA

Table 8 provides a full summary of the Schmitt Trigger Data comparing the Applet results, Hand Analysis, and Measured Data. Overall actual values fall within a close range to their theoretical values.

Source	Schmitt Trigger 1			Schmitt Trigger 2		
	Vref (V)	VTL (V)	VTH (V)	Vref (V)	VTL (V)	VTH (V)
Applet (Rapid Prototyping)	0.889	0.72	1.08	1.58	1.4	1.8
Theoretical (Hand Analysis Compensated)	0.908	0.808	1.008	1.49	1.34	1.64
Actual (Prototype Measure)	0.947	0.780	1.08	1.54	1.32	1.76

TABLE 8 - SCHMITT TRIGGER DATA SUMMARY

The circuit was integrated into the system and its behavior was checked. Circuit 1's Schmitt Trigger can be seen in Figure 35and Circuit 2's Schmit Trigger in Figure 36. The waveform in purple is the output of the precision rectifier, while the green output is the Schmitt Trigger.


(A)

(B)

FIGURE 35 - CIRCUIT 1 SCHMITT TRIGGER (A) LOGIC LOW (B) LOGIC HIGH

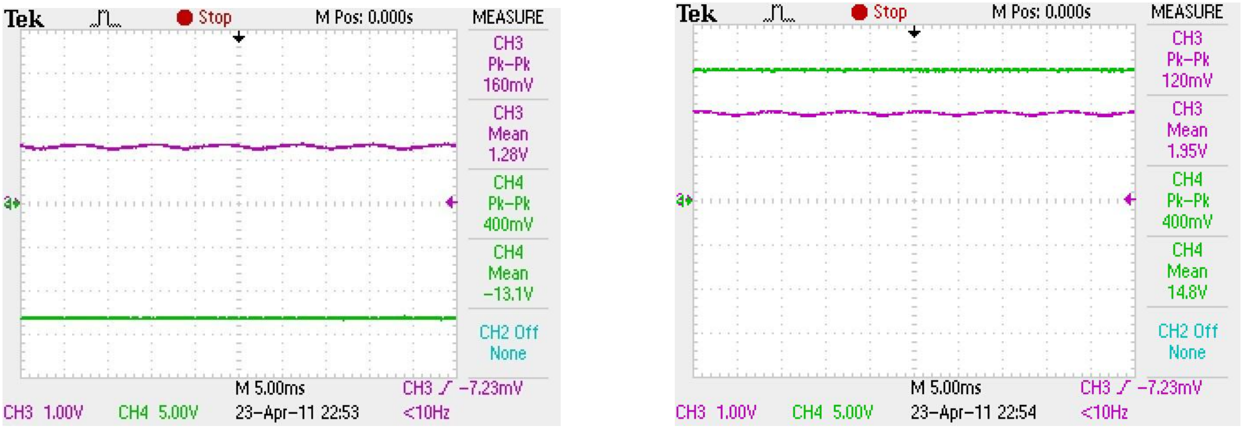

(C) (D)

FIGURE 36 - CIRCUIT 2 SCHMITT TRIGGER (C) LOGIC LOW (D) LOGIC HIGH

5.4 MSP430 CONTROL BLOCK

The final major functional block involves the control system of the Smart Recloser. This block is comprised of three main components. The first is a circuit block that scales down the binary outputs of the signal processing circuits to be compatible with the MSP430. Next is the MSP430 microcontroller itself which embodies the control system with developed code. Lastly, two relay driving circuits are implemented to allow the MSP430 to control the two relays used in the system. The first relay functions as a recloser and switches from normal to high impedance mode. The second relay switches between Circuit 1 and 2 depending on the mode of operation.

VOLTAGE SUMMER

The first circuit in this block involves multiple components and topologies cascaded together to produce a single output within the specified range of the MSP430 of 0 to 3.6V. The purpose of this block is first, to limit the -15V “logic low” of the Schmitt Triggers to 0V. Next, because only one of the signal processing circuits is active at any given time, the outputs are reduced and merged together by a Voltage Summing Circuit. This was done to simplify the code used in the MSP430 due to the limited coding background available. The transfer function of the Voltage Summer is characterized by equation (7).

$$\mathrm{V_{Out}} = -\mathrm{R_{Feedback}}\left[\frac{\mathrm{V_1}}{\mathrm{R_1}} + \frac{\mathrm{V_2}}{\mathrm{R_2}} + \cdots + \frac{\mathrm{V_n}}{\mathrm{R_n}}\right] \tag{7}$$

The advantage of this circuit is that it can weight and reduce the input voltages without having to resort to a voltage divider first. An inverting voltage summer was used because they are much more stable when compared to the non-inverting version of this topology. Lastly, because the output is inverted, an inverting amplifier of unity gain is implemented to counteract this effect.

DESIGN CONSIDERATIONS

In order to negate the -15V swing of the Schmitt Triggers, a 1N4148 diode was used. It was used due to its low cost and wide availability. Its reverse breakdown voltage of 75V greatly exceeded the application of 15V.

The next step involved the design of the voltage summer. The output voltage of the Schmitt Triggers needed to be reduced from 15V to between 2.2V and 3.6V. The low end threshold of 2.2V is the minimum fault voltage that was programmed into the MSP430, while the high end threshold of 3.6V is the maximum rated input voltage of the microcontroller. Considering standard resistor values, a fraction of 1/5.1 becomes apparent. Solving the transfer function for $\mathrm{R_{Feedback}}$ = 1 kΩ and $\mathrm{R_n}$ = 5.1 kΩ, leads to an acceptable theoretical output of -2.94V.

Lastly, because this topology inverts the input, an inverting amplifier of unity gain is implemented to revert the voltage back into the positive spectrum.

VOLTAGE SUMMER SCHEMATIC

The schematic in Figure 37 depicts the finalized design for the Voltage Summing circuit.

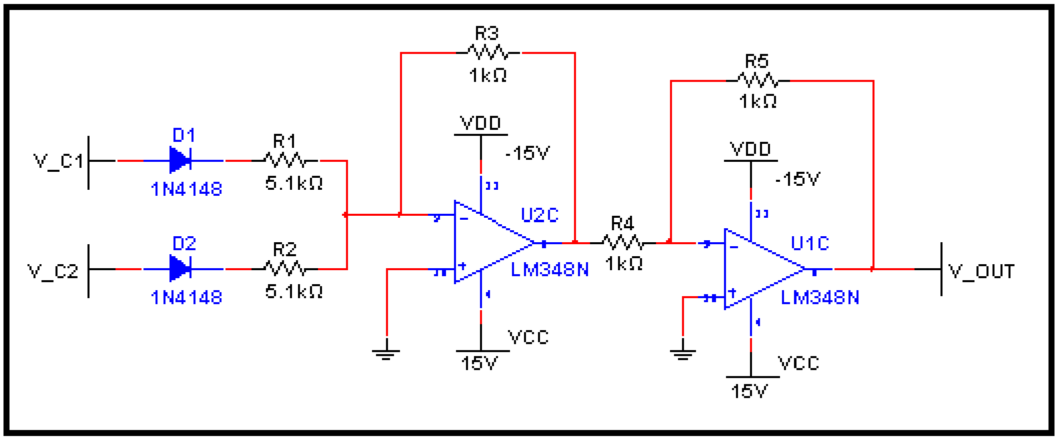


FIGURE 37 - VOLTAGE SUMMER SCHEMATIC

- D1 and D2 are 1N4148 diodes
- R1 and R2 are 1% tolerance, 1/4 watt 5.1k Ω resistors.
- R3, R4, and R5 are 1% tolerance, 1/4 watt 1k Ω resistors.
- U2C and U1C are LM741 Op-Amps contained within the LM348 Chipset.

The tolerance of all of the resistors used in this circuit is selected to be 1% because of the sensitivity that comes with using a microcontroller. It is important to make sure that the output of this circuit stays within a very limited range as to not increase the risk of damage the microcontroller.

PERFORMANCE DATA

Once the circuit block was integrated into the system, its performance data was collected. When there was no fault and Circuit 1 was engaged, the summing circuit's output was recorded to be 93.4 mV. When a fault was triggered, the circuit's output changed to 2.87V. Lastly, when the fault was cleared and Circuit 2 was still engaged, the circuit's output dropped to 88.2 mV. In comparing to the theoretical outputs of 0V for "logic low" and 2.94V for "logic high", it is found that the actual values are within an acceptable range to the ideal values. The differences can be attributed to a number of micro scale non-idealities in the components used.

MSP430 MICROCONTROLLER

The MSP430 is a microcontroller that the MQP team has worked on in previous ECE classes, and was heavily decided to be used for this application because of the previous experience. With that being said, it is also a good microcontroller for our application because of its low power consumption and can operate at 3 V. Due to the fact that our device will be operating along power distribution lines, that might fault, it must have its own independent power supply. With that being said, a possibility for normal operation could be solar power or just batteries. Ultimately, the low power consumption of the MSP430 was not the main reason we chose it, despite it fitting in with specifications related to our project. The main reason it was chosen is due to the fact that the group was proficient it its implementation, and it would be able to complete its required duties.

It also is a good choice for the Smart Recloser because we can take advantage of the analog to digital converter that allows us to constantly monitor the input to the MSP430. If the input received to the MSP430 ever exceeds the threshold set, it will take appropriate action dependent on its current state within the state machine block. This block can be shown below and describes the “logic” that the MSP430 implements into our prototype.

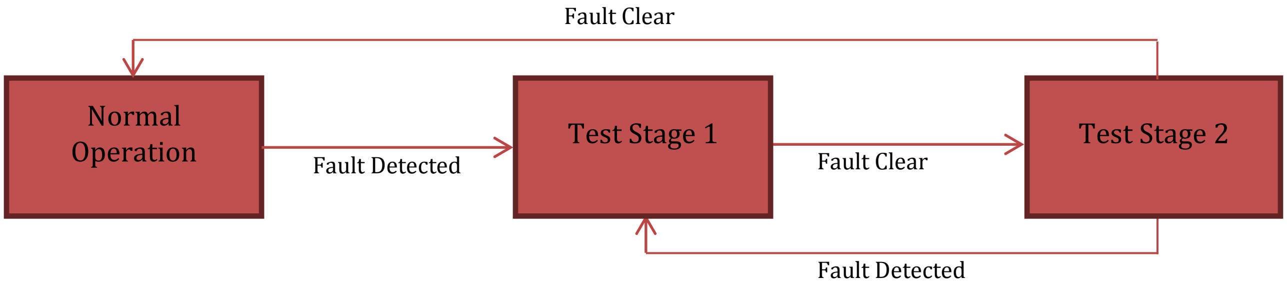


FIGURE 38: MSP430 STATE MACHINE

The logic is simple, but does ensure that two checks are completed to guarantee that the fault is clear. Another advantage to using a microcontroller is that the code can easily be changed to implement more test stages to further guarantee that the fault has cleared. The state machine above was implemented for our prototype, but dependent on the power lines fault requirements, it can be changed to reach these standards.

Shown below in the figure is an image of the MSP430 with an arrow pointing to the analog to digital converter and also to the output pins. Also shown is the figure are the precise pins are that used for our application.

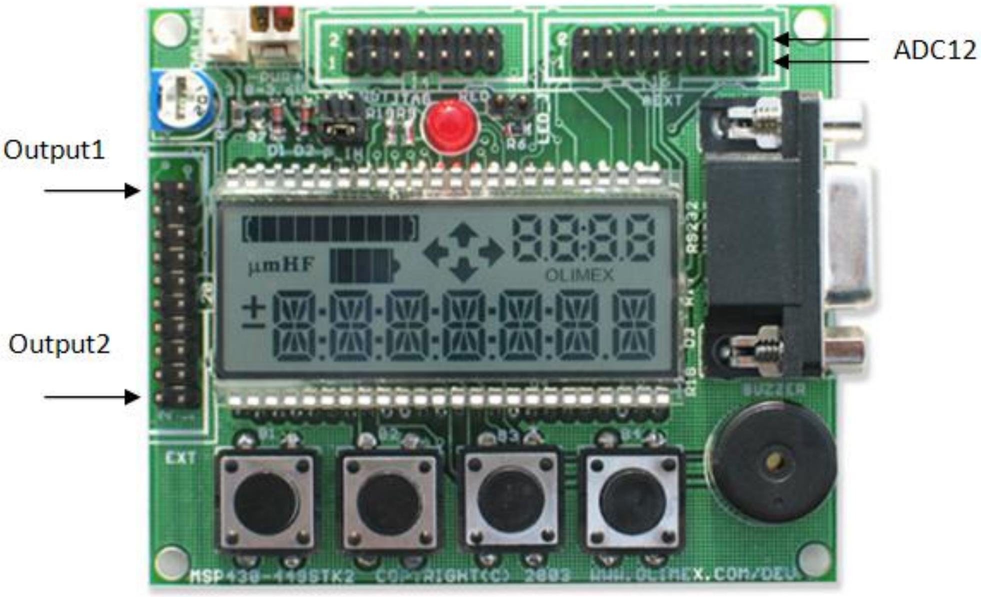


FIGURE 39 - MSP430 MICRCONTROLLER WITH PIN DESIGNATION

The first role the microcontroller is to receive the digital output of the summing circuit (either 0 or 3 V) and relay a digital signal to the two electromagnetic relays. These electromagnetic relays will be triggered such that while one is turned on, the other is off and vice versa. The MSP430 is continuously reading in a signal from the summing circuit. The electromagnetic relays (described earlier) will break the regular power lines, and switch to high impedance mode in order to decide when the fault is gone or not.

For our design there is one input to the MSP430 through the analog to digital converter (designated ADC12 in the figure), and two outputs. The MSP430 uses digital inputs (0 or 3 V) and outputs (0 or 3 V). The input for the summing circuit is monitored by the ADC12, which allows us to convert the input to a voltage that the MSP430 can use to determine if the fault has cleared or not. Output 1 controls the Zettler 1 relay and output 2 controls the Zettler 2 Relay. After a fault has occurred, the microcontroller checks the input to see if the fault has disappeared. Then our code waits for a period of time (the amount of time that the code waits can be modified to by the user), checks if the line again for a fault, and if the fault is still gone then the microcontroller will send a signal to reset the recloser. A further description of the MSP430's control for each of the operating modes is shown below.

Normal Operating Mode

- If Input 1 is logic high (this means a fault has occurred) then Output 1 will be logic high, there will then be a short delay, and then Output 2 will be logic high. This will open the transmission line, and a switch from using Circuit 1 to Circuit 2 will occur.
- If Input 1 is logic low (this means that no fault has occur, and therefore no change in the system is desired), then Output 1 and Output 2 is logic low.

High Impedance Mode

- If Input 2 is logic high (this means that a fault has occurred, and it has not been cleared yet) then Output 1 and Output 2 will remain logic high. This will keep the transmission line open, and will allow us to keep using Circuit 2.

- If Input 2 is logic low (this means that the fault has been cleared) then Output 2 will be logic low, there will be a short delay then Output 1 will change to be logic low as well. These actions will reconnect the transmission line and will switch the monitoring circuit from Circuit 2 to Circuit 1.

The code that is implemented in the MSP430 can be seen in Appendix D.

RELAY DRIVER

The MSP430's output is limited to the voltage supplied by its voltage source. In this case, it is two AA 1.5V batteries, thus limiting its voltage output to a maximum of 3V. The Zettler relays used in this design, functioning as a recloser and managing Circuit 1 & 2, require a voltage of at least 7.2V to engage the coil. A basic relay driving circuit was implemented in order to bridge the gap between these components. The circuit functions by using a transistor as a switch. By applying enough base current, the transistor is forced into its saturation region and current flows freely from collector to emitter activating the coil of the electromagnetic relay.

DESIGN CONSIDERATIONS

There are a number of factors that go into the function of this circuit. The first step is determining what voltage will be needed to power the coil. Since the VCC of 15V throughout most of the analog circuitry is much greater than the 7.2V minimum voltage for this application, a separate 9V battery was used to power the relay coils. The applied voltage across the relay coil has an effect on its impedance and current drawn. In the data sheet for the Zettler Relay, it can be found that the coil resistance with a 9V supply is approximately 155Ω +/- 10%. Through Ohm's Law, it is determined that the relay will require 58.05 mA of current to trip.

Next, the transistor must be considered. The BC337 NPN transistor was selected because it is a common selection for this application. In the data sheet, it can be found that this transistor's minimum DC current gain is 100. To function as a switch and saturate the transistor, the base current must be greater than the collector current divided by this value. As the current through the relay is the same as the current through the transistor, it is determined that a minimum base current of 580.6 uA must be applied. The MSP430's voltage output was measured to be 2.9V. Taking into consideration that it was battery powered and this value was likely to drop over extended use, a 1 kΩ resistor was used at the base to set the current to 2.9 mA, which would be more than sufficient and allow for the battery voltage to deplete as low as .58V before ceasing function.

Lastly, in order to protect the transistor from the coil's reluctance to stop current flow when it is switched off, a diode is connected in parallel with the coil to allow a path for current to flow. The 1N4001 diode was selected as it is a common choice for this application. It is rated for 50V at 30A, which is more than sufficient.

RELAY DRIVER SCHEMATIC

The schematic in Figure 40 depicts the finalized design for the Relay Driver circuit.

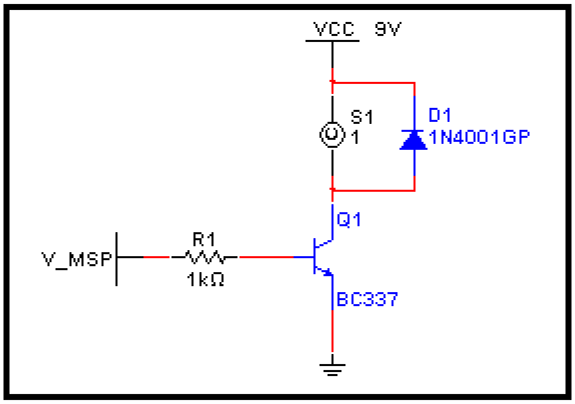


FIGURE 40 - RELAY DRIVER SCHEMATIC

- R1 is a 5% tolerance, 1/4 watt 1k Ω resistor.
- Q1 is a BC337 NPN Transistor
- D1 is a 1N4001 diode.
- S1 represents the Zettler AZ 735 coil.

PERFORMANCE DATA

The relay driver was integrated into the system and its performance was verified. It responded to the MSP430's outputs appropriately and consistent, predictable control over the relays was confirmed.

5.5 OVERALL SYSTEM PERFORMANCE

By looking at our block diagram and figuring out the system response time, we were able to determine that after a fault occurs along the power line, it takes around 10.28 ms to detect and disengage the circuit. Compared to the auto-recloser research, it took the regular system between 0.07 s and 0.16 s to detect the fault and disengage the circuit. This is most likely due to the physical separation time needed with arcing to guarantee that the circuit has disengaged. Because our prototype is of a smaller scale, it is easier to switch between normal operating mode and high impedance mode using the relatively quick Zettler Relays.

Device	Delay
Transducer	
RC Filter	2 ms
Non-Inverting Amplifier	3 μs
Precision Rectifier	10 μs
Schmitt Trigger	15 μs
MSP430 Input Circuit	15 μs
MSP430	0
Relay Driver Circuit	0
Zettler Relay 1	8 ms
Zettler Relay 2	8 ms
Total Delay	18.28 ms

FIGURE 41: PROPAGATION DELAY

When reclosing the circuit, our propagation delay is similar to detecting the fault and disengaging the circuit except that the microcontroller checks to guarantee that the fault is clear. In between its checks, it waits a designated amount of time which will increase the total delay. This is not really an issue, because the verification that the fault is gone is more important than immediately reclosing the circuit with a fault still along the line.

VI. RECOMENDATIONS

Future work for this device could take many forms. One recommendation for what future work on this device should focus on is designing the device to be placed on actual transmission lines. The prototype is limited to being tested solely in that lab due to a few restricting factors, that should be improved upon. One such factor is the power resistor that is used to limit the current on the transmission line in the case of a fault. The prototype currently uses a power resistor that can limit the current of our 100 V voltage source. However, this power resistor has not been tested at 120 V, on an actual transmission line. Having a working prototype of the device that has been tested on an actual transmission line would be a major step.

Future work should also focus on the mechanical aspects in respect to how this device will be attached to the transmission line. The prototype currently employs a transducer which detects the magnetic flux of the transmission line. Testing should be done using a transducer and an actual transmission line to determine the most effective method to position the transducer on the transmission line so that the magnetic flux can be properly detected so that it may be utilized with the Smart Recloser. After this has been determined and the device has been calibrated to be used in conjunction with the magnetic flux that the transducer can detect, work must be completed to discover an effective means to attach the device to the transmission line. Also, since the device will most likely be subjected to the natural elements there must be work done to ensure that the device will be able to be functional in cases such as extreme temperature fluctuations, as well as snow and rain.

The prototype also has a power source that could be improved upon. It currently employs common 9 V batteries to power the transistor network, as well as the circuitry, and the microcontroller. Future work could be done to determine if a smaller battery with a longer lifetime expectancy, such as a lithium-ion type battery, would be more adequate for this device. A smaller battery source would allow the device to be lighter and therefore more easily attached to a transmission line. The lifetime of the battery would be essential because it is not feasible to have this device on a transmission line and expect someone to have to change the batteries every few weeks. A further advancement on the powering of this device would be to have a self sustaining power source. One such type of means that may be feasible is the employment of solar, or wind power to power the device.

Lastly, although the calibration of the device should not often have to be performed and is fairly simple, future work is recommended to make this work simpler. The calibration of this device may have to occur if the input voltage generated by the transducer is for some reason altered. It involves the switching of a single resistor of the Schmitt-Trigger to account for this possible change to the input voltage that is fed to the circuitry. The resistor that is being altered controls the trip voltage levels of the Schmitt-Trigger, and can account for any such change of the input voltage. It is recommended that future work look into using a potentiometer instead of a typical resistor for the resistor that may have to be altered in the Schmitt-Trigger. This would allow for quick calibration by the turning of a knob, and would be advantageous for the device because it would make the device more adaptable, and allow it to be used in a wider range of applications.

VII. CONCLUSION

The MQP team was able to successfully build and test a fully functional Smart Recloser prototype that is comparable to the current auto recloser devices used in overhead power distribution lines. Although the prototype could not be tested along actual power lines, the prototype was able to prove several important points that include the plausibility of a “smart” recloser and the importance of control within a system.

The Smart Recloser is able to detect a fault along power lines as well as determine if the fault is no longer present along the line. This is a huge improvement over the current reclosers that are unable to determine if a fault still remains along the line or not. As previously stated, the ability to determine if a fault still exists is extremely important to ensure that the quality of electricity delivered to its subscribers is stable. This quality control is due to the implementation of a microcontroller in the current system, along with a high impedance mode that allows for non-invasive monitoring during a fault. The control that was displayed by the MSP430, not only validates the importance of a microcontroller in the current recloser system, but also in any system that requires any control. The microcontroller was able to monitor the fault, detect if the fault was clear, and also keep track of the amount of checks completed before validating that the fault is completely clear along the line. Besides continually monitoring the fault, the microcontroller was able to take appropriate action depending on its state and switch between circuits and normal operating/high impedance mode.

VIII.WORKS CITED

[1] Naval Facilities Engineering Command. (1990, April) Whole Building Design Guide. [Online]. http://www.wbdg.org/ccb/NAVFAC/OPER/mo201.pdf

[2] (2007, August) Engineering-Tips Forums. [Online]. http://www.eng-tips.com/viewthread.cfm?qid=195054&page=252

[3] Federacion De Colegios De Ingenieros Mecanicos. Network Protection and Automation Guide. [Online]. http://www.fecime.org/referencias/npag/chap14-20-218-231.pdf

[4] Wayne Storr. (2011, April) Passive Low Pass Filter. [Online]. http://www.electronics-tutorials.ws/filter/filter_2.html

[5] Texas Instruments. (2002, December) LM348 Quad General-Purpose Operational Amplifier. [Online]. http://focus.ti.com/lit/ds/symlink/lm348.pdf

[6] Elliot Rod. (2005) Precision Rectifiers. [Online]. http://sound.westhost.com/appnotes/an001.htm

[7] Gareth Harris. (2006, April) Non-Inverting Schmitt Trigger Calculations. [Online]. http://www.random-science-tools.com/electronics/schmitt-trigger-calculator.htm

THIS PAGE IS INTENTIONALLY LEFT BLANK

IX. APPENDIX

SECTION A – PSPICE SIMULATION CODE

Autorecloser Behavior

```
V 1 0 sin(0 100 60)
RS 1 2 40              ;; Source Resistance
LS 2 3 2.7m IC = 0     ;; Source Inductance – 5 Mile Line @ 2.27 nH/inch
;; http://www.ece.uci.edu/docs/hspice/hspice_2001_2-269.html

SW 3 4 10 0 SWCH1
RFE 4 5 1m
RL 5 0 200
SF 5 6 11 0 SWCH2
RF 6 0 10

.Model SWCH1 VSwitch(RON=1m ROFF=10MEG VON=1 VOFF=0)
VP1 10 0 Pulse(0 5 0 1u 1u 66.68m 133.38m)
.Model SWCH2 VSwitch(RON=1m ROFF=10MEG VON=1 VOFF=0)
VP2 11 0 Pulse(0 5 33.34m 1u 1u 66.68m 100.04m)

.Probe
.TRAN 300m 300m 0 300u UIC
.END
```

Smart Recloser Behavior

```
Note: Unsuccessful in characterizing the voltage dependant source of
SWCH1, so resorted to using a PWL to simulate behavior. However it
should be noted that the voltages aren’t accurate, just the behavior.

V 1 0 sin(0 100 60)
RS 1 2 40              ;; Source Resistance
LS 2 3 2.7m IC = 0     ;; Source Inductance – 5 Mile Line @ 2.27 nH/inch

SW 3 4 10 0 SWCH1
RH 3 4 1000            ;; For High Impedance Mode
RFE 4 5 1m
RL 5 0 200
SF 5 6 11 0 SWCH2
RF 6 0 10

.Model SWCH1 VSwitch(RON=1m ROFF=10MEG VON=1 VOFF=0)
VS1 10 0 PWL(0 5 66.68m 5 67m .3 75m .1 134.34m .1 138.34m 1.5 143m 5)
.Model SWCH2 VSwitch(RON=1m ROFF=10MEG VON=1 VOFF=0)
VP2 11 0 Pulse(0 5 33.34m 1u 1u 100m 300m)

.Probe
```

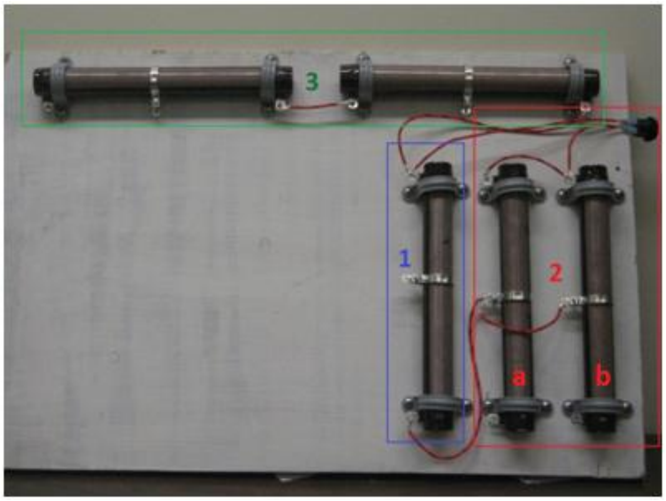
```
.TRAN 300m 300m 0 300u UIC
.END
```

SECTION B – TESTING APPARATUS PRODUCT IMAGES

Wall Outlet Transformer

Zettler Power Relay

Power Resistor Array



Fault Switch

SECTION C – LOW-PASS FILTER RESPONSE DATA

Freq (f)	V_O (V)
1	5
5	5
10	5
15	5
20	4.96
25	4.88
30	4.8
35	4.72
40	4.56
45	4.44
50	4.32
55	4.16
60	4.12
65	3.92
70	3.84
75	3.62
80	3.5
85	3.38
90	3.24
95	3.14
100	3.06
200	1.82
500	0.8
1000	0.42
2000	0.218
5000	0.0936
10000	0.052

SECTION D – MSP430 CODE

```
#include "msp430x44x.h"             // Definitions, constants, etc for
msp430F449
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <in430.h>
// ********** FUNCTION
DECLARATIONS**********************************************
void init_sys(void);                            // MSP430 Initialization
routine
void setuplogiclow1(void);                //gives output of  0V
void setuplogichigh1(void);               //gives output of  3V
void setuplogiclow2(void);                //gives output of  0V
void setuplogichigh2(void);               //gives output of  3V
//void setupKeypad(void);                  // Keypad initialization
void getKeys(void);                       // Read from ketpad
void setupButtons(void);                  // Button initialization
void getButtons(void);                    //search for button being pushed
void swDelay(unsigned int max_cnt);              // simple SW delay loop
void clearLCD(void);                      // Clears LCD memory segments so
that LCD is blank
void initLCD(void);                       // Setup code to interface LCD with
MSP430F449
void writeLetter(int position,char letter);     // display single character
on LCD
void buzzerOn(void);                      // turns buzzer on
void buzzerOff(void);                     // turns buzzer off
void LEDOff(void);                  // sets up green LED port and turns LEDs
off
void make_value(unsigned long int dec);
void checkvoltage(void);
void writeWord(const char *word);

//void setupKeypad(void);   // Keypad initialization

void LEDdisplayHex(unsigned char num);  // displays hex code for digit num

void runtimerb(void);
void stoptimerb(int reset);
void checktime(void);
__interrupt void Timer_B0(void);

//void display_value(unsigned long int dec);
int int_to_ASC(unsigned long int price, char chars[]);
/*************************************************************************/
/*
                              */
/* Global variable declarations
                        */
/*
                              */
```

```
/**********************************************************************/
char*LCD = LCDMEM;           // pointer to LCD Memory Segments
unsigned char hitKey=0;
unsigned char hitButton=0;
char display[];
int i;
unsigned int input, busy;
double voltage=0;

/**********************************************************************/
/*
                              */
/* main() variable declarations
                   */
/*
                              */
/**********************************************************************/
/***************** LCD CONSTANTS
************************************************/
#define     a      (0x80)    // definitions for LCD seegments on the Olimex
LCD. 4-Mux operation is assumed
#define     b      (0x40)    // For more details on 4-Mux operation, gather
your LCD datasheet,
#define     c      (0x20)    // TI's MSP430F449 User Guide (look for LCD
Controller, then 4-Mux),
#define     d      (0x01)    // and MSP-449STK-2 schematic. You will need ALL
these 3 when defining
#define     e      (0x02)    // each number or character. Remember, the
Olimex LCD doesn't use a LCD driver!
#define     f      (0x08)    // You tell the LCD what characters to display.
It's very time consuming!!
#define     g      (0x04)
#define     h      (0x10)
#define     LEN     8        // The length

#define         V_BIT   0.00061

#define     LEN     8        // The length
#define         V_BIT   0.00061           // used to convert the input to a
voltage reading
/******************** MAIN FUNCTION ********************/
void main(void){


  WDTCTL = WDTPW + WDTHOLD;   // Stop watchdog timer
  init_sys();                 // Initialize the MSP430
  setupButtons();             // Setup buttons
  setuplogiclow1();
  setuplogiclow2();
  writeWord("MQP");
  int i=0;
      while(1){

            checkvoltage();
```

```
            make_value(voltage);
                //writeWord(voltage);

                  if((voltage > 2000)||(i==1)){                   //a 2
voltage inputs signifies a fault
                        writeWord("FAULT");

                                                      //Engage Circuit 1
                        setuplogichigh1();                  //Tell the
recloser to open and switch to high impedence mode
                                swDelay(5);                       //Wait
approximately 2 sec
                        setuplogichigh2();                  //Switch to
Circuit 2
                        i=1;

                        if(voltage < 2000){                 //This means that
the fault has cleared

                              swDelay(15);                        //Wait
approximately 2 sec, if you want to wait longer that increase the swdelay
time
                              checkvoltage();                     //see if
the fault is still clear, and ensure it is not a persistent fault
                        make_value(voltage);                //Format it for
the MSP 430 to read

                                    if((voltage < 2000)&& (i==1)){  //see if
the fault is still clear, and ensure it is not a persistent fault
                                    swDelay(5);

                              checkvoltage();

                              make_value(voltage);

                                    if((voltage < 2000)&& (i==1)){      //see
if the fault is still clear, and ensure it is not a persistent fault
                                          clearLCD();
                                          writeWord("LOW");
                                    setuplogiclow1(); //Tell the recloser to
close, and switch out of high impedence mode
                                          swDelay(5);
                                    setuplogiclow2(); //Switch back to
circuit 1
                                          i=0;
                                                            }
                                                }
                                     }
                              }

                  if((voltage < 2000) && (i==0)){                 //No fault
has occurred just keep monitoring the line

                            clearLCD();
```

```
                        writeWord("LOW");
                      setuplogiclow1();
                            swDelay(5);
                      setuplogiclow2();

                                      }
                            }
                         }

void checkvoltage(void){
    ADC12CTL0 = SHT0_6 + SHT1_6 + REF2_5V + REFON + ADC12ON;      //set up
ADC12
    ADC12CTL1 = SHP;
      //set up ADC12
    ADC12MCTL0 = SREF_1 + INCH_0;
      //set up ADC12
    P6SEL |= BIT0;
      //set up ADC12
      for(i = 0; i <10; i++){
            ADC12CTL0 |= ADC12SC + ENC;                                 //set
up ADC12
            busy = 1;
            while (busy != 0){
                  busy = ADC12CTL1 & ADC12BUSY;
            }
            input = ADC12MEM0 & 0x0FFF;
            voltage = input*V_BIT*1000;
      //turn input onto a voltage
      //store in array []
      }
}

/******************* setuplogic high and low() ************************/

void setuplogichigh1(void)
{
  P3DIR |= BIT3;       // P3.3 Column Select output
  P3SEL &= ~(BIT3);    // P3.3 I/O Function
  P3OUT |= BIT3;       // P3.3 High
}
void setuplogiclow1(void)
{
  P3DIR |= BIT3;       // P3.3 Column Select output
  P3SEL &= ~(BIT3);    // P3.3 I/O Function
  P3OUT &= ~BIT3;      // P3.3 Low
}
void setuplogichigh2(void)
{
  P1DIR |= BIT1;       // P1.1 Column Select output
  P1SEL &= ~(BIT1);    // P1.1 I/O Function
  P1OUT |= BIT1;       // P1.1 High
}
void setuplogiclow2(void)
{
```

```
  P1DIR |= BIT1;       // P1.1 Column Select output
  P1SEL &= ~(BIT1);    // P1.1 I/O Function
  P1OUT &= ~BIT1;      // P1.1 Low
}

void setupKeypad(void)
{
  P1DIR |= BIT7|BIT6|BIT5;       // P1.7-5 Column Select output
  P1SEL &= ~(BIT7|BIT6|BIT5);    // P1.7-5 I/O Function
  P1OUT |= BIT7|BIT6|BIT5;       // P1.7-5 High

  P2DIR &= ~(BIT3|BIT2|BIT1|BIT0);    // P2.3-0 Row Select input
  P2SEL &= ~(BIT3|BIT2|BIT1|BIT0);    // P2.3-0 I/O Function
}
void setupButtons(void)
{
  P3SEL &= ~(BIT7|BIT6|BIT5|BIT4);  // 0000 xxxx
  P3DIR &= ~(BIT7|BIT6|BIT5|BIT4);  // 0000 xxxx
}
/****************** swDelay() ***********************/
void swDelay(unsigned int max_cnt)
{
   unsigned int  cnt1=0, cnt2;

   while (cnt1 < max_cnt)
   {
     cnt2 = 0;
     while (cnt2 < 65535)
       cnt2++;
     cnt1++;
   }
}
void make_value(unsigned long int dec)
{
  for(int m = 6; m > 0; m--){
    if(m > 3){
      display[m] = (char)((dec % 10) + 0x30);
    } else if(m <= 3){
      display[m-1] = (char)((dec % 10) + 0x30);
    }
    dec = (float)(dec * 0.1);
  }
  display[3] = (char)(0x2E);      //Period in ASCII
  display[7] = (char)(0x00);      //Null in ASCII
  //if (display[2]=='1')
    //writeWord (display);          //used for testing purposes
}


/****************** getKeys() ***********************/
void getKeys(void)
{
  unsigned char lookup[]={'1','4','7','+','2','5','8','0','3','6','9','#'};
  unsigned char KY[12];
```

```
  unsigned char RW=0x01;
  unsigned char CL=0x20;
  for(int i=0; i<12; i++)
  {
    P1OUT &= ~CL;                  // Set current Column
    KY[i] = (P2IN&0x0F)==(~RW&0x0F);
    if(KY[i]==1)                   // Key was pressed
    {
      hitKey = lookup[i];          // Gets key value
      break;
    }
    else
      hitKey=0;                    // no key was pressed
    RW <<=1;                       // Goto next row
    if(RW>0x08)
    {
      P1OUT |= BIT7|BIT6|BIT5;     // P1.7-5 High
      RW=0x01;                     // Goto 1rst Row
      CL <<=1;                     // Goto next column
    }
  }
  P1OUT |= BIT7|BIT6|BIT5;    // P1.7-5 High
}

/****************** getButtons()*********************/
void getButtons(void)
{
  unsigned char i = 0;
  unsigned char input = (~(P3IN >> 4)) & 0x0F; //Get high nibble

  switch (input){
    case 0:
      i = '0';
      break;
    case 1:
      i = '1';
      break;
    case 2:
      i = '2';
      break;
    case 4:
      i = '3';
      break;
    case 8:
      i = '4';
      break;
    default:
      break;
  }

  hitButton = i;

}
```

```
/******************** initSys() *****************************/
void init_sys(void)
{
  initLCD();                  // Setup LCD for work
  clearLCD();                 // Clear LCD display
  //setupKeypad();              // Setup Keypad ports
  LEDOff();
}



// **************************** initLCD
***************************************
void initLCD(void)   // initialize the various registers for LCD to work
{                    // (code obtained from sample demos of MSP430F449)
    FLL_CTL0 = XCAP10PF;          //set load capacitance for 32k xtal
    // Initialize LCD driver (4Mux mode)
    LCDCTL = LCDSG0_7 + LCD4MUX + LCDON; // 4mux LCD, segs16-23 = outputs
    BTCTL  = BT_fLCD_DIV128;             // set LCD frame freq = ACLK
    P5SEL  = 0xFC;                       // set Rxx and COM pins for LCD
}


// **************************** clearLCD
***************************************
void clearLCD(void) // makes the LCD blank
{                   // clear LCD memory to clear display
    unsigned int iLCD;
    for (iLCD =0; iLCD<20; iLCD++)  // clears all 20 LCD memory segments
    {
      LCD[iLCD] = 0;
    }
}

int int_to_ASC(unsigned long int price, char chars[])
{
  char ascii[LEN];        // LEN is the length of float.

  for(int i=6;i>0;i--)
  {
    if(i>4)
    {
      ascii[i]= (char)((price%10) + 0x30);
    }
    else if(i<=4)
    {
      ascii[i-1]= (char)((price%10) + 0x30);
    }
    price = (float)(price * 0.1);
  }
  ascii[4]=(char)(0x2E);  // Means '.' in ASCII
  ascii[7]=(char)(0x00);  // Means 'NULL' in ASCII
```

```
  // Compare Arrays
  for(int n=0;n<LEN;n++)
  {
    if(ascii[n]!=chars[n])
    {
      return 1;
    }

  }
return 0;
}
// **************************  writeLetter
**********************************
void writeLetter(int position,char letter) // writes single character on the
LCD.
{                                          // User can specify position as
well
    // DO NOT PLAY WITH THE CODE BELOW ---------------------------------------
---
    if (position == 1)  // this is position adjustment for compatibility
      position = position + 6;
    else
      if ( (position > 1) &  (position < 8) )
        position = ((position * 2) - 1) + 6;  // adjust position
    // -----------------------------------------------------------------------
---

    switch(letter)
    {
       // letter  // LCDM7                          // LCDM8
// End
       case 'A':  LCD[position-1] = a + b + c + e;      LCD[position] = b + c
+ g;        break;
       case 'B':  LCD[position-1] = c + h + e;          LCD[position] = b + c
+ g;        break;
       case 'C':  LCD[position-1] = a + h;              LCD[position] = b +
c;            break;
       case 'D':  LCD[position-1] = b + c + h + e;      LCD[position] = c +
g;            break;
       case 'E':  LCD[position-1] = a + h + e;          LCD[position] = b + c
+ g;        break;
       case 'F':  LCD[position-1] = a;                  LCD[position] = b + c
+ g;        break;
       case 'G':  LCD[position-1] = a + c + h + e;      LCD[position] = b +
c;            break;
       case 'H':  LCD[position-1] = b + c + e;          LCD[position] = b + c
+ g;        break;
       case 'I':  LCD[position-1] = a + h + f;          LCD[position] = d;
break;
       case 'J':  LCD[position-1] = b + h + c;          LCD[position] = c;
break;
       case 'K':  LCD[position-1] = d + g;              LCD[position] = b + c
+ g;        break;
```

```
       case 'L':  LCD[position-1] = h;                  LCD[position] = b +
c ;           break;
       case 'M':  LCD[position-1] = b + c + g;          LCD[position] = b + c
+ f;        break;
       case 'N':  LCD[position-1] = b + c + d;          LCD[position] = b + c
+ f;        break;
       case 'O':  LCD[position-1] = a + b + c + h;      LCD[position] = b +
c;            break;
       case 'P':  LCD[position-1] = a + b + e;          LCD[position] = b + c
+ g;        break;
       case 'Q':  LCD[position-1] = a + b + c + h + d;  LCD[position] = b +
c;            break;
       case 'R':  LCD[position-1] = a + b + d + e;      LCD[position] = b + c
+ g;        break;
       case 'S':  LCD[position-1] = a + c + h + e;      LCD[position] = b +
g;            break;
       case 'T':  LCD[position-1] = a + f + b;          LCD[position] = d +
b;            break;
       case 'U':  LCD[position-1] = b + c + h;          LCD[position] = b +
c;            break;
       case 'V':  LCD[position-1] = g;                  LCD[position] = b + c
+ e;        break;
       case 'W':  LCD[position-1] = b + c + d;          LCD[position] = b + c
+ e;        break;
       case 'X':  LCD[position-1] = d + g;              LCD[position] = e +
f;            break;
       case 'Y':  LCD[position-1] = b + c + h + e;      LCD[position] = f;
break;
       case 'Z':  LCD[position-1] = a + h + g;          LCD[position] = e;
break;

       // number  // LCDM7                              // LCDM8
// END
       case '0':  LCD[position-1] = a + b + c + h;      LCD[position] = b +
c;            break;
       case '1':  LCD[position-1] = b + c;              LCD[position] = d &
a;            break;
       case '2':  LCD[position-1] = a + b + e + h;      LCD[position] = c +
g;            break;
       case '3':  LCD[position-1] = a + b + c + e + h;  LCD[position] = g;
break;
       case '4':  LCD[position-1] = b + c + e;          LCD[position] = b +
g;            break;
       case '5':  LCD[position-1] = a + c + h + e;      LCD[position] = b +
g;            break;
       case '6':  LCD[position-1] = a + c + h + e;      LCD[position] = b + c
+ g;        break;
       case '7':  LCD[position-1] = a + b + c;          LCD[position] = d &
a;            break;
       case '8':  LCD[position-1] = a + b + c + e + h;  LCD[position] = b + c
+ g;        break;
       case '9':  LCD[position-1] = a + b + c + e ;     LCD[position] = b +
g;            break;
```

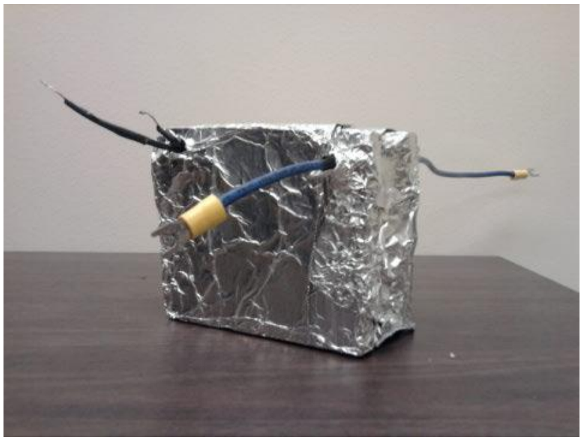
```
       // others
       case '.':                                         LCD[position] = h;
break;  // decimal point
       case '^':                                         LCDM2 = c;
break;  // top arrow
       case '!':                                         LCDM2 = a;
break;  // bottom arrow
       case '>':                                         LCDM2 = b;
break;  // right arrow
       case '<':                                         LCDM2 = h;
break;  // left arrow
       case '+':                                         LCDM20= a;
break;  // plus sign
       case '-':                                         LCDM20= h;
break;  // minus sign
       case '&':                                         LCDM2 = d;
break;  // zero battery
       case '*':                                         LCDM2 = d + f;
break;  // low battery
       case '(':                                         LCDM2 = d + f + g;
break;  // medium battery
       case ')':                                         LCDM2 = d + e + f +
g;            break;  // full battery */
    }
}


// ******************************  writeWord
***********************************
void writeWord(const char *word)  // displays a word upto 7 characters -- why
7?
                                  // words must be in upper case (why?)
{
   unsigned int strLength = 0;   // variable to store length of word
   unsigned int i;               // dummy variable

   strLength = strlen(word);     // get the length of word now
   for (i = 1; i <= strLength; i++) // display word
   {
     writeLetter(strLength - i + 1,word[i-1]);  // displays each letter in
the word
   }

}


/**************** buzzerOn() ***********************/
void buzzerOn(void)
{
  FLL_CTL0 |= XCAP10PF;                 // Configure load caps
  P1DIR |= BIT2|BIT0;                   // P1.2,0 output
  P1SEL &= ~BIT2;                       // P1.2 I/O option
  P1OUT &= ~BIT2;                       // P1.2 output = 0
  P1SEL |= BIT0;                        // P1.0 TA0 option
```

```
  CCTL0 = OUTMOD_7;                      // CCR0 reset/set
  CCR0 = 0x0f;                           // PWM Period
  TACTL = TASSEL_1 + MC_1 + ID_0;        // ACLK, up mode, 1 divider
}

void buzzerOn1(void)
{
  FLL_CTL0 |= XCAP10PF;                  // Configure load caps
  P1DIR |= BIT2|BIT0;                    // P1.2,0 output
  P1SEL &= ~BIT2;                        // P1.2 I/O option
  P1OUT &= ~BIT2;                        // P1.2 output = 0
  P1SEL |= BIT0;                         // P1.0 TA0 option
  CCTL0 = OUTMOD_7;                      // CCR0 reset/set
  CCR0 = 0x0A;                           // PWM Period
  TACTL = TASSEL_1 + MC_1 + ID_0;        // ACLK, up mode, 1 divider
}

void buzzerOn2(void)
{
  FLL_CTL0 |= XCAP10PF;                  // Configure load caps
  P1DIR |= BIT2|BIT0;                    // P1.2,0 output
  P1SEL &= ~BIT2;                        // P1.2 I/O option
  P1OUT &= ~BIT2;                        // P1.2 output = 0
  P1SEL |= BIT0;                         // P1.0 TA0 option
  CCTL0 = OUTMOD_7;                      // CCR0 reset/set
  CCR0 = 0x08;                           // PWM Period
  TACTL = TASSEL_1 + MC_1 + ID_0;        // ACLK, up mode, 1 divider
}

void buzzerOn3(void)
{
  FLL_CTL0 |= XCAP10PF;                  // Configure load caps
  P1DIR |= BIT2|BIT0;                    // P1.2,0 output
  P1SEL &= ~BIT2;                        // P1.2 I/O option
  P1OUT &= ~BIT2;                        // P1.2 output = 0
  P1SEL |= BIT0;                         // P1.0 TA0 option
  CCTL0 = OUTMOD_7;                      // CCR0 reset/set
  CCR0 = 0x02;                           // PWM Period
  TACTL = TASSEL_1 + MC_1 + ID_0;        // ACLK, up mode, 1 divider
}

void buzzerOn4(void)
{
  FLL_CTL0 |= XCAP10PF;                  // Configure load caps
  P1DIR |= BIT2|BIT0;                    // P1.2,0 output
  P1SEL &= ~BIT2;                        // P1.2 I/O option
  P1OUT &= ~BIT2;                        // P1.2 output = 0
  P1SEL |= BIT0;                         // P1.0 TA0 option
  CCTL0 = OUTMOD_7;                      // CCR0 reset/set
  CCR0 = 0x0C;                           // PWM Period
  TACTL = TASSEL_1 + MC_1 + ID_0;        // ACLK, up mode, 1 divider
}

/***************** buzzerOff() ************************/
```

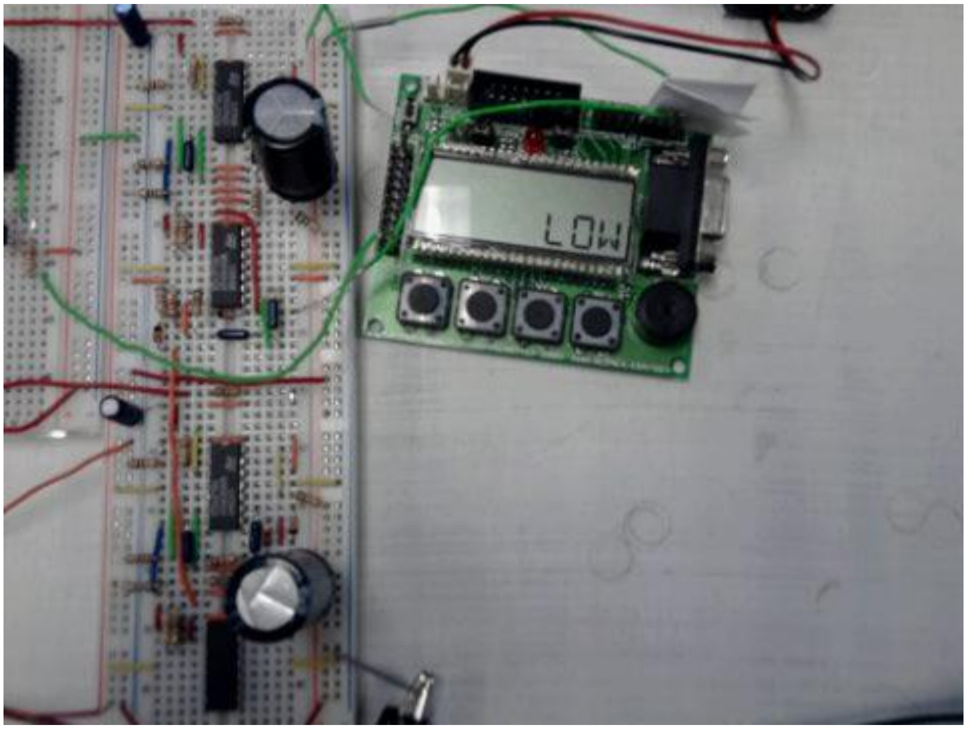
```
void buzzerOff(void)
{
  TACTL = MC_0;                    // Stop Timer
  P1DIR |= BIT2|BIT0;              // P1.2,0 output
  P1SEL &= ~(BIT2|BIT0);           // P1.2,0 I/O option
  P1OUT &= ~(BIT2|BIT0);           // P1.2,0 output = 0
}


void LEDOff(void)
{
  P2DIR |= (BIT7|BIT6|BIT5|BIT4);      // Set P2.7-2.4 to output direction
  P2SEL &= ~(BIT7|BIT6|BIT5|BIT4);     // P2.7-2.4 I/O option
  P2OUT |= (BIT7|BIT6|BIT5|BIT4);      // P2.7-2.4 output = 1 (LEDs off)
}

void LEDdisplayHex(unsigned char num)
{
  unsigned char   tmp_num;

  tmp_num = (~num)<<4;
  P2OUT = tmp_num & 0xF0;
}
```

SECTION E – COMPLETED PROJECT PHOTOS

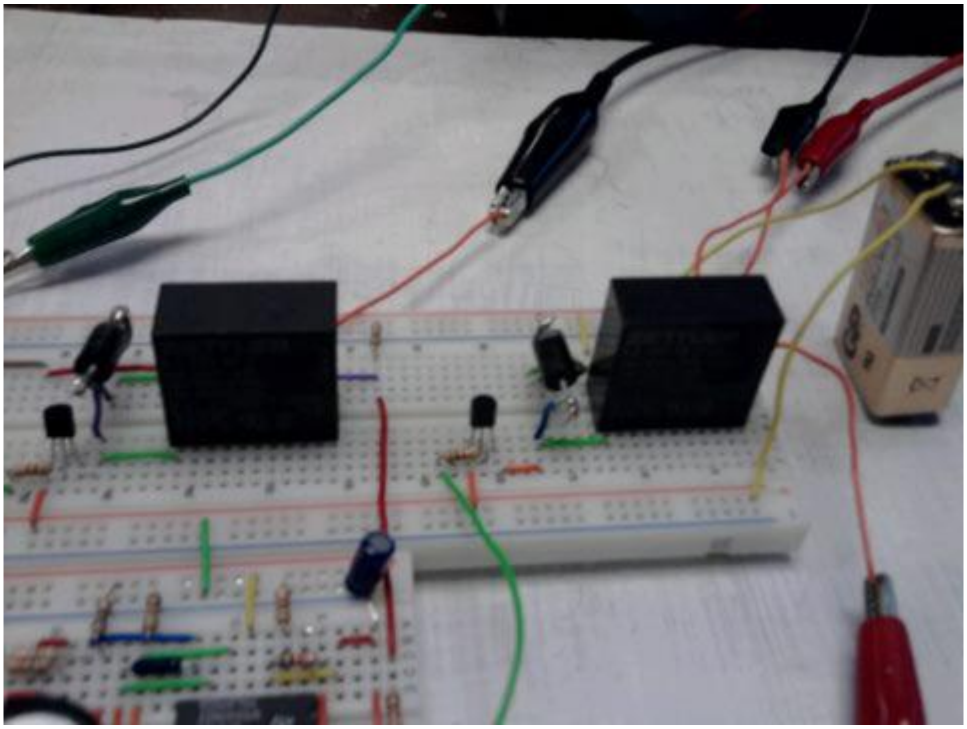
Shielded Transducer



MSP430 and Analog Circuitry

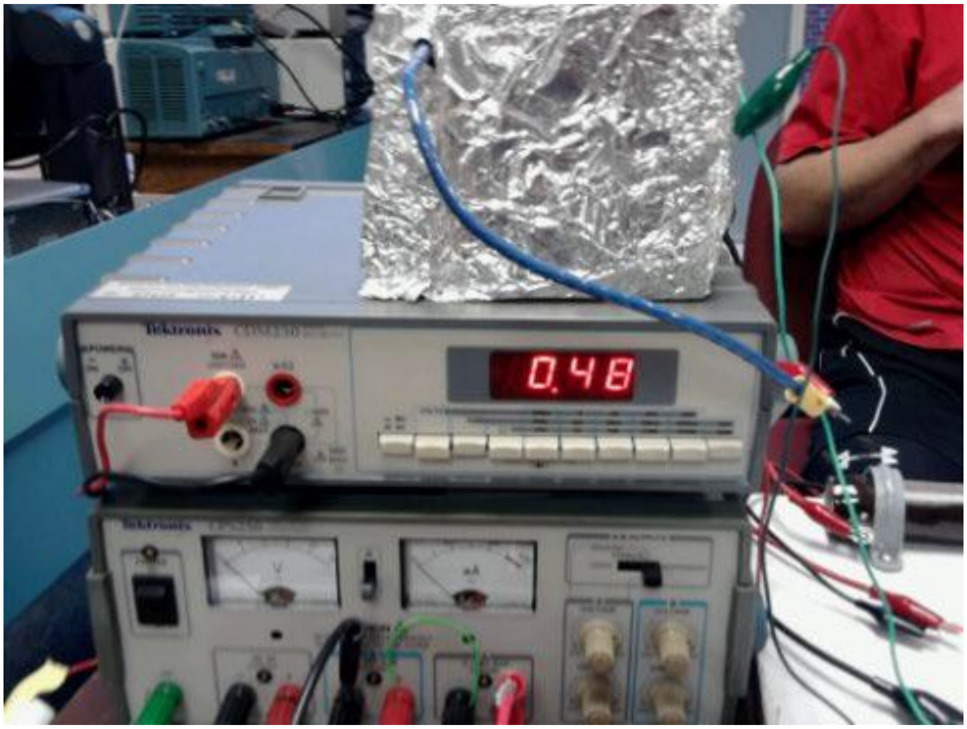


Relay Drivers/Relays/"Recloser"

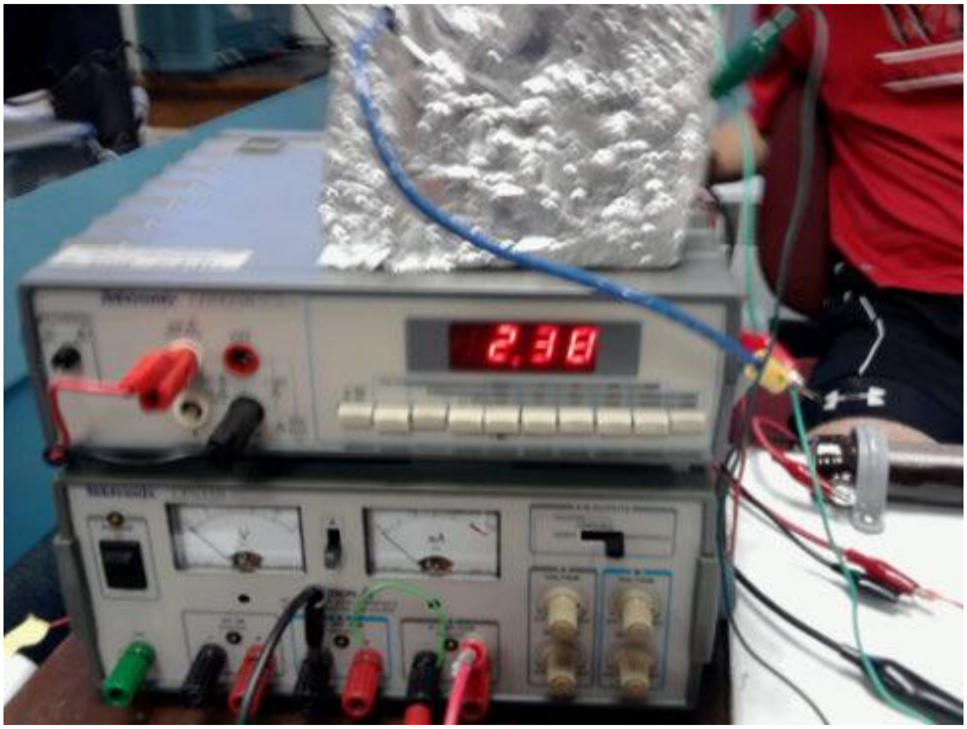


Full Apparatus

Normal Operation



Fault



SECTION F – APPENDED DATA SHEETS OF VARIOUS PRODUCTS

1) Zettler AZ755 Power Relay

2) 200 Ohm, 225 Watt Power Resistor

3) Cherry Switch

4) Relay Driving Basics

AZ755

20 AMP MINIATURE POWER RELAY

FEATURES

- Dielectric strength 5000 Vrms
- Low cost
- Epoxy sealed version available
- 20 Amp switching — single pole contacts
- Isolation spacing greater than 8mm
- UL, CUR file E44211
- TÜV file R9659060

CONTACTS

Arrangement	SPST (1 Form A, 1 Form B) SPDT (1 Form C)
Ratings	Resistive load: Max. switched power: 480 W or 5540 VA Max. switched current: 20 A Max. switched voltage: 150 VDC* or 380 VAC * Note: If switching voltage is greater than 30 VDC, special precautions must be taken. Please contact the factory.
Rated Load **UL, CUR**	20 A at 277 VAC N.O. resistive, 50k cycles 16 A at 240 VAC general use, 100k cycles 12 A at 277 VAC N.O. resistive., 100k cycles 20 A at 24 VDC resistive 1 HP 240 VAC TV-8 120 VAC N.O. (silver tin oxide only)
TÜV	16 A at 30 VDC, 250 VAC resistive, 100k cycles 13 A at 420 VAC res., 100k cycles (1 Form A)
Material	Silver cadmium oxide or silver tin oxide
Resistance	< 50 milliohms initially (24 V, 1 A voltage drop method)

COIL

Power	
At Pickup Voltage (typical)	270 mW
Max. Continuous Dissipation	1.9 W at 20°C (68°F) ambient
Temperature Rise	34°C (61°F) at nominal coil voltage
Temperature	Max. 130°C (266°F)

NOTES

1. All values at 20°C (68°F).
2. Relay may pull in with less than “Must Operate” value.
3. Specifications subject to change without notice.

GENERAL DATA

Life Expectancy	Minimum operations
Mechanical	5 x 10^6
Electrical	5 x 10^4 at 16 A 250 VAC Res. 2 x 10^4 at 20 A 277 VAC Res.
Operate Time (typical)	8 ms at nominal coil voltage
Release Time (typical)	5 ms at nominal coil voltage (with no coil suppression)
Dielectric Strength (at sea level for 1 min.)	5000 Vrms coil to contact 1000 Vrms between open contacts
Insulation Resistance	1000 megohms min. at 20°C 500 VDC 50% RH
Dropout	Greater than 10% of nominal coil voltage
Ambient Temperature	At nominal coil voltage
Operating	-40°C (-40°F) to 85°C (185°F)
Storage	-40°C (-40°F) to 105°C (221°F)
Vibration	0.062" (1.5 mm) DA at 10–55 Hz
Shock	10 g
Enclosure	P.B.T. polyester
Terminals	Tinned copper alloy, P.C.
Max. Solder Temp.	270°C (518°F)
Max. Solder Time	5 seconds
Max. Solvent Temp.	80°C (176°F)
Max. Immersion Time	30 Seconds
Weight	18.5 grams
Packing unit in pcs	50 per plastic tray / 500 per carton box

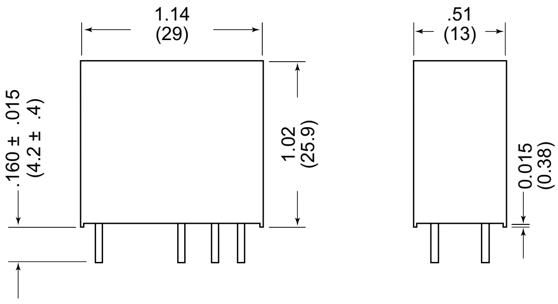
AZ755

RELAY ORDERING DATA

COIL SPECIFICATIONS				ORDER NUMBER*	
Nominal Coil VDC	**Must Operate VDC**	**Max. Continuous VDC**	**Coil Resistance Ohm**	**Form A (SPST)**	**Form C (SPDT)**
5	3.6	9.4	47 ±10%	AZ755–1A–5D	AZ755–1C–5D
6	4.3	11.4	69 ±10%	AZ755–1A–6D	AZ755–1C–6D
9	6.5	17.4	155 ±10%	AZ755–1A–9D	AZ755–1C–9D
12	8.6	22.8	275 ±10%	AZ755–1A–12D	AZ755–1C–12D
18	13.0	27.9	620 ±10%	AZ755–1A–18D	AZ755–1C–18D
24	17.3	45.7	1,100 ±15%	AZ755–1A–24D	AZ755–1C–24D
48	34.6	89.0	4,400 ±15%	AZ755–1A–48D	AZ755–1C–48D
60	43.2	115.3	6,880 ±15%	AZ755–1A–60D	AZ755–1C–60D
110 **	73.9	170.5	22,900 ±15%	AZ755–1A–110D	AZ755–1C–110D

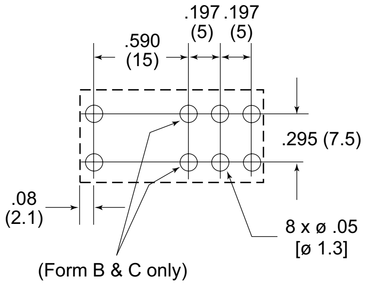
* Substitute “1B” in place of “1A” or “1C” to indicate 1 Form B contact arrangement.
Add suffix “E” at the end of order number for epoxy sealed version. Add suffix “A” for silver tin oxide contacts. Add suffix “F” for Class F.
** 110VDC coil not TÜV approved.

MECHANICAL DATA



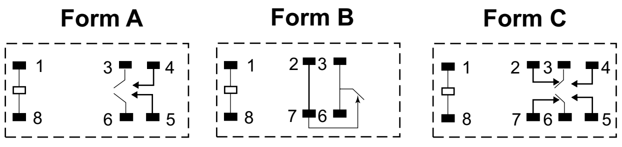
Terminal No.	Dimensions Tol.: ± 0.005 (0.13)
1,2,4,5,7,8	0.018 (0.457) x 0.038 (0.965)
3,6	0.011 (0.279) x 0.038 (0.965)

PC BOARD LAYOUT



Viewed toward terminals

WIRING DIAGRAMS



Viewed toward terminals

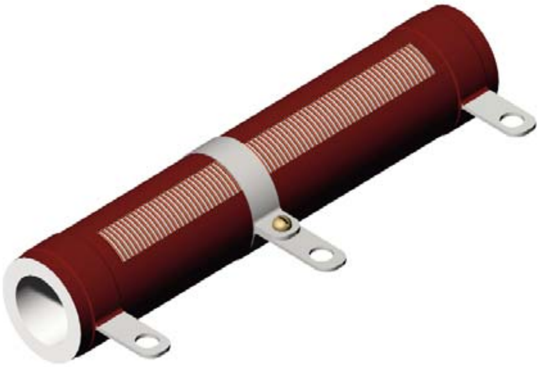
Dimensions in inches with metric equivalents in parentheses. Tolerance: ± .010"

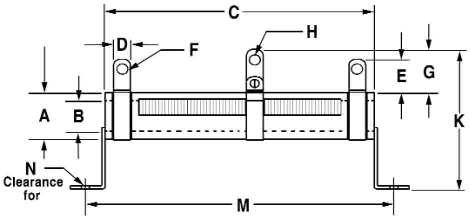
AVT
AST

ADJUSTABLE WIREWOUND RESISTORS

12 WATTS THRU 225 WATTS

H.E.I. Adjustable Resistors are constructed with steatite ceramic cores and terminated with welded terminals and wound with the finest alloy resistance wires welded to the terminals. Our special formula of vitreous enamel (**AVT**) or silicone (**AST**) coatings are then used to insulate the resistors. A section of the windings are made bare for connecting to a band terminal that implements the adjustment function. This construction insures long life, durability, and reliability.





A wide variety of special resistor core sizes are available. Any tapped, combination fixed and adjustable can be made to custom specifications, including tolerance, brackets, and terminals.

TYPE	WATT	A TYP	B +/-.032(0.8)	C +/-.032(0.8)	D +/-.032(0.8)	E TYP	F TYP	G TYP	H TYP	K TYP	M TYP	N
AVT-10	12	.31 (7.9)	.19 (4.8)	1.75 (44.5)	.19 (4.8)	.44 (11.2)	.10 (2.5)	.50 (12.7)	.10 (2.5)	.94 (23.9)	2.19 (55.6)	6-32
AVT-25	25	.56 (14.2)	.31 (7.9)	2.00 (50.8)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	1.69 (42.9)	2.75 (69.8)	8-32
AVT-25A	30	.75 (19)	.50 (12.7)	2.00 (50.8)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	2.31 (58.7)	2.88 (73.2)	10-32
AVT-25B	30	.63 (16)	.45 (11.4)	2.00 (50.8)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	1.69 (42.9)	2.63 (66.8)	8-32
AVT-50	50	.56 (14.2)	.31 (7.9)	4.00 (102)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	1.69 (42.9)	4.75 (121)	8-32
AVT-50A	60	.75 (19)	.50 (12.7)	4.00 (102)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	2.31 (58.7)	4.88 (124)	10-32
AVT-50B	70	.75 (19)	.55 (14)	4.50 (114)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	2.31 (58.7)	5.38 (137)	10-32
AVT-75	75	.56 (14.2)	.31 (7.9)	6.00 (152)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	1.69 (42.9)	6.75 (171)	8-32
AVT-75A	90	.75 (19)	.50 (12.7)	6.00 (152)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	2.31 (58.7)	6.88 (175)	10-32
AVT-100	100	.75 (19)	.50 (12.7)	6.50 (165)	.25 (6.4)	.56 (14.2)	.17 (4.3)	.69 (17.5)	.17 (4.3)	2.31 (58.7)	7.38 (187)	10-32
AVT-130	130	1.13 (28.7)	.75 (19)	6.50 (165)	.31 (7.9)	.63 (16)	.17 (4.3)	.88 (22.4)	.17 (4.3)	2.81 (71.4)	7.38 (187)	10-32
AVT-160	175	1.13 (28.7)	.75 (19)	8.50 (216)	.31 (7.9)	.63 (16)	.20 (5.1)	.88 (22.4)	.17 (4.3)	2.81 (71.4)	9.38 (238)	10-32
AVT-200	225	1.13 (28.7)	.75 (19)	10.50 (267)	.31 (7.9)	.63 (16)	.20 (5.1)	.88 (22.4)	.17 (4.3)	2.81 (71.4)	11.38 (289)	10-32

AVT - VITREOUS **AST - SILICONE**

inches (mm)

ORDER INFORMATION

AVTK - 10 - 250 - QC - BKTS

POWER

TYPE
V=VITREOUS
S=SILICON

INCLUDE FOR
TOLERANCE(K=±10%)

RESISTANCE

ADD FOR
QUICK
CONNECT

ADD FOR
STANDARD
BRACKETS

OPTIONAL FEATURES AVAILABLE

RESISTANCE TOLERANCE: Standard tolerance is ±10%.
Available Tolerances: F = ±1%, J = ±5%, K = ±10%

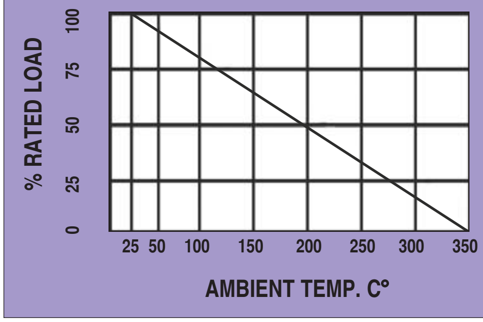
MOUNTING BRACKETS: Push-in friction grip mounting brackets are available as shown above. When required, add “BKTS” to the part number.

QUICK CONNECT TERMINALS: Resistors with a core O.D. of 9/16 or 3/4 inch can be supplied with Quick Connect Terminals except the adjustable band. As “QC” to the part number as shown.

RESISTANCE VALUE CHART

.10	.75	10	30	68	125	300	600	1.1K	3.0K	6.8K
.13	1.0	12	33	75	150	330	680	1.2K	3.5K	7.5K
.15	1.5	15	35	82	180	350	700	1.5K	3.9K	8.0K
.20	2.0	18	39	100	200	390	750	1.8K	4.0K	8.2K
.25	3.0	20	40	120	220	400	800	2.0K	4.7K	10.0K
.30	4.0	22	47	125	225	470	820	2.2K	5.0K	12.5K
.33	5.0	25	50	150	250	500	900	2.5K	5.6K	15.0K
.50	7.5	27	56	180	270	560	1.0K	2.7K	6.0K	20.0K

WATTAGE DERATING CHART FOR HIGHER AMBIENT TEMPERATURES



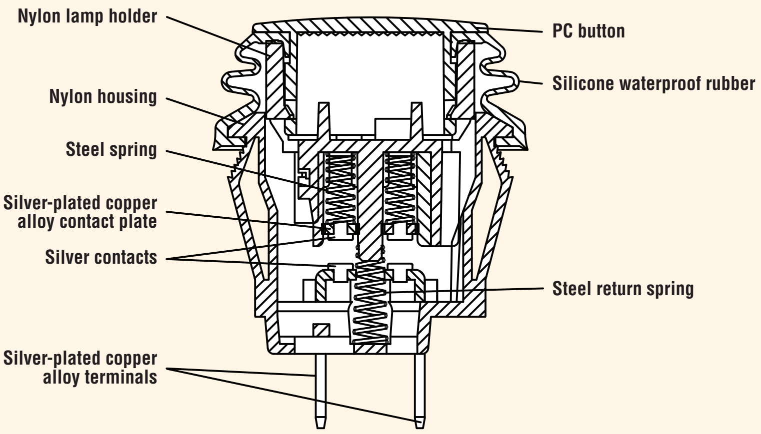
CHERRY

WATERPROOF ROUND PUSHBUTTON SWITCH

KF Series

Features

- Enclosed switch complying to IP65
- Available in single- or double-pole
- Lighted and non-lighted versions
- Maintained-action and momentary options
- Snap-in mounting
- Mechanical life: 100,000 cycles
- DC electrical life: 10,000 cycles



Electrical Ratings

Switch Series	VDC	UL1054 Rating	Electrical Life at Rated Load According to UL (Min. Operations)
KF	20A 14VDC; 10A 28VDC	10A 125VAC; 6A 250VAC; 1/2HP, 125/250VAC	6,000

Specifications

Electrical

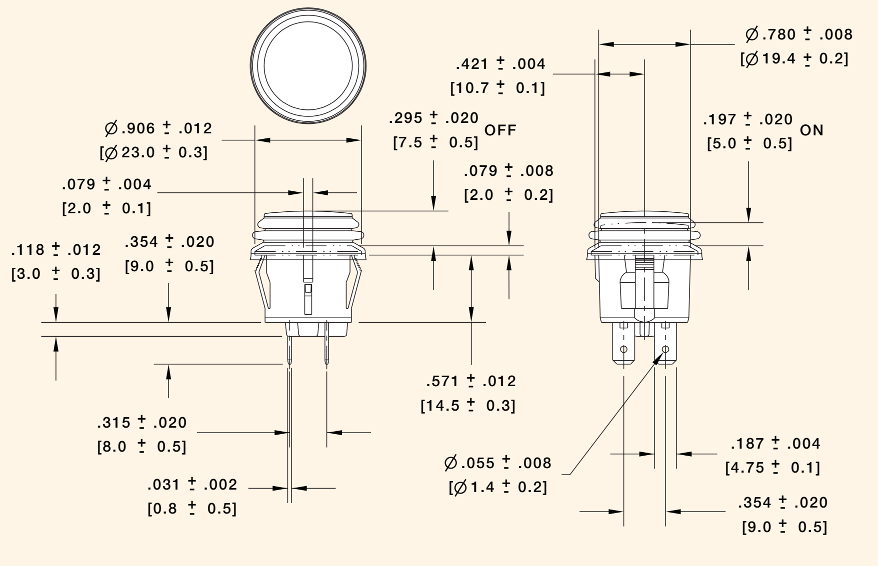
Dielectric Strength:	1500VAC for 1 minute
Insulation Resistance:	10^8 ohms min at 500VDC
Initial Contact Resistance:	0.050 ohms max
Temperature Rating:	-25°C to +85°C

Materials

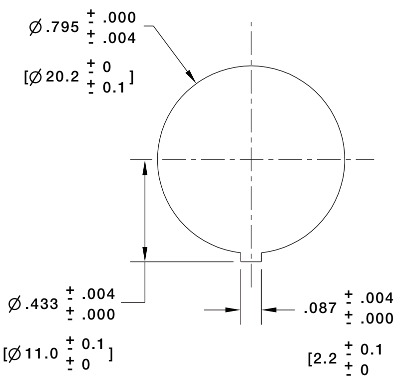
Actuator:	Polycarbonate 94V-2
Housing:	Nylon 66 94V-2
Spring:	Steel
Contacts:	Silver Alloy
Movable Arm:	Silver-plated Copper Alloy
Terminals:	Silver-plated Copper Alloy
Waterproof Rubber:	Silicone

CHERRY

Dimensions inches (mm)

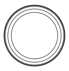
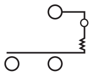
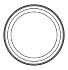
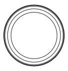
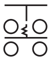
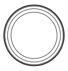
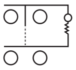
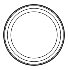


Panel Cut-Out Dimensions for Panel Thickness inches (mm)



Panel Thickness
.028 to .098 (0.7 to 2.5)

Examples of Common Part Numbers and Descriptions inches (mm)

Part Number	Circuitry	Switch Function	Terminal Type	Actuator/ Housing Color	Legend Marking
KFB2ANA1BBB		On – Off	0.187 (4.75) Quick Connect	Black/Black	
KFB2ALA2RBB		On – Off (Illuminated)	0.187 (4.75) Quick Connect	Red/Black (Red Lens)	
KFB2DNA1BBB		(On) – Off	0.187 (4.75) Quick Connect	Black/Black	
KFB3ANA1BBB		On – Off	0.187 (4.75) Quick Connect	Black/Black	
KFB3ALA2RBB		On – Off (Illuminated)	0.187 (4.75) Quick Connect	Red/Black (Red Lens)	

Ordering Information

KF B 3 A N A 1 B B B

Series

Code	Contact Rating
B	10A 125VAC (UL) 6A 250VAC (UL) 1/2HP 125/250VAC (UL) 20A 14VDC 10A 28VDC

Code	Circuitry
2	Single-Pole
3	Double-Pole

Code	Switch Function
A	On – Off
D	(On) – Off

Code	Non-Lighted/Lighted
N	Non-Lighted
L	110V Neon Bulb
E	220V Neon Bulb
M	12V LED Dependent
T	12V LED Independent

Code	Terminal Type
A	0.187 (4.75) Quick Connect

Code	Actuator Type
1	Non-Lighted
2	Lighted

Code	Actuator Color
A	Amber
B	Black
G	Green
R	Red

Code	Housing Color
B	Black

Code	Actuator Marking
B	Blank

RELAY DRIVING BASICS

Relays are components which allow a low-power circuit to switch a relatively high current on and off, or to control signals that must be electrically isolated from the controlling circuit itself. Newcomers to electronics sometimes want to use a relay for this type of application, but are unsure about the details of doing so. Here's a quick rundown.

To make a relay operate, you have to pass a suitable 'pull-in' and 'holding' current (DC) through its energising coil. And generally relay coils are designed to operate from a particular supply voltage — often 12V or 5V, in the case of many of the small relays used for electronics work. In each case the coil has a resistance which will draw the right pull-in and holding currents when it's connected to that supply voltage. So the basic idea is to choose a relay with a coil designed to operate from the supply voltage you're using for your control circuit (and with contacts capable of switching the currents you want to control), and then provide a suitable 'relay driver' circuit so that your low-power circuitry can control the current through the relay's coil. Typically this will be somewhere between 25mA and 70mA.

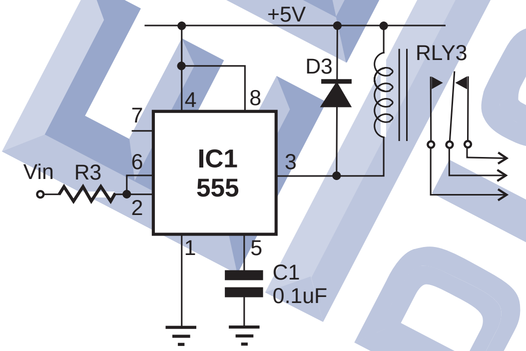
Often your relay driver can be very simple, using little more than an NPN or PNP transistor to control the coil current. All your low-power circuitry has to do is provide enough base current to turn the transistor on and off, as you can see from diagrams A and B.

In A, NPN transistor Q1 (say a BC337 or BC338) is being used to control a relay (RLY1) with a 12V coil, operating from a +12V supply. Series base resistor R1 is used to set the base current for Q1, so that the transistor is driven into saturation (fully turned on) when the relay is to be energised. That way, the transistor will have minimal voltage drop, and hence dissipate very little power — as well as delivering most of the 12V to the relay coil.

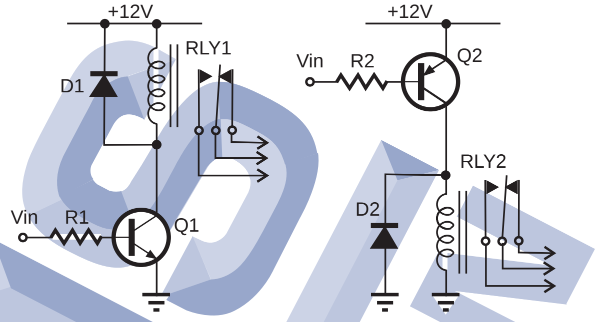
How do you work out the value of R1? It's not hard. Let's say RLY1 needs 50mA of coil current to pull in and hold reliably, and has a resistance of 240Ω so it draws this current from 12V. Our BC337/338 transistor will need enough base current to make sure it remains saturated at this collector current level.

To work this out, we simply make sure that the base current is greater than this collector current divided by the transistor's minimum DC current gain hFE. So as the BC337/338 has a minimum hFE of 100 (at 100mA), we'd need to provide it with at least 50mA/100 = 0.5mA of base current.

In practice, you'd give it roughly double this value, say 1mA of base current, just to make sure it does saturate. So if your control signal Vin was switching between 0V and +12V, you'd give R1 a value of say 11kΩ, to provide the 1mA of base current needed to turn on both Q1 and the relay.



C. 555 'timer' IC as relay driver. Relay on for Vin > 2/3Vcc (3.3V), & off for Vin < 1/3Vcc (1.66V)



A. NPN driver, relay on for Vin = +12V & off for Vin = 0V

B. PNP driver, relay on for Vin = 0V & off for Vin = +12V

If our relay has a coil resistance of say 180Ω, so that it draws say 67mA at 12V, we'd need to reduce R1 to say 8.2kΩ, to increase the base current to about 1.4mA. Conversely if the relay coil is 360Ω and draws only 33mA, we could increase R1 to 15kΩ, giving about 0.76mA of base current. Each time we go for about twice the relay coil current divided by Q1's hFE — get the idea?

As you can see a power diode D1 (1N4001 or similar) is connected across the relay coil, to protect the transistor from damage due to the back-EMF pulse generated in the relay coil's inductance when Q1 turns off.

The basic NPN circuit in diagram A is fine if you want the relay to energise when your control voltage Vin is **high** (+12V), and be off when Vin is low (0V). But what if you want the opposite? That's where you'd opt for a circuit like that shown in diagram B, using a PNP transistor like the BC327 or BC328. This is essentially the same circuit as in A, just swung around to suit the PNP transistor's polarity.

This time transistor Q2 will turn on and energise the relay when Vin is **low** (0V), and will turn off when Vin is high (+12V). Otherwise everything works just as before, and the value of base resistor R2 is worked out in the same way as for R1. In fact because the minimum hFE of the BC327/328 PNP transistors is also 100 at 100mA, you could use exactly the same values of R2 to suit each relay resistance/current.

The simple transistor driver circuits of A and B are very low in cost, and are generally fine for driving most relays. However there may be occasions, such as when your control circuit is based on CMOS logic, where the base current needed by these circuits is a bit too high.

For these situations the circuit shown in C might be of interest, because it needs rather less input current. As you can see it uses a readily available and very low cost 555 IC as the relay driver, plus only one extra component: bypass capacitor C1.

Although we normally think of the 555 as a timer/oscillator, it's actually very well suited for driving a small relay. Output pin 3 can both source and sink 200mA (enough to handle most small relays comfortably), and the internal flipflop which controls its output stage is triggered swiftly between its two states by internal comparators connected to the two sensing inputs on pins 2 and 6. When these pins are taken to a voltage above 2/3 the supply voltage, the output switches low (0V); then they are taken below 1/3 the supply voltage, the output swings high. And the 555 can happily work at 5V, as you can see, so it's very suitable for driving a 5V relay coil from this supply voltage.

Because the sensing inputs of the 555 are voltage sensing and need only a microamp or so of current, the value of input resistor R3 can be much larger than for the transistor driver circuits. Typically you'd use a value of say 100kΩ, or even 220kΩ for a circuit operating from 12V.

Although the push-pull output stage of the 555 automatically shunts the relay coil when pin 3 is high, damping the back-EMF, it's probably still a good idea to fit diode D3 as well — especially when using this circuit from a 12V supply. That's because the negative-going back-EMF pulse could cause damage to the transistors inside the 555.

Capacitor C1 is fitted to make sure that the 555 doesn't turn on the relay in response to noise spikes on te supply line.

By the way if you need the very low input current of this circuit, but want to make the relay operate when Vin is low rather than high, simply connect the relay coil and D3 from pin 3 of the 555 to ground — just like the arrangement shown in diagram B.

Finally in all of these circuits, it's a good idea to fit the supply line of the relay/driver stage with a reasonably high value of bypass capacitor (say 100uF), to absorb the current transients when the relay turns on and off. This will ensure more reliable operation, and help prevent interference with the operation of your control circuitry.

THE MAJOR INTERNATIONAL TESTING AGENCIES

Most of the developed countries have national testing agencies and certification bodies, which perform testing and issue licences to certify that equipment complies with the electrical safety standards of that country (which are often based on, or derived from international standards).

Some of the main national testing/certification agencies are shown at right, together with the symbols that are used on equipment to certify that it has been tested and complies to their standards.

Germany	VDE	DVE	France	LCIE	[illegible]
Austria	OVE	ÖVE	Netherlands	KEMA	KEMA KEUR
Belgium	CEBEC	CEBEC	Norway	NEMKO	N
Canada	CSA	SA	Sweden	SEMKO	S
Denmark	DEMKO	D	Switzerland	SEV	+S
Finland	FIMKO	FI	United States	UL	UL RU

Australia's standards and certification agency is Standards Australia (SA), of PO Box 1055, Strathfield (www.standards.org.au). The testing is done by accredited testing laboratories.