# Design and construction of a numerical over current relay

A thesis

Submitted to the EEE Department of BRAC University

by


MD RAYHANUL AMIN RUMI (11121030)

S.M. AMANAT HOSSAIN (11221057)

RAZAUL HAIDER HANIF (11221012)

MAHDI MD MOSTAFA (12121117)


in partial fulfillment of the requirements for the Degree of Bachelor of Science in Electrical and Electronics Engineering

December 2015

## Declaration

This is to declare that this thesis named "Design and construction of a numerical over current relay" is submitted by the author listed below for the degree of Bachelor of Science in Electrical and Electronics to the department of Electrical and Electronics Engineering under the school of Engineering and Computer Science, BRAC University. We, therefore affirm that the project work and results are conducted solely by us and no other. Material of the study and work found by other researcher has been properly referred acknowledged. This thesis paper, neither in whole nor in part, has been previously submitted elsewhere for appraisal.

Date of Submission: 17 December 2015.


MD RAYHANUL AMIN RUMI

(11121030)

S.M. AMANAT HOSSAIN

(11221057)

RAZAUL HAIDER HANIF

(11221012)

MAHDI MD MOSTAFA

(12121117)

Thesis Supervisor

Dr. S. Shahnawaz Ahmed

Professor

Department of Electrical and Electronic Engineering

Bangladesh University of Engineering and Technology

**ACKNOWLEDGEMENT**

We express our sincere gratitude to our supervisor, Dr. S. Shahnawaz Ahmed, Professor, Department of Electrical and Electronic Engineering, Bangladesh University of Engineering and Technology (BUET) for his valuable guidance and assistance to complete this project. We are grateful to our peers. We also take this opportunity to thank every other individual who has supported, encouraged and lent us hand whenever it was required throughout our project.

.

# Abstract

The project aims at the design and the construction of a laboratory replica for a commercial numerical over current relay using microcontroller and micro CT. It will acquire, sample and covert into digital form the analog current signals from a power system in real time. Then it simulates the working principles of a conventional electromechanical overcurrent relay in through a numerical (mathematical) model. Appropriate equations and algorithm have been used. When a current in a part of a power system exceeds a pre-set value either due to a fault or a large overload the relay will isolate the part through sending trip signal to a circuit breaker. In the developed laboratory model, the system is assembled by a lamp board and the circuit breaker is simulated by a hardware solid state (MOSFET) switch. The constructed relay has been tested extensively for various values of load current and two different tripping characteristics (Inverse Definite Minimum Time: IDMT and Definite Time Lag: DTL) have been verified.

The project is expected to help students acquire skills in digital logic circuits design and implementation, and applying it in a power system for isolating the faulty part. As commercial numerical relay is expensive, so we have developed for lab demonstration a prototype using cheap and off-the-shelf components to show the working procedures of a numerical over current relay in a power system. The project will also help students develop hands-on skill on numerical relay and enrich their knowledge on the working principle of a relay-breaker combination circuit.

## Table of Content

## List of Figures — Page



## Abbreviation

| | |
|---|---|
| CT | Current Transformer |
| PT | Potential Transformer |
| IDMT | Inverse Definite Minimum Time |
| DTL | Definite Time Lag |
| IDE | Integrated Development Environment |
| USB | Universal Serial Bus |
| IC | Integrated Circuit |
| Arrayval | An array inside the code |
| Ms | Millisecond |
| RMS | Root Mean Square |
| Top | Operating Time of Relay |

## List of table Page

## 1. Introduction:

Electrical Power System protection is required for protection of both user and the system equipment itself from fault, hence electrical power system is not allowed to operate without any protection devices installed. Power System fault is defined as undesirable condition that occurs in the power system. These undesirable conditions are short circuit, current leakage, ground short, over current and over voltage. With the increasing loads, voltages and short-circuit duty in distribution system, over current protection has become more important role today. The ability of protection system is demanded not only for economic reason but also consumers just expect 'reliable' service. In a Power System Protection, the system engineer needs a device that can monitor current, voltage, frequency and in some case over power in the system. Thus a device called Protective Relay is created to serve the purpose. The protective relay is most often relay coupled with Circuit Breaker such that it can isolate the abnormal condition in the system. In the interest of reliable and effective protection, some designers of power distribution select relay alternative to electro-magnetic circuit breakers as a method of circuit protection.

A "Numerical over Current Relay" is a type of protective relay which operates when the load current exceeds a preset value. In a typical application the over current relay is used for over current protection, connected to a current transformer and calibrated to operate at or above a specific current level. This project will attempt to design and formulate over current protection relay using micro controller, micro CT, solid state relay, resistor via coding with digital signal processing with the arduino. It includes a IDMT, DT relay which has a purpose of protection from over current. The micro controller will cause the circuit breaker to trip when the current from load current reaches the setting value in the micro controller. In order to design it, first the load current need to measure in order to monitor it using current transformer including testing the fault (over current) and when such condition arise, it will isolate the faulty part in the shortest time possible without harming the any other electrical devices. This project also includes algorithm for instantaneous over current relay and IDMT (Inverse Definite Minimum Time) relay and DTL (Definite lag time). The overcurrent relay of IDMT is the relay which starts to operate after the intended time delay. The time delay is also known as operation time. The advantage of the IDMT type overcurrent relay is that the greater the fault currents, the shorter are their operating time. Numerical relay is better known as present generation relay due to its standard technology. They use microprocessor within built software with predefined values for the limit and sensitivity of the fault. It is a compact size relay with very fast sensing ability with high time accuracy since it has a low CT burden. An overcurrent relay is implemented on a high speed and high performance digital signal processor. Thus, the overcurrent relays must have high reliability and accuracy to detect any fault currents present and determine the operation time. The overcurrent relay of IDMT is the relay which starts to operate after the intended time delay. The time delay is also known as operation time. The advantage of the IDMT relay is that the greater the fault currents, the shorter are their operating time. The overcurrent relay is now the latest technology used in the power system for its vast advantage and easy to denote faults.In this project, microcontroller will be used to control and operate the tripping coil in circuit breaker. This is basically a skill development and lab purpose demonstration for the undergrad student. In this project all the equipment are off the selves . It is easy to the students to deal with in a low range load like lamp board connected with 100 W light bulb to demonstrate. Increasing the amount of bulb increases the load current and the breaker will not trip until set current becomes smaller than line current. The project is designed for the undergrad students to have a clear view over protection system of a transmission line, the relays used in the power systems are very much expensive so we developed a lab model to show the same task happening.

## 1.1 Literature Review

In the last decades, several techniques were developed in the field of relay modernization to minimize the fault occurring at the power system transmission. The importance of the coordination of distinct protective device is increasing now since the equipment are very much expensive in this field. A power system consists of many number of equipment so eventually more number of circuit breakers and relays are required to protect the system from faulty situation of over current. A relay must receive sufficient chance to protect the zone under its primary protection whenever a fault is seen in the line, thus high current. If the primary protection does not remedy the fault, the back-up protection installed must initiate tripping, and therefore, overcurrent relay participation in power distribution network is a major concern of protection power engineer. The relay coordination in distribution network is highly constrained optimization problem. The demand for electrical power is increasing at a faster rate in mainly economically emerging countries as well as developing countries. So it is necessary to install of transmission lines reaching out all the areas of country. Further, the efficiency of transmission should be high when a large part of power is to be transmitted over very long distance. These transmission lines are required to be protected by comprehensive and quite involved protective schemes so that the power interruption are reduced to minimum with regard the times of interruption and the area are affected since this sector is an important area as well as high expense in required for the set up. The protective scheme must operate fast and selectively before the power system become unstable. We have gone through several papers on internet all are intend to minimize the faulty situation of the power system using various methods, one of the prime focus is working with numerical over current relay due to its high efficient working activity which uses latest technologies and to minimize the error to almost zero. Many papers on the issue definitely describe the important of the work. So, we know, how much important of the correct relay coordination is used nowadays to avoid mal operation in the transmission line. Many courses regarding this is taught to the undergrad and post grad student and several lab task is eventually helping all to ensure the best uses for the over current relay. Researcher's working day and night to produce the best quality numerical overcurrent relay with high efficiencies. This sort of protection are expensive to use for academic purpose thus we developed the relay for academic use rather using off the self-material since commercial relays are expensive to dealt with. This project demonstrates the working principle of the relay breaker combination circuit clearly to the undergraduate students.

## 1.2 Objective

This Project is expected to help the students achieve skills in digital logic circuits, applying it in a power system for isolating a part for facing over current. The main purpose of this project is to make a numerical relay suitable for academic use. So this project must be done in an economic way . Therefore all the elements used to construct this numerical relay are off the shelf and we did the coding in a much efficient way with a efficient algorithm. As a result this project does not cost too much. This project also demonstrates the working principle of a relay-breaker circuit. As this numerical relay is made for laboratory use, the main objective of this project is to help students developing their hands-on skills on numerical relay and also enrich their knowledge on the working principle of a relay-breaker circuit.

## 1.3 Organization of the thesis

In the first chapter, background and literature of overcurrent relay has been discussed along with scope and objective. The second chapter is mainly focused on the theory and overview of over current relay. It includes electromechanical as well as numerical over current relay. In chapter three, the main project work has been discussed with flowcharts.  Chapter three includes and construction of IDMT and DTL numerical overcurrent relay. Chapter four brings the overall conclusion of the project and explains the further work that can be done on this content and appendix contains the used codes for DTL and IDMT type numerical relay implementation.

# 2. THEORY OF OVERCURRENT RELAY

Overcurrent Relays in power system are like the hidden solders in the system, they become active when there is any fault, in order to protect system under its protective area. Faults may occur in any part of power system. In such condition a heavy fault current pass through the system. The relay detects intolerable or undesirable conditions and also can sense how serious the fault is (for faster or slower operation to trip). Relay must be operated correctly and make the circuit breaker tripped for disconnecting the faulty portion of the circuit from rest of the healthy system means the relay gives signal to circuit breakers whether to disconnect or remain silent to ensure safety and prevent damage to personnel and property in its serving area.

## 2.1 OVERVIEW OF ANALOG RELAY

**DEFINITE TIME:**

Definite Time Overcurrent Relay has two conditions that must be fulfilled for tripping, firstly current must surpass the pre-set current and secondly the fault must be continuous no less than the time equivalent to time setting of the relay. For operation definite time overcurrent relay working time is steady and this operation is free of the size of current over the pickup value as it has pickup and time dial settings, wanted time delay can be set with the assistance of an intentional time delay system with Constant tripping time autonomous of in feed variety and faulty area .

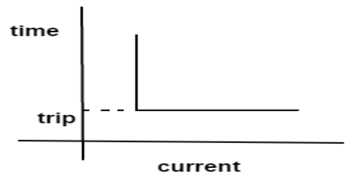



Fig2.1: Tripping characteristic of definite time relay

**INVERSE DEFINITE MULTIPLE TIME RELAY:**

Overcurrent Relay has inverse characteristics with respect to the currents being monitored. Most mainstream relays utilized on medium-and low-voltage frameworks for some, and current computerized transfers' attributes are still essentially in light of this kind of relay, it is advantageous concentrating on the operation of this in point of interest to comprehend the embraced in the computerized relays.

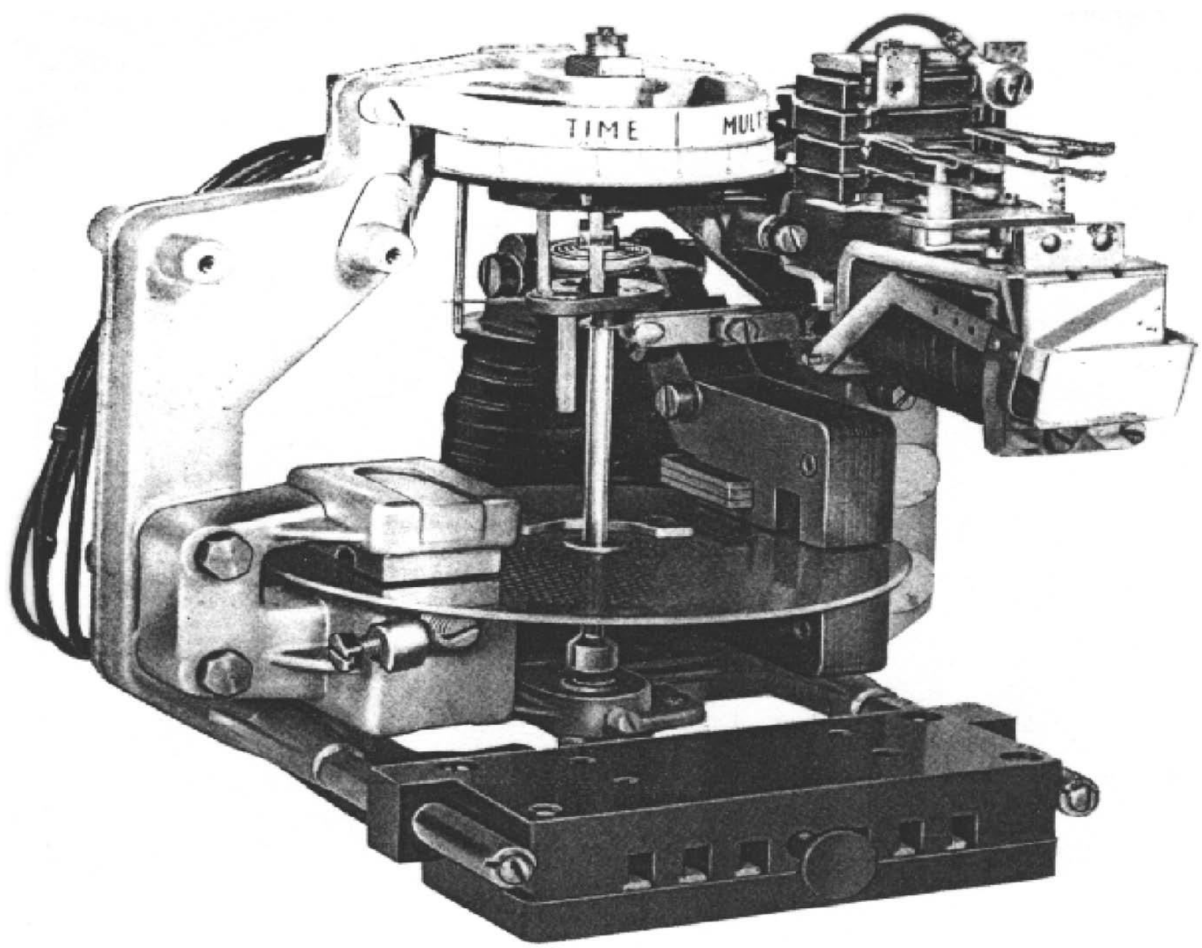



Figure2.2 mechanical relay

It comprises of an upper electromagnet that has been furnished with two windings one primary and the other auxiliary. Essential is joined with a current transformer in the line which is under insurance and is furnished with eight tappings. These tappings are joined with an attachment setting bridge by which the quantity of swings to be utilized can be balanced as a part of request to have the coveted current setting. The second winding called auxiliary is empowered by the induction impact and is wound over the central appendage of the upper magnet and also it is spread over the two appendages of the lower magnet. By this system, the leakage flux from the upper magnet entering the disk has been dislodged in phase from the flux entering the plate from the lower magnet. The diverting torque is produced on the circle in the design as of now clarified. The axle of the disc conveys a moving contact which connects two altered contacts after the plate has turned through a certain point which has been set some time recently. Any setting for this edge is conceivable fluctuating from 0 to 360°. The variety of this point confers to

the hand-off, different time settings. The speed of the disc is reliant upon the which thusly is subject to the current setting, the load current increments from this setting it will the speed of turn of the plate coming about into of operation time. Accordingly the time current of the relay follow opposite square law. The Definite Minimum Time characteristics of the relay are by the utilization of a saturated upper magnet. This guarantee there is no further increment in flux when the current has come to a sure esteem and any further expansion of Current won't influence the relay operation. This outcomes in a smoothed current time characteristic and the relay acquires its name as Inverse Definite Minimum Time Lag (I.D.M.T.) relay.
It can be seen that the working time of an IDMTL relay is Inversely Proportional to Current; it has a long working time at low products of setting Current and generally short working time at high Multiples of Setting Current. Two adjustments are possible on:

**The current pick-up or plug setting:** Sets the setting current by method for a plug bridge, which differs the effective turns on the upper electromagnet.

**The time multiplier setting**: Adjusts the working time at a given multiple of setting, by modifying by method for the torsion head, the separation that the disk needs to travel before contact is made.

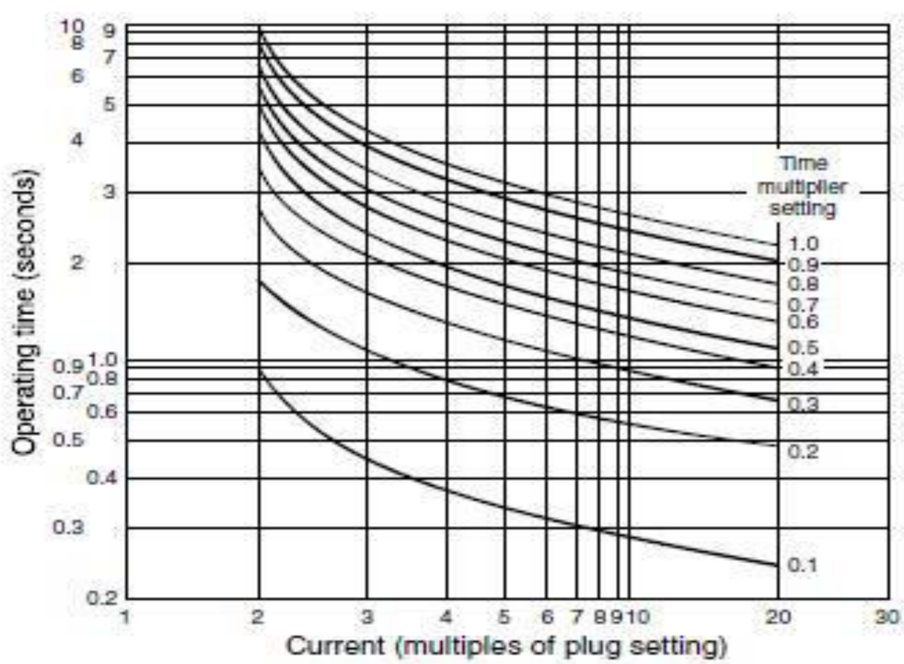



Fig2.3: Time/current characteristic

**Current (plug) pick-up setting**:

This setting decides the level of Current at which the transfers will get or its disk will begin to pivot. The relay should work at 130% setting and reset at 70% setting. In this connection, the plug setting is that current at which the working and limiting torques are in a condition of parity. By and by, the relay should not work at the setting, and to guarantee this, a relay may show a slight inclination to reset at the typical setting. The relay subsequently regularly grabs in the scope of 105–130% its present Plug Setting. Normally the accompanying scopes of ostensible current are utilized, giving a 1:4 proportion in seven.

**Time multiplier setting**

This dial pivots the disk and its going with drawing contact nearer to the settled contact, along these lines lessening the measure of separation to be gone by the moving contact, henceforth accelerating the stumbling time of the relay. The season of operation of the relay is picked by and large selecting the current and time plug settings.

**Burden**

It is the typical continuous load imposed on the Current transformers by the relay, ordinarily communicated in VA or in some cases in ohms. For electromechanical relays, this is ordinarily expressed as 3 VA ostensible. The present day electronic relays offer a much lower figure, which is one of their ideals. In any case, for the electromechanical sort, the determination of the plug setting has an impact on the Burden. As expressed before, the working loop is twisted to give Time/Current Curves of the same shape on each of the seven taps, which are chosen on the plug bridge.

## 2.2 NUMERICAL RELAY

A microchip Relay is a computerized electronic relay, which determines its qualities by method for a pre-programmed algorithm and counts (calculations), in light of the chose settings and the deliberate current and/or voltage signals.

**BASIC EQUATION:**

The formula used to determine the Inverse time characteristic in an overcurrent relay that consent to be scientifically characterized as takes after:

$t[s] = k\beta / [(I/I>)^{\alpha} - 1]$

Where

*T* = operating time in seconds

*K* = time multiplier

*I* = current value

*I>* = set current value.

The unit incorporates four determined qualities with distinctive degrees of Inverse. The level of backwards is controlled by the estimations of the Constants α and β

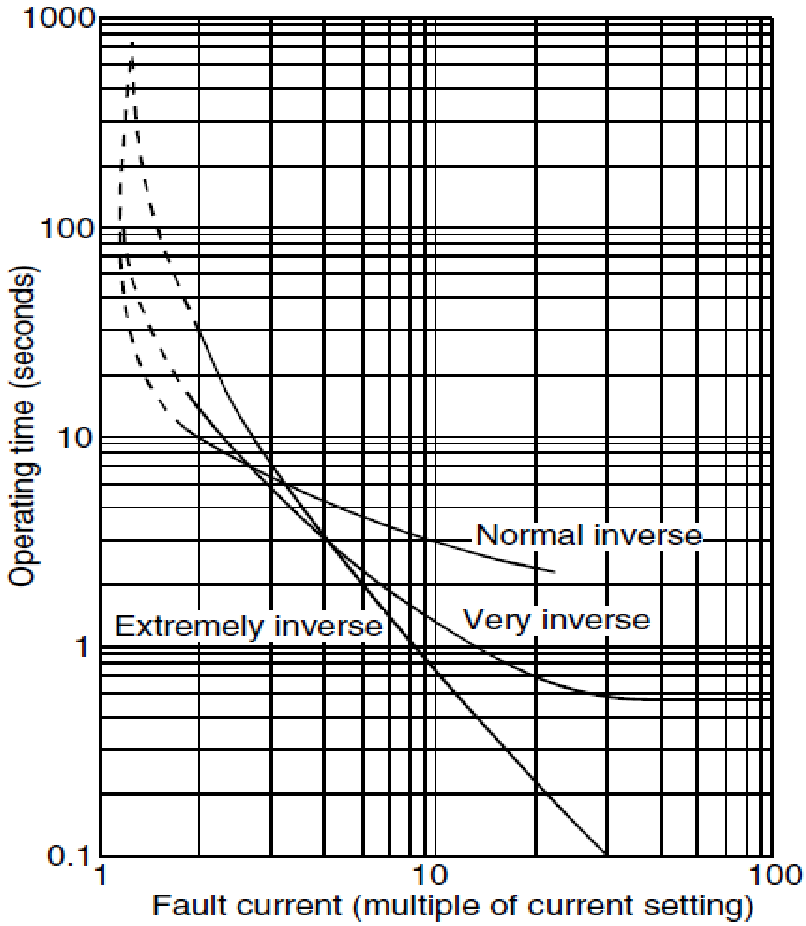



Fig2.4: fault current characteristic

Table2.1: Inverseness type

| **Degree of Inversity of the Characteristic** | α | B |
|---|---|---|
| Normal inverse | 0.02 | 0.14 |
| Very inverse | 1.00 | 13.50 |
| Extremely inverse | 2.00 | 80.00 |
| Long-time inverse | 1.00 | 120.00 |

**BASIC PRINCIPLE:**

In computerized transfers Digital Signals are utilized for information preparing rather than Analog signs. Simple signs are Continuous Signals and can't be prepared effectively as a result of their few constraints when contrasted with advanced signs. Computerized signs are in type of coded square heartbeats which speaks to discrete components of data. In computerized framework, the signs are in double shape just two discrete qualities alluded to as paired coefficients 0 and 1 or consistent values genuine and false. The quantity of paired digits expected to encode the different discrete components of data affects the outline of an advanced framework. The advanced framework by and large works on gatherings of 8 to 32 bits of data without a moment's delay. The scope of the computerized arrangement of encoding the data by a n bit gathering is 2 n. Thus advanced frameworks with bigger piece working gathering can prepare a more extensive scope of encoded data. The data to be handled may be literary, numerical and legitimate.

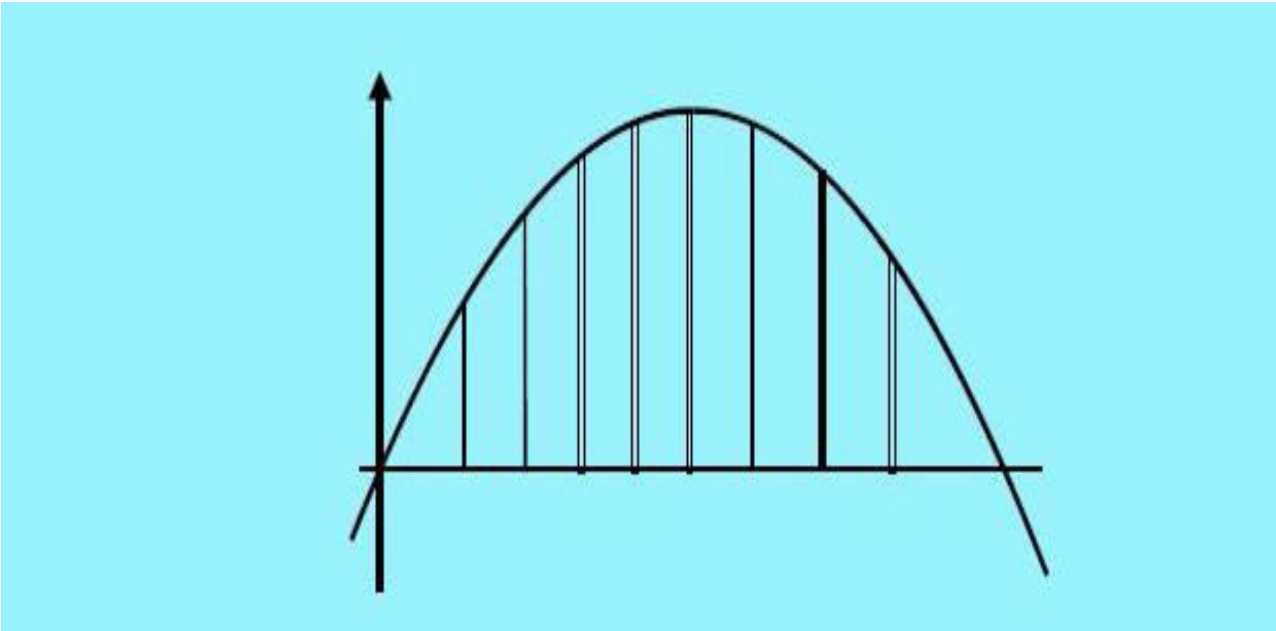
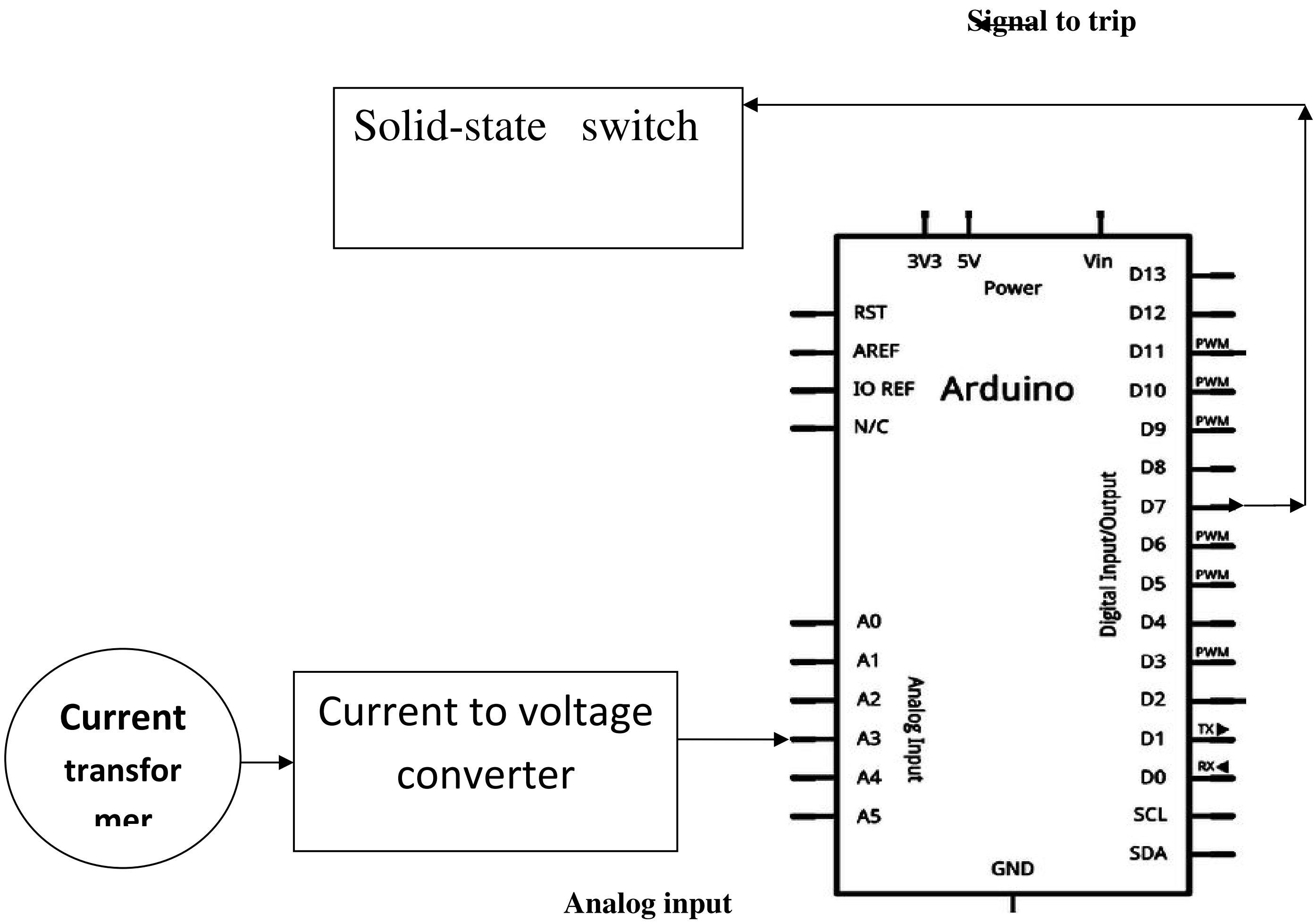


Fig 2.5**:** Sampling of the sine wave positive cycle

It is now easier and faster when this process is done through micro processor, like now at the point when the mean estimation of two back to back half-waves is figured, the DC-segment is eliminated just about 100% with no requirement for non-linear air hole Transformers or comparable segments. Then again, the count of the mean quality devours high value, which is not needed for high short out Current levels where an instantaneous trip is called for. For this case another trip criteria is basically included. On the off chance that the Current in the first half-wave surpasses double the Setting, it is clear that the mean estimation of the two half-waves will surpass the set level and along these lines a trek can be completed quickly without the need to sit tight for the following half-wave All deliberate numerical qualities can obviously effortlessly be exchanged over the serial communication, be put away in memory banks, for later recovery when flaw reasons are being examined. The Sampling is likewise utilized as a part of another great approach to minimize the transient over-reach. At the point when the working time for a stage has passed and the trip order is to be completed, the stage will sit tight for still one single example surpassing the set level before the excursion is connected to the yield relay. In this activated state, the transfer will sit tight for a brief timeframe and if no further Samples are distinguished, the Relay will reset. This implies the Retardation time or the Transient over-shoot is very short.

## 2.3 BLOCK DIAGRAM OF NUMERICAL OVERCURRENT RELAY



**Current transformer:** Current transformer gives output at its secondary terminal from the load side.

**Current to voltage converter:** Convert current input to analog voltage input to microprocessor.

**Microprocessor:** take input analog voltage does sampling, calculate Root Mean square value and compare with given algorithm if condition is fulfilled then output signal to trip (Breaker or solid state switch)

## 3. Development of OC Relay and Test Result

Relay is used to generate trip signals by using the information achieved from monitoring devices (CT, PT), and give instructions to open a circuit under faulty conditions or to give alarms when the equipment is being protected, is approaching towards possible destruction in the transmission line. Relay passes trip signal to the circuit breaker and then the circuit breaker breaks the circuit and isolate faulty equipment.

There are many kinds of relay.

1. over current relay.
2. Distance relay.
3. Differential relay.
4. Pilot relay.

This project demonstrates the over current relay. Over current relay takes input from the monitoring devices as current transformer and potential transformer. This relay monitors current and has inverse characteristics with respect to the currents being monitored. This project includes IDMT (Inverse definite minimum time) and DTL (Definite time lag) type numerical over current relay. The components of this project are given bellow.

1. Arduino Uno.
2. Current Transformer (YHDC TA1309-100)
3. Solid State Relay (G3NA-240B)
4. Resistor (R)
5. Potentiometer (10 k-ohm)

The Above Instruments has been described below:

1. Arduino Uno: Arduino is an open-source platform. It consists of both a physical programmable circuit board often referred to as a microcontroller and a piece of software or IDE (Integrated Development Environment) that runs on our computer. This software is used to write and upload computer code to the physical board. Arduino does not need a separate piece of hardware (also known as program burner) in order to load new code in to the board rather it uses a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Arduino uno has fourteen digital input/output pins; six of them can be used as PWM outputs, six for analog inputs, a USB connection, a power jack, a reset button and more. It can be simply connected to a computer with a USB cable or power it with a AC-to-DC adapter or battery. A different type of arduino uses different IC. But all of them are from ATmega line of IC's from the ATMEL Company. The IC of arduino uno is ATmega 328p-pu.

2. Current transformer (YHDC TA1309-100): The rated input of this current transformer is 0-5 Ampere. Maximum detection input is 8Ampere .rated output is 0-5 milli-ampere. The turn ratio of this CT is 1:1000. It means the output current is 1000 times less than the input current. Operating frequency is 20 Hz - 20 kHz and operating temperature is -35 to 85 degree Celsius.

3. Solid State Relay (G3NA-240B): A solid state relay is used to control high-current AC loads from lower voltage DC control circuitry. Solid state relays have several advantages over mechanical relays. One such advantage is that they can be switched by a much lower voltage and at a much lower current than most mechanical relays. Also, because there's no moving contacts, solid state relays can be switched much faster and for much longer periods without wearing out. This particular solid state relay can switch current loads of up to 40A at 24 to 240 VAC .The rated input voltage is 5 to 24 VDC.

Along with these three components we used a resistor of 470 ohm and two 10 Kilo-ohm potentiometer.

## 3.1 Design and Construction procedure of IMDT (Inverse Definite Minimum Time) and DTL (Definite Time Lag) type numerical Over-Current Relay is discussed below.

### IDMT Type:

The IDMT type numerical over current relay is been discussed here. In the IDMT type, the relay first calculates the line current by sampling the corresponding voltage across the resistor. This resistor is connected with two output terminals of current transformer. This current transformer reduces the current 1000 times of the original line current. As a result the voltage across the resistor becomes less than 5 volts. After calculating the line current, the relay checks whether the line current is greater than the set current or not. This relay does the checking process three times just to ensure whether it is a permanent fault current or just an inrush current. If the over current is due to a permanent fault then the relay will issue a trip signal and the solid state relay which works as a circuit breaker in our project will disconnect the load from the power supply. The elaborate procedure of the IDMT relay is given below.

In this project the inversity of the IDMT relay is changeable. The values of alpha and beta set the inversity in four different types. The table given bellow represents the values of alpha and beta for different types of inversity.

Table3.1: Degree of Inversity

| alpha | beta | Inversity |
|---|---|---|
| 0.02 | 0.14 | Normal inverse |
| 1 | 13.50 | Very inverse |
| 2 | 80 | Extremely inverse |
| 1 | 120 | Long time inverse |

Now in the coding part of the Arduino, all the integers are assigned and the values of alpha and beta is taken correspondingly from the table into two different array alpha[4] and beta[4]. In the code, the two array becomes double alpha[4] = {0.02, 1, 2 ,1} and double beta[4] = {0.14, 13.5, 80, 120}. Now a array name arrayval is created and the length of this array is 100. This array will be used to store the sample values. After that the pin7 of the arduino which is the relay pin becomes high. This relay pin is the input signal of the solid state relay. Another input pin of the solid state relay is connected with the ground of the arduino. In our project this solid state relay works as a circuit breaker. When pin7 of arduino is high the solid state relay will have a 5 volt signal as input as a result the two output terminals which are connected in series with the main supply line will be connected with each other. Thus the load has been connected with the power supply. After the load gets connected, a delay of 3000ms is introduced to avoid inrush current. Since inrush current sustains for one or two seconds from the instant of connecting the load to the power supply .Thus by delaying 3 seconds, the relay avoids the inrush current. Without this delay, the relay will count an over current at the instant of the connection of the loads to the power supply as a result it will issue a trip signal. After the three seconds delay the value of a global variable "emergency" is set to zero. Now using the instruction "analogRead" the relay takes reading from the A3 pin of arduino and stores the value in a integer called val. The A3 pin is connected with a AC voltage wave. This AC voltage wave is the output of the resistor and the resistor gets this voltage as the output of the Current Transformer. The two terminals of the current transformer is connected with the resistor. This current transformer reduces the current 1000 times of the original line current. As a result the current passing through the resistor is becoming few milliampere and the voltage becomes less than 5 volts as the resistance of the resistor is also low. Thus it becomes arduino compatible. Now to read the set current again the instruction "analogRead" is used. The value of the set current is the analogRead output multiplies with 0.009775. The set current pin is A1 and this instruction lets the arduino to read the analog value from pin A1 and multiplies it with a value 0.009775.We used a 10 kilo ohm potentiometer to change the set current. This feature makes our project much more user friendly. The middle pin of the potentiometer is connected to the A1 pin of the Arduino. And the input pin of the potentiometer is connected with 5 volt and ground. Thus By moving the potentiometer knob twisting, we can change the output current of the potentiometer as well as the set current value. The value 0.00975 comes from the digital to analog conversion (bit to ampere conversion). We took 10 A as the highest output of the Current Transformer (actual maximum line current = 10x1000) then in digital range 1023 bits is 10 A and 1 bit is 10/1023=0.009775 Amp. So multiplying the analogRead value with 0.009775 gives the set current value in ampere. Now to read the inversity level again the instruction "analogRead" is used. The set inversity pin is A2 and this instruction lets the arduino to read the analog value from pin A2. We divide the inversity level into three stages. The inversity levels are changeable. To change the inversity, we need to twist the potentiometer knob just like the set current potentiometer. A new 10 kilo ohm potentiometer is used to control inversity.Using the potentiometer, we divide the arduino'sanalogRead into three segments. We know highest range of arduino's bit range is 1023 and lowest is 0. So we select inversity as 3, when the inversity pin's value is greater than 767. When the inversity pin's value is greater than 511 then inversity = 2 and when inversity pin's value is greater than 255 then inversely is 1 other ways inversity is = 0. Now we set a condition to detect the Zero crossing of an AC voltage wave. In this case the used Instruction is also analogRead. Now this relay will read the value From the A3 pin only when the AC voltage crosses zero value. We know the representation of the highest voltage peak is 1023 bits. In this case we put a condition that allows the arduino to start taking values only when the read values from A3 pin are greater or equal to 2 bit and also less than 10 bits. We could take any other lower value than 10 bits instead of 10 bits on that condition but then sometimes the microcontroller will miss zero crossing level thus it can miss one or few half waves. As a result the RMS calculation will be slower. So if we take 5 instead of 10 on that condition then the Zero crossing detection will be much precise but much slower. So we take 10 instead of 5. it gives us a better accuracy as well as fast calculation of RMS. Thus zero crossing point is ensured. This relay will take maximum 80 samples per half cycle. But if Zero crossing is not ensured and the sample rate becomes greater than 80 then the whole process will start from the beginning. After ensuring the zero crossing point and maximum 80 samples per half cycle, the relay go to the previously declared array "arrval" the length of this array is 100 and the relay will put all the zero crossing sample values in the array. Thus that array becomes arrayval[N]. The highest length of that array is 100 but maximum 80 sample values are allowed to take the vacant position of that array. After that the relay checks whether the sample rate is greater than 10 or not. If sample rate N is greater than 10 then all the array value of arrval[N] will be converted into 0 to 5v range from 0 to 1024 bit range then all the values of arrayval is squared and added. The result has been assigned as the value of the sum integer. Then the relay calculates RMS using its formula:

SQRT (sum/ N). Here N is the sampling rate

Now to find out the value of current we did a regression analysis using MATLAB. We noted some main line current values using an ammeter for different loads and we also noted its corresponding RMS voltages across the resistor. Those noted ammeter values of line current are [0.05, 0.4748, 0.7986, 1.607, 3.533] and the corresponding RMS voltage values across the resistor are [0, 0.25, 0.5, 1, 2,]. Then in MATLAB, we plotted RMS vs Ammeter current. Using quadratic line solution we found an equation. This equation gives the value of actual line current for each RMS voltages across the resistor.

After the regression analysis the equation of the quadratic line becomes:

Y= P1*x^2+P2*x+P3

Here, P1=0.19907

P2=1.3293

P3=0.07813

X=RMS

Y=current

Current = 0.19907*RMS*RMS+1.3293*RMS+0.07813

Thus we can find the actual line current

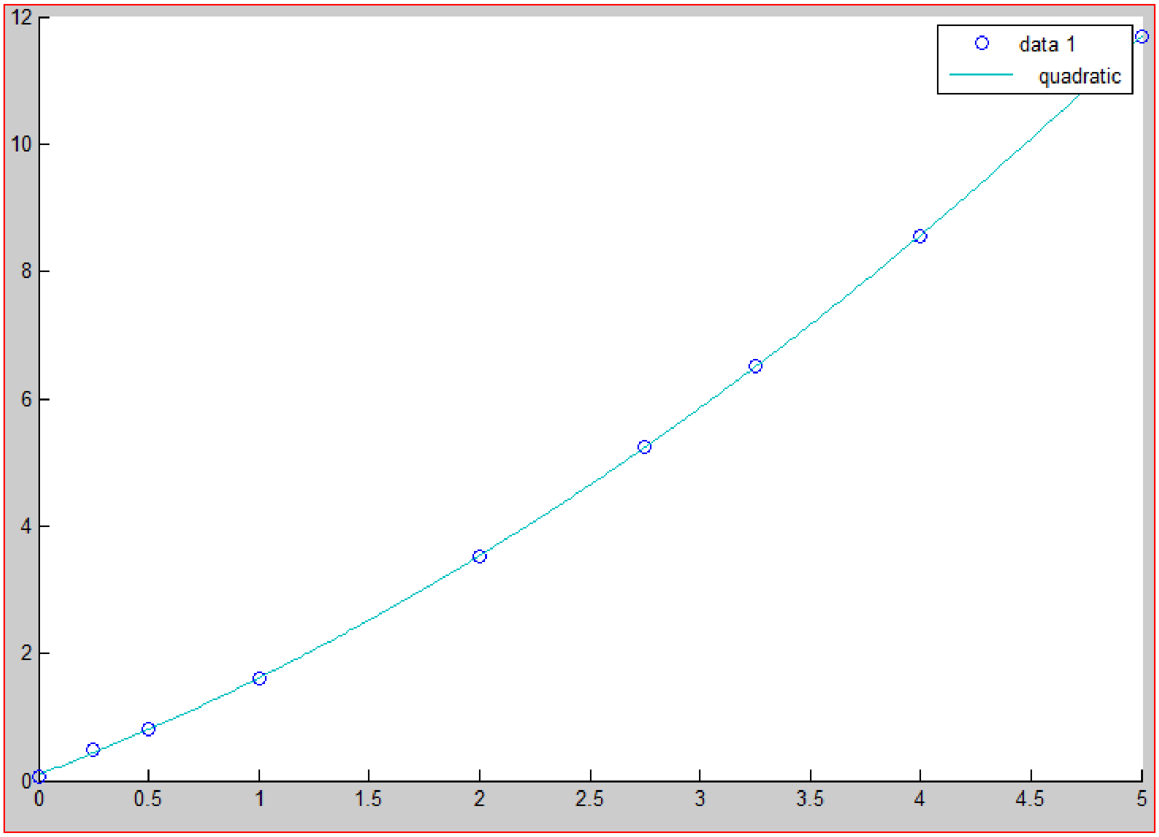



Fig 3.1.1: RMS voltage across Resistor VS Corresponding Line current.

Now the next step is to find out whether the line current is greater than our set current or not. For that we not only checked whether the line current is greater than the set current or not but also checked the value of the integer emergency. The integer "emergency" represents the relay condition. The global value of that integer is 0. But it changes eventually. Now if the line current is greater than the set current and at the first time the emergency is also 0 then the value of the integer emergency will be 1, the sampling rate N will be 0 and then the system will wait for one second. We made the relay to wait for one second because sometimes over current generates due to inrush current or we can say when the load increases but that over current only stays for few cycles. After few cycles the line current goes under the normal condition. So we cannot say this kind of over current is due to any power system fault. So the relay waited one second to detect whether the over current is due to any power system fault or not. After delaying one second the relay will start from the beginning and it will count the RMS voltage value as well as the line current value again. After doing that the relay will again come to next checking stage As this time the emergency is 1. So now it will check whether the line current is greater than set current or not and it will also check whether the emergency is 1 or not and this time the emergency is 1 as in the previous checking stage the emergency became 1. Now If the line current is not greater than set current then the emergency will be 0 and the sampling rate N will also be 0. As a result the relay will start counting the RMS voltage and line current again from the bingeing. But if the line current is greater than the set current and emergency is 1 then the emergency will be 2 and the relay will calculate the value of operating time (Top) using this equation:

Top = K x beta / [(Line current / Set Current) ^ alpha - 1)] here K=1.

In this project development we make k=1. After calculating the operating time (Top) the relay will multiply the Top with 1000 to convert the unit of operating time (Top) From millisecond to second. Then we made the relay to wait for the seconds resulted from the operating time. This relay then again waits for extra one second. This one second extra delay is included here to ensure the fault condition. Now after the operating time and extra one second delay the relay will again start from the beginning. It will again calculate the RMS voltage and line current. At this moment the emergency is 2. Now the relay will again check whether the line current is greater than set current or not .Now if the line current is not greater than the set current then the relay will again start from the beginning. It will again calculate the RMS voltage and line current and this time the emergency will be 0 and sampling rate N will be 0. And these condition results from the new beginning of the whole process. But if the line current is greater than set current then the relay will issue a trip signal. This trip signal is nothing but the signal to make the pin7 of the arduino low. This pin7 is the relay pin and it is the input signal of the solid state relay. Another input pin of the solid state relay is connected with the ground of the arduino. In our project this solid state relay works as a circuit breaker. When pin7 of arduino is low, the solid state relay will have a O volt signal as input as a result the two output terminals which are connected in series with the main supply line will be disconnected from each other. Thus the load is also disconnected from the main supply line. This relay does the checking process three times just to ensure whether it's a permanent fault current or just a inrush current. As it's a microprocessor based numerical relay, it needs a few milliseconds to do the calculations. As it is going to be used for academic purposes. We introduced some extra delay for better understanding of its working principle.

**DTL Type:**

In this project the DTL type relay behaves almost identically to the previously discussed IDMT type relay. The only differences between them lie in their tripping time. In IDMT type  the relay first calculate the line current and then check whether there is any fault or not by comparing the line current and set current. If it detects any fault current, it calculates an operating time for that specific fault current. This operating time depends on the value of the fault current and set current. The operating time also depends on the chosen inversity type. But in case of DTL type there is no operating time calculation. DTL type over current relay first calculates the line current. The line current calculation process is same in both DTL and IDMT type relay. After the line current calculation the DTL type relay checks whether there is any fault or not by comparing the line current and set current. But in this case the set current is given in the relay program by the user. And the user can easily change the value of set current by accessing the main program of the DTL type relay. If the DTL type relay finds any fault it will issue a delay of two seconds. this delay is introduced to avoid the inrush current. Besides this in power system some fault sometimes vanishes automatically. So this delay is introduced to ensure whether is a permanent fault or not. After the delay if still the fault current is present in the system then the relay will pass a trip signal to the solid state relay which acts like a circuit breaker in our project. Then the solid state relay will disconnect the load from main power supply. The elaborate procedure of the DTL relay is given below. Now in the coding part of the Arduino all the integers are assigned and the value of set current is determined. After that a array name arrayval is created and the length of this array is 100.This array will be used to store the sample values. Now the pin7 of the arduino which is the relay pin becomes high. This relay pin is the input signal of the solid state relay. Another input pin of the solid state relay is connected with the ground of the arduino. In our project this solid state relay works as a circuit breaker. When pin7 of arduino is high the solid state relay will have a 5 volt signal as input as a result the two output terminals which are connected in series with the main supply line will be connected with each other. Thus the load gets connected with the power supply. After the load gets connected a delay of 3000ms is introduced to avoid inrush current. Since inrush current sustains for one or two seconds from the instant of connecting the load to the power supply. Thus by delaying 3 seconds the relay avoids the inrush current. Without this delay the relay will count a over current at the instant of the connection of the loads to the power supply and it will issue a trip signal. After the three seconds delay the value of a global variable “emergency” is set to zero. Now using the instruction “analogRead” the relay takes reading from the A3 pin of arduino and stores the value in a integer called val. The A3 pin is connected with a AC voltage wave. This AC voltage wave is the output of the resistor(R) .And the resistor gets this voltage as the output of the Current Transformer .The two terminals of the current transformer is connected with the resistor. This current transformer reduces the current 1000 times of the original line current. As a result the current passing through the resistor is becomes few mili ampere and the voltage becomes less than 5 volts. As the resistance of the resistor is low. Thus it becomes arduino compatible. Now we set a condition to detect the Zero crossing of a AC voltage wave. In this case the used Instruction is also analogRead. Now this relay will read the value From the A3 pin only when the AC voltage crosses zero value. We know the representation of the highest voltage peak is 1023 bits. In this case we put a condition that allow arduino to start taking values only when the read values from A3 pin are greater or equal to 2 bit and also less than 10 bits. We could take any other lower value than 10 bits instead of 10bits on that condition but then sometimes the microcontroller will miss zero crossing level thus it can miss one or few half waves. As a result the RMS calculation will be slower. So if we take 5 instead of 10 on that condition then the Zero crossing detection will be much precise but much slower. So we take 10 instead of 5 .it gives us a better accuracy as well as fast calculation of RMS. Thus zero crossing point is ensured. But if Zero crossing is not ensured and the sample rate becomes greater than 80 then the whole process will start from the beginning. After ensuring the zero crossing point and maximum 80 samples per half cycle, the relay go to the previously declared array “arrval” the length of this array is 100 and the relay will put all the zero crossing sample values in that array. Thus that array becomes arrayval[N]. The highest length of that array is 100 but maximum 80 sample values are allowed to take the vacant position of that array. Now all the array value of arrval[N] will be converted in to 0 to 5v range from 0 to 1024 bit range . Then all the values of arrayval is squared and added then result has been assigned as the value of the sum integer.

Then the relay calculates RMS using its formula:

SQRT sum/ N. Here N is the sampling rate.

Now to find out the value of current we did the same regression analysis as we did in the IDMT type relay. we noted some main line current values using an ammeter for different loads and we also noted its corresponding RMS voltages across the resistor. Those noted ammeter values of line current are [0.05, 0.4748, 0.7986, 1.607, 3.533] and the corresponding RMS voltage values across the resistor are [0, 0.25, 0.5, 1, 2,]. Then in MATLAB we plotted RMS vs Ammeter current. Using quadratic line solution we found an equation. This equation gives the value of actual line current for each RMS voltages across the resistor.

After the regression analysis the equation of the quadratic line becomes

$$Y= P1*x^2+P2*x+P3$$

Here, P1=0.19907

P2=1.3293

P3=0.07813

X=RMS

Y=current

Current = 0.19907*RMS*RMS+1.3293*RMS+0.07813

Thus we can find the actual line current.

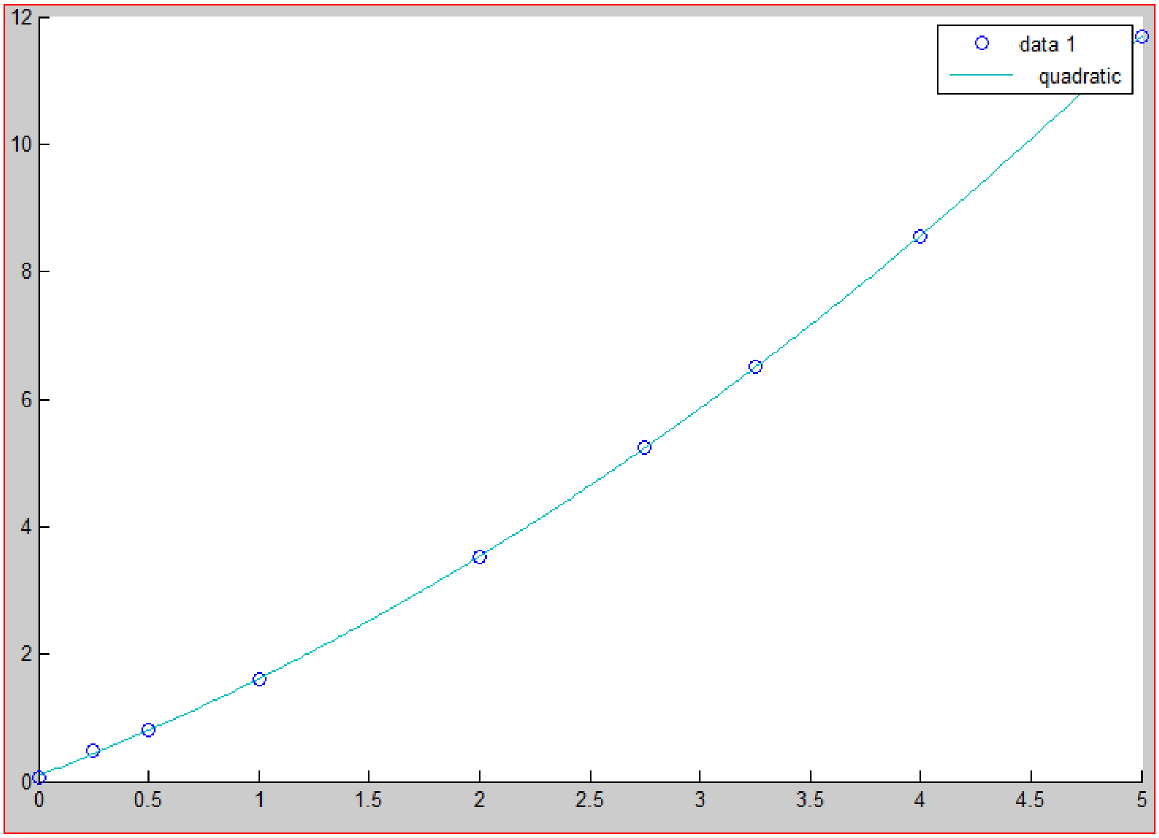
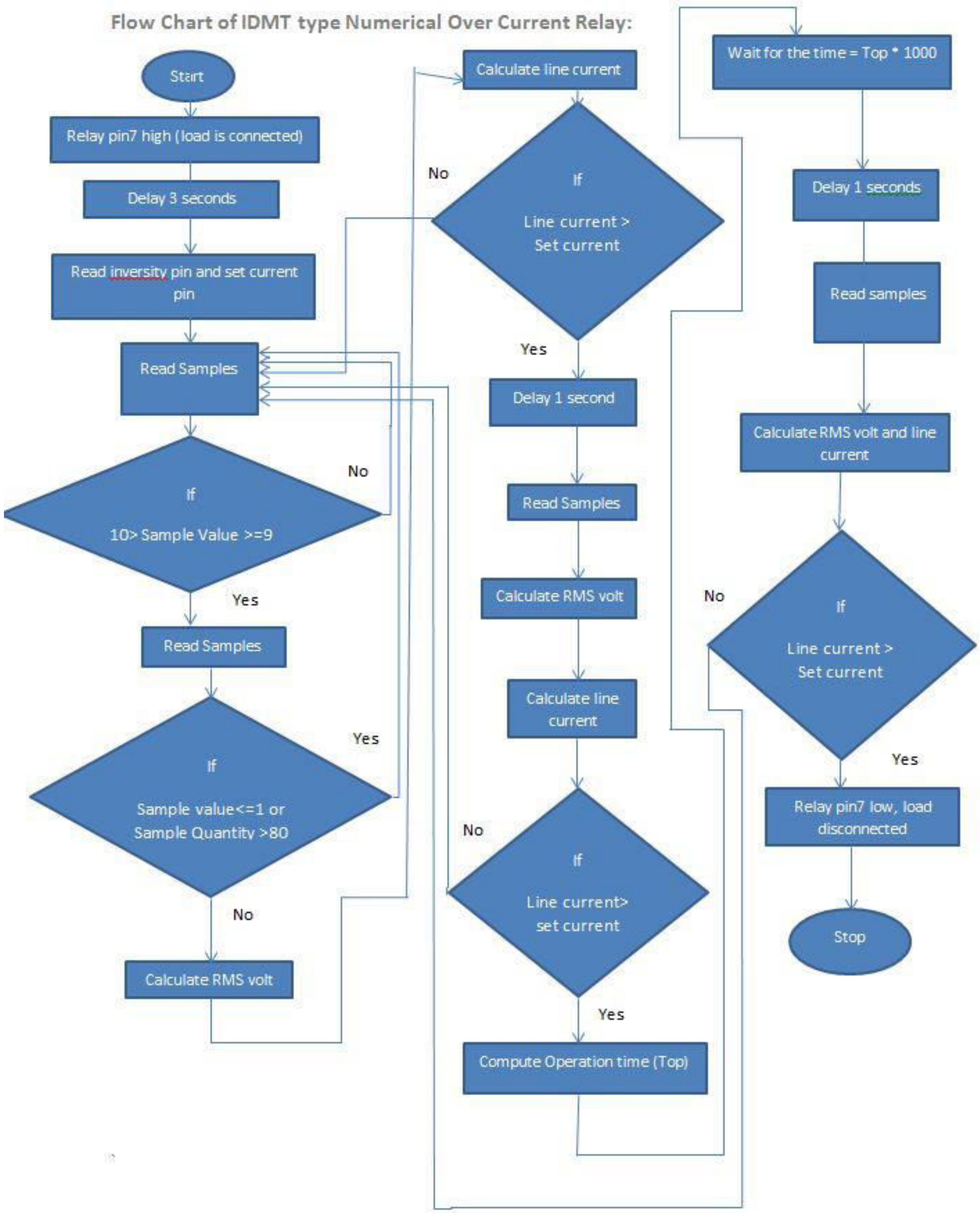


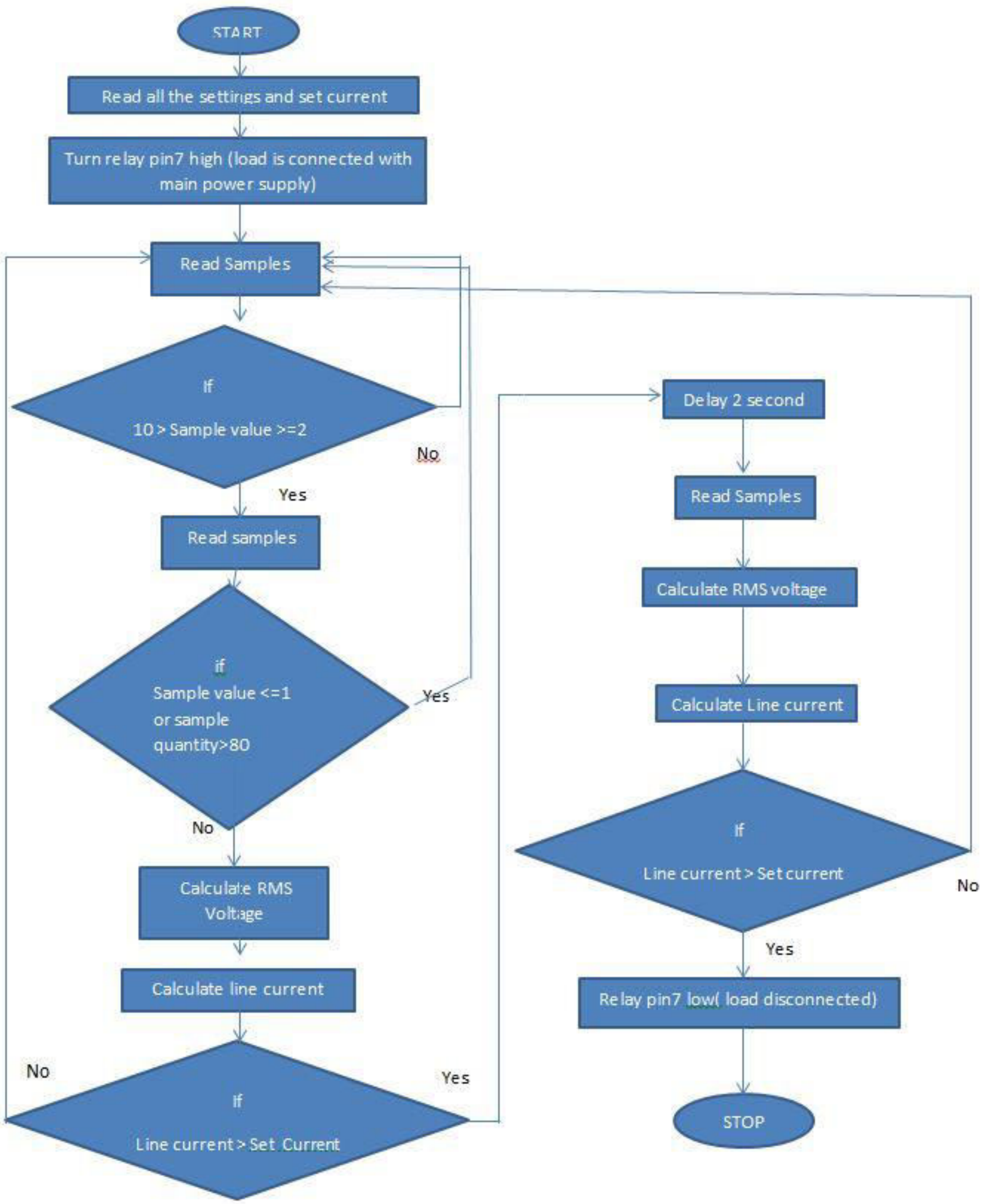

Fig 3.1.2: RMS voltage across Resistor (R) VS. Corresponding Line current

Now the next step is to find out whether the line current is greater than our set current or not. For that we not only checked whether the line current is greater than the set current or not but also checked the value of the integer emergency. The integer "emergency" represents the relay condition. The global value of that integer is 0. But it changes eventually. Now if the line current is greater than the set current and at the first time the emergency is also 0 then the value of the integer emergency will be set to 1. Then the system will wait for two second. We made the relay to wait for two seconds because sometimes over current generates due to inrush current or we can say when the load increases but that over current only stays for few cycles. After few cycles the supply line current goes under the normal condition. So we cannot say this kind of over current is due to any power system fault. So the relay waited two second to detect whether the over current is due to any power system fault or not. After the 2 second time delay the sampling rate N will be 0 and the value of double "sum" will be 0. As a result the relay will start from the beginning and it will count the RMS voltage value as well as the line current value again. After doing that the relay will again come to next checking stage As this time the emergency is 1. Now if the line current is not greater than the set current and at this moment the emergency is also 1 then the relay will allow the load keep connected with supply line and it will again start from the beginning. Relay will again calculate the RMS voltage and line current and this time the emergency will be set to 0. But if the line current is greater than set current then the relay will issue a trip signal. This trip signal is nothing but the signal to make the pin7 of the arduino low. This pin7 is the relay pin and it is the input signal of the solid state relay. Another input pin of the solid state relay is connected with the ground of the arduino. In our project this solid state relay works as a circuit breaker. When pin7 of arduino is low, the solid state relay will have a O volt signal as input as a result the two output terminals which are connected in series with the main supply line will be disconnected from each other. Thus the load is also disconnected from the main supply line. This relay does the checking process two times just to ensure whether it's a permanent fault current or just an inrush current.



Flow Chart of IDMT type Numerical Over Current Relay:
Start
Relay pin7 high (load is connected)
Delay 3 seconds
Read inversity pin and set current pin
Read Samples
No
If
10> Sample Value >=9
Yes
Read Samples
Yes
If
Sample value<=1 or Sample Quantity >80
No
Calculate RMS volt
Calculate line current
No
If
Line current > Set current
Yes
Delay 1 second
Read Samples
Calculate RMS volt
Calculate line current
No
If
Line current> set current
Yes
Compute Operation time (Top)
Wait for the time = Top * 1000
Delay 1 seconds
Read samples
Calculate RMS volt and line current
No
If
Line current > Set current
Yes
Relay pin7 low, load disconnected
Stop

Flow Chart of DTL type Numerical Over Current Relay:



## 3.2 Tabulation of Results:

**IDMT (Inverse Definite Minimum Time) relay:**

**Normal Inverse Situation:**

$\alpha$=0.02, $\beta$=0.14, R=470 Ω.

Top = kβ / [( IL / Iset ) ^ α – 1] (k=1)

Table3.2.1: Normal Inverse Situation of IDMT type relay

| ISET | ILine | CT Output | RMS Volt | Hand Calculation of Top using formulae | Top | Trip Signal |
|---|---|---|---|---|---|---|
| 2.5 A | 2.2 A | 2.2 mA | 1.034 V | No over current | -------- | Low |
| 2.5 A | 3.0 A | 3.0 mA | 1.410 V | 38.323 sec | 37.97 s | High |
| 2.5 A | 3.9 A | 3.9 mA | 1.645 V | 15.671 sec | 15.42 s | High |
| 2.5 A | 4.8 A | 4.8 mA | 1.786 V | 10.661 sec | 10.41 s | High |

**Very inverse Situation:**

α=1, β=13.5, R=470 Ω.

Top = β / [ ( IL / Iset ) ^ α – 1]

Table 3.2.2: Very Inverse Situation of IDMT type relay

| ISET | ILine | CT Output | RMS Volt | Hand Calculation | Top | Trip Signal |
|---|---|---|---|---|---|---|
| 2.5 A | 2.2 A | 2.2 mA | 1.034 V | No over current | --------- | Low |
| 2.5 A | 3.0 A | 3.0 mA | 1.410 V | 67.500 sec | 66.70 s | High |
| 2.5 A | 3.9 A | 3.9 mA | 1.645 V | 24.107 sec | 23.93 s | High |
| 2.5A | 4.8 A | 4.8 mA | 1.786 V | 14.671 sec | 14.20 s | High |

**Extremely Inverse Situation:**

α=2, β=80, R=470 Ω.

Top = β / [ ( IL / Iset ) ^ α – 1]

Table3.2.3: Extremely Inverse Situation of IDMT type relay

| ISET | ILine | CT Output | RMS Volt | Hand Calculation | Top | Trip Signal |
|---|---|---|---|---|---|---|
| 1 A | 2.2 A | 2.2 mA | 1.034 V | 20.833 sec | 20.32 sec | High |
| 1 A | 3.0 A | 3.0 mA | 1.410 V | 10.000 sec | 9.91 sec | High |
| 1 A | 3.9 A | 3.9 mA | 1.645 V | 5.629 sec | 5.19 sec | High |
| 1 A | 4.8 A | 4.8 mA | 1.786 V | 3.629 sec | 3.32 sec | High |

**Long time inverse situation:**

α=1, β=120, R=470 Ω.

Top = β / [ ( IL / Iset ) ^ α – 1]

Table 3.2.4: Long time Inverse Situation of IDMT type relay

| ISET | ILine | CT Output | RMS Volt | Hand Calculation | Top | Trip Signal |
|---|---|---|---|---|---|---|
| 1 A | 2.2 A | 2.2 mA | 1.034 V | 100.00 sec | 99.87 sec | High |
| 1 A | 3.0 A | 3.0 mA | 1.410 V | 60.00 sec | 59.12 sec | High |
| 1 A | 3.9 A | 3.9 mA | 1.645 V | 41.37 sec | 40.87 sec | High |
| 1 A | 4.8 A | 4.8 mA | 1.786 V | 31.58 sec | 31.19 sec | High |

**DTL (Definite lag time)**

Iset = 2.5 sec

Table 3.2.5 :Tabulation of result of DTL type relay

| ILine | CT Output | RMS Volt | Trip Signal |
|---|---|---|---|
| 2.2 A | 2.2 mA | 1.034 V | Low |
| 3.0 A | 3.0 mA | 1.410 V | High |
| 3.9 A | 3.9 mA | 1.645 V | High |

## 3.3 Difference from commercial relay

Our constructed numerical relay is different from other commercial numerical relay. We constructed the project in a simple fashion using lamp board and off the shelf equipment found in the university laboratory. In the actual relay, the delay is calculated by counting electrical cycles which are moreover required for high end use at the real transmission line and in our project we implemented delaying by time delay in millisecond. In our constructed relay, we do not have memory storage of the occurrence of the trip but we have a reset button, which will instantly connect the load with the power supply after every disconnection of the load due to over current In practical relay, there is memory storage which gives the details of the trip. Our project is mainly designed for lab demonstration purpose. Practical relay is expensive and is sometimes difficult for academic use but this project is been implemented using off the shelf and easy to use equipment. In Commercial relay filter and limiter are used for more precise work. In this numerical over current relay we did not use any filter limiter. This portion is left as further work. The reason for this limiting are only to protect the internal circuit from being destroyed .In this project there is only one power supply but in real power system, power is transmitted in ring main system. We also used different power source to run the relay function and this power source is completely independent from main power supply.

# GENERAL CONCLUSION

## 4.1 Conclusion:

The proposed numerical over current relay is designed to perform as like overcurrent relay for Power system. Numerical relaying has turned into a practical different option for the customary relaying frameworks utilizing electromechanical and static relays. Regarding all efforts in production, transmission, and distribution, modern systems cannot afford all humans ‘needs. Therefore, there is a need for substructures and modern networks. In addition, investment should take place to run intelligent networks. Due to increase in electrical energy consumption, load compression, and electricity generator sources as well as the need for quick and safe work of control and productive equipment’s, using digital protective equipment’s especially digital relays is both necessary and essential in protecting power systems. These relays are reliable devices in protecting equipment’s and electrical systems because they are able to early detect errors and prevent their extension. The project was made by using off the shelf thing ( things which are available in the laboratory) for laboratory using purpose and also for increasing student’s skill and understanding such as identify the main components and features of a protection scheme and understand how to implement using this relays, evaluate the relay settings necessary to protect a distribution, transmission or industrial/commercial network, understand the challenges and the influence of modern technology to power system protection .

## 4.2 Further work

Regarding the above mentioned information about intelligent protective relays, the followings are proposed for more researches.

- To modify the relay as directional over current.
- Filter, limiter can be use to protect the internal circuit from being destroyed.
- Nonvolatile memory can be used in further work for memory storing..
- Decrease the impact of CT's saturation during high fault conditions which can affect protection relays operations.

# REFERENCES


1. https://components.omron.com/components/web/pdflib.nsf/0/4025B38ED1A091FC85257201007DD683/$file/G3NA_0911.pdf (14-12-15)
2. http://p.globalsources.com/IMAGES/PDT/SPEC/858/K1040583858.pdf (13-12-15)
3. Switchgear Protection And Power Systems By Sunil S. Rao
4. . L. Hewitson, M. Brown, B. Ramesh, Practical Power Systems Protection, Elsevier
5. Availablehttp://www.gedigitalenergy.com/multilin/manuals12-12-15
6. Science & Technology Books, 2004.
7. J. L. Blackburn and T. J. Domin, Protective Relaying Principles and Applications, 3rd ed., H. L. Willis, Ed., New York: CRC Press, 2007,pp. 63, 175, 340, 453.
8. [2] GE JBC51M Phase Directional Overcurrent Relays, Instruction ManualGEK-49848C, GE Digital Energy Multiline..
9. [3] J. Roberts and A. Guzmán, "Directional Element Design and Evaluation," proceedings of the 21st Annual Western Protective Relay Conference, Spokane, WA, October 1994.
10. [4] B. Fleming, "Negative-Sequence Impedance Directional Element," proceedings of the 10th Annual Doble Pro Test User Group Meeting, Pasadena, CA, February 1998.
11. [5] N. Fischer, D. Haas, and D. Costello, "Analysis of an Auto transformer Restricted Earth Fault Application," proceedings of the 61st Annual Conference for Protective Relay Engineers, College Station, TX, April 2008.
12. [6] K. Zimmerman and J. Mooney, "Comparing Ground Directional Element Performance Using Field Data," proceedings of the 20thAnnual Western Protective Relay Conference, Spokane, WA, October 1993.
13. [7] Gabriel Benmouyaland Standey E. Zocholl , 2002. The Impact of High Fault Current and CT Rating
14. Limits on Overcurrent Protection" Schweizer Engineering Laboratories Inc, 2002.
15. [8] H. Farhangi, 2010. The Path of the Smart Grid IEEE Power& : 18-28 vol8;2010.
16. [9] Ibrahim. M. El-Amin and Nabil H.Al-Abbas, 2006.Saturation of Current Transformers and its Impacts on Digital Overcurrent Relays, IEEE Conference Publication. 4(2) : 199-204, 2013
17. [10] M. Amin and F. Wallenberg, Toward a Smart Grid IEEE Power & Energy pg:34-41 l 2014;11(3)
18. https://www.sparkfun.com/products/10636 (13--12-15)
19. http://electrical-engineering-portal.com/types-and-applications-of-overcurrent-relay-1(14-12-15)
20. file:///C:/Users/rumi/Downloads/REVIEW_OF_MICROPROCESSOR_BASED_PROTECTIV.pdf(14-12-15)
21. http://www.allinterview.com/showanswers/90014/what-is-idmt-relay-how-it-works.html(15-14-15)

# APPENDIX

## IDMTL

```
int setCurrentPin = A1;
int setInversityPin = A2;
// voltage input
int inputVol = A3;
int relay = 7;

// summation of samples
double sum=0;
// store samples
double arrval[100];
// number of samples
int N = 0;

double rms;
double  val;

// Table of inversity
double alpha[4] = {0.02, 1, 2 ,1};
double beta[4] = {0.14, 13.5, 80, 120};

// 0 = normal, 1 = very inverse, 2 = extremely inverse, 3 = long time
inverse
int inversity;

// calculate rms current
double current;

// a temporary variable to indicate the current state of overcurrent
int emergency;

// Initial setup function
void setup(){
  // Begin Serial communication to display in serial monitor
  Serial.begin(9600);

  // Assign zero to arrval
  for(int i=0;i<100;i++){
    arrval[i] = 0;
  }

  pinMode(relay, OUTPUT);
  digitalWrite(relay, HIGH);
  // To avoid inrush current
  delay(3000);
```

```
  emergency = 0;
}

void loop(){
  // Final stage of overcurrent, keep the circuit open
  if(emergency==3){
    N = 0;
    return;
  }

  val = analogRead(inputVol);


  //Get Set Current and measure of inversity in a range of 0 to 10A,
0.009775 = 10/1023
  double setCurrent = analogRead(setCurrentPin)*0.009775;

  int tmp = analogRead(setInversityPin);
  if(tmp>767){
    inversity = 3;
  }else if(tmp>511){
    inversity = 2;
  }else if(tmp>255){
    inversity = 1;
  }else{
    inversity = 0;
  }

  //Take samples
  if(val>=2 && val<6){
    while(true){
      val = analogRead(inputVol);
      if(val<=2 || N>80){
        break;
      }
      arrval[N] = val;
      N++;
    }
  }

  //Calculate rms value of current and Top
  if(N>10){
    sum = 0;
    int i  = 0;
    while(i<N){
      // digital to analog conversion
       val = arrval[i]*5/1024;
       sum = sum + val*val;
       i++;
    }
    sum = sum/N;
```

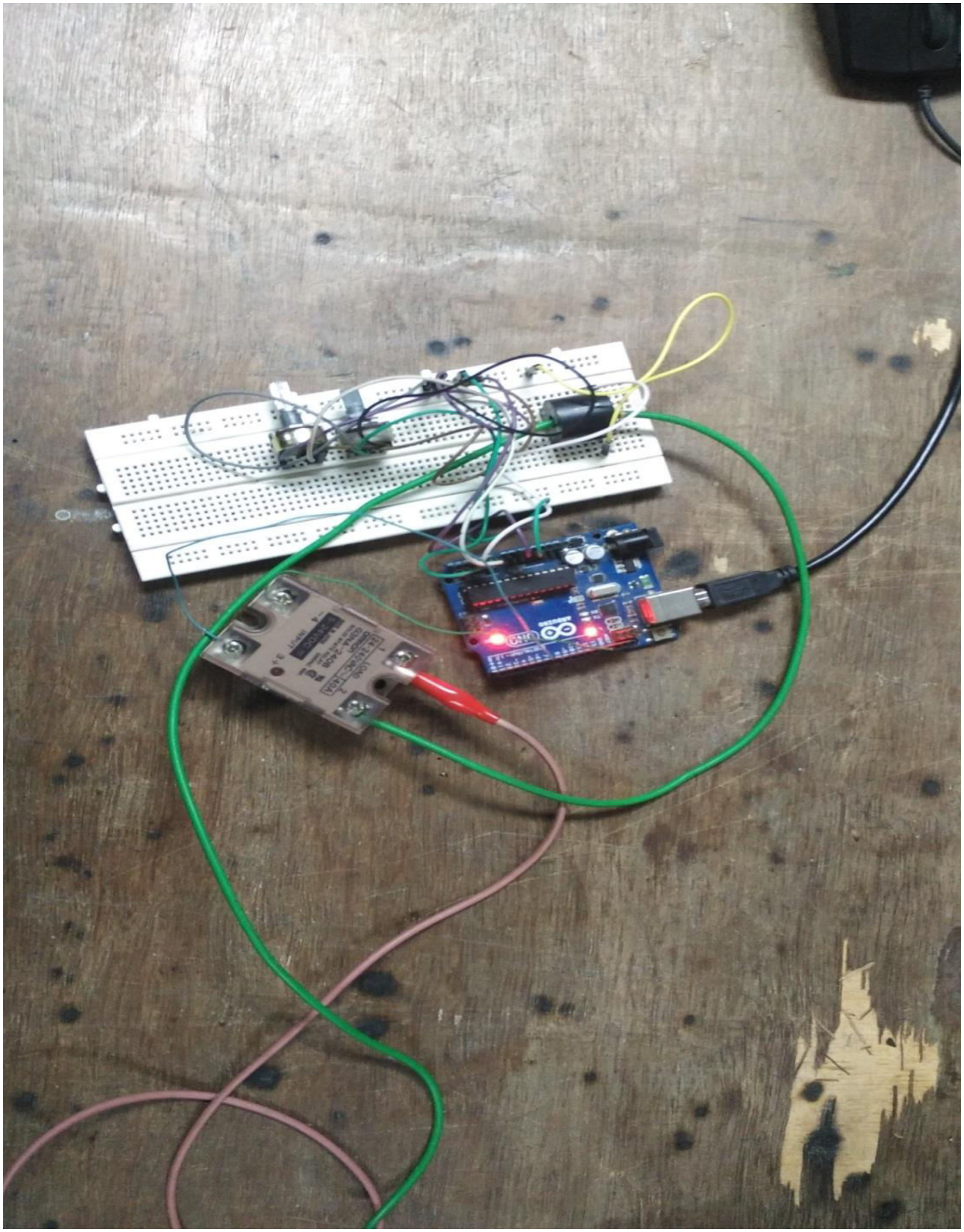
```
    rms = sqrt(sum);
    //Serial.println(rms);
    rms = (rms*100)/100.0;

    // collect data of source current from ameter and the rms value of
the above line
    // 2nd order regression
    current = 0.19907*rms*rms+1.3293*rms-0.0218624+.1;

    Serial.print("N= ");
    Serial.println(N);
    Serial.print("I = ");
    Serial.println(current);
    Serial.print("Set Current: ");
    Serial.println(setCurrent);
    Serial.print("Inversity: ");
    Serial.println(inversity);
    if(current>setCurrent && emergency==0){
      emergency = 1;
      N = 0;
      delay(1000);
      return;
    }
    if(current>setCurrent && emergency==1){
      emergency = 2;
      double Top = beta[inversity]/(pow(current/setCurrent,
alpha[inversity])-1);

      Serial.print("Top: ");
      Serial.println(Top);
      delay(Top*1000);
      delay(1000);
    }else if(current>setCurrent && emergency==2){
      emergency = 3;
      Serial.println("Disconnecting....");
      digitalWrite(relay, LOW);
    }else{
      emergency = 0;
    }
  }
  N = 0;
}
```

DTL

```
int inputVol = A3;
double sum=0;
double arrval[100];
int N = 0;
double rms;
int relay = 7;
double  val;
int c= 0;
void setup(){
  Serial.begin(9600);
  for(int i=0;i<100;i++){
    arrval[i] = 0;
  }
  pinMode(relay, OUTPUT);
  digitalWrite(relay, HIGH);
}

void loop(){
  val = analogRead(inputVol);
  //Serial.print("Analog value: ");
  //Serial.println(val);
  if(val>=2 && val<10){
    while(true){
      val = analogRead(inputVol);
    //  Serial.print("inside: ");
     // Serial.println(val);

      if(val<=1 || N>80){
        break;
      }
      arrval[N] = val;
      N++;
    }

    sum = 0;
    int i  = 0;
    while(i<N){
      val = arrval[i]*5/1024;
        // Serial.print("Volatage: ");
        // Serial.println(val);
      val=val*val;
       //Serial.print("Square: ");
       //Serial.println(val);
      sum+=val;
      //Serial.print("Sum: ");
       //Serial.println(sum);
     i++;
    }
    sum = sum/N;
```

```
  rms = sqrt(sum);
  //Serial.println(rms);
  rms = (rms*100)/100.0;
  rms = 0.19907*rms*rms+1.3293*rms-0.0218624+0.1;
  //Serial.print("c = ");
  //Serial.println(c);
  if(rms>2.5){
    if(c>20){
      digitalWrite(relay,LOW);
      Serial.println("Overcurrent disconnecting..");
      c = 0;
     }
     c++;
   }else if(rms<3){
     digitalWrite(relay,HIGH);
     c=0;
   }
   sum = 0;
  }
 if(N>10){
  Serial.print("N= ");
  Serial.println(N);
  Serial.print("I = ");
  Serial.println(rms);
 }
 N = 0;
}
```

**Over current relay setup**



## Over current relay setup with lamp board (load)