DETECTION OF FAULT LOCATION IN UNDERGROUND CABLE USING ARDUINO
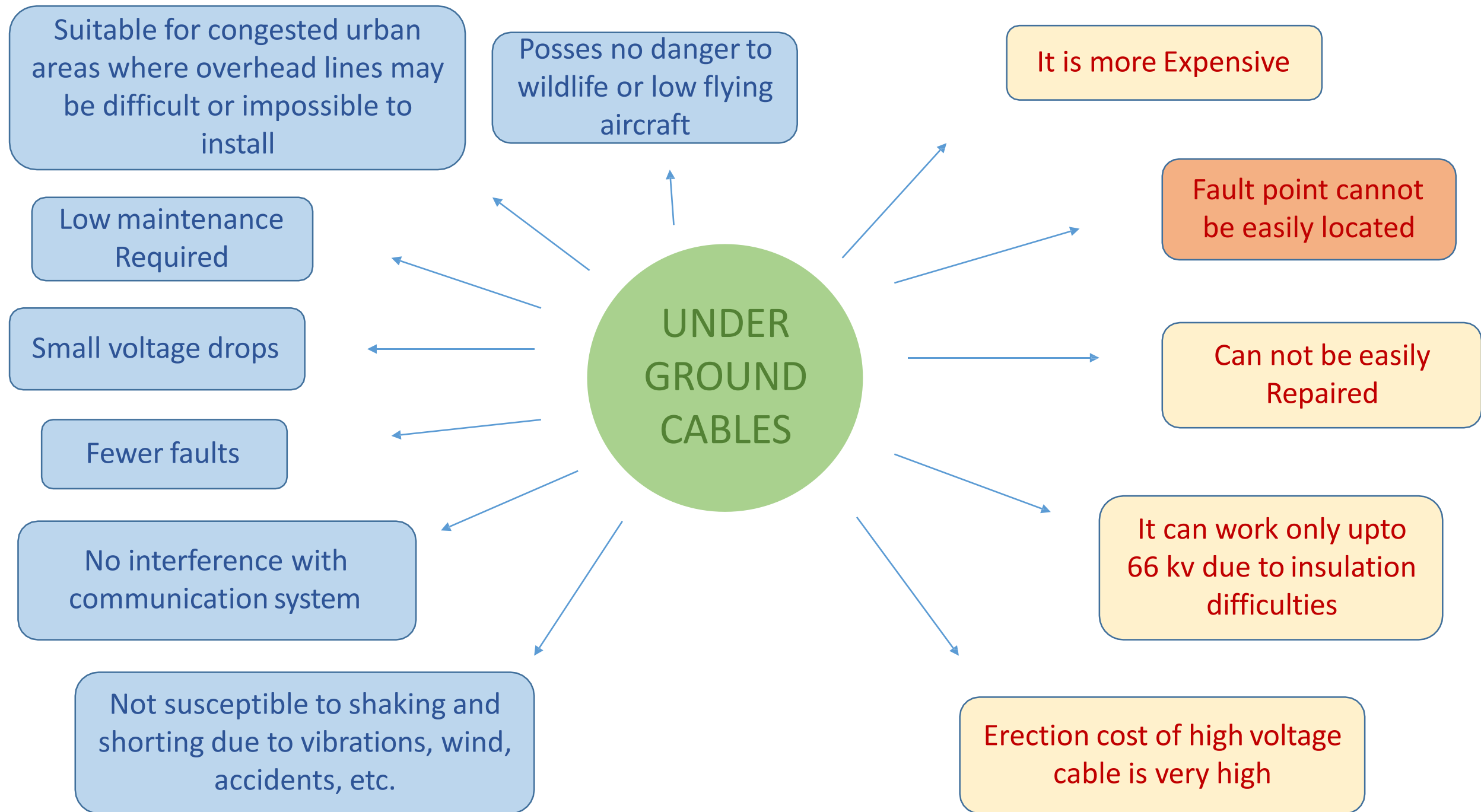
Flow of Presentation

- Introduction to Project

- Underground cable v/s Overhead cables

- Faults in underground cables

- Methods for the Detection of faults

- Introduction to the Circuit

- Working of the Circuit

- Different circuit components

Introduction

- The objective of this project is to determine the distance of underground cable fault from base station in kilometers using an Arduino board.

- Many time faults occur due to construction works and other reasons.

- Cables have some resistance. We are mainly focusing that resistance. Resistance can vary with respect to the length of the cable.

- If the length of the cable is increase, the value of the resistance will also increase.

- If any deviation occurs in the resistance value, we will call that is fault point and finding that place through Arduino technology.

- That fault point represents the standard of distance (kilometre) from the base station.This value displayed by display unit.

- Before attempting to locate underground cable faults on cable, it is necessary to know where the cable is located and what route it takes. If the fault is on secondary cable, knowing the exact route is even more critical.

- Since it is extremely difficult to find a cable fault without knowing where the cable is, it makes sense to master cable locating and tracing and to do a cable trace before beginning the fault locating process.

- Success in locating or tracing the route of electrical cable and metal pipe depends upon knowledge, skill, and perhaps, most of all, experience.

- Although locating can be a complex job, it will very likely become even more complex as more and more underground plant is installed. It is just as important to understand how the equipment works as it is to be thoroughly familiar with the exact equipment being used.

UNDER GROUND CABLES

Suitable for congested urban areas where overhead lines may be difficult or impossible to install

Posses no danger to wildlife or low flying aircraft

It is more Expensive

Low maintenance Required

Fault point cannot be easily located

Small voltage drops

Can not be easily Repaired

Fewer faults

No interference with communication system

It can work only upto 66 kv due to insulation difficulties

Not susceptible to shaking and shorting due to vibrations, wind, accidents, etc.

Erection cost of high voltage cable is very high

Faults in Underground cables

| Open circuit Fault	Short circuit Fault	Earth Fault

Open circuit Fault

- When there is a break in the conductor of a cable, it is called open circuit fault.
- The open-circuit fault can check by a megger.
- The megger will indicate zero resistance in the circuit of the conductor that is not broken.
- However if a conductor is broken the megger will indicate an infinite resistance

Short circuit Fault

- When two conductors of a multi core cable come in critical contact with each other due to insulation failure, it is so called as short circuit fault.
- The two terminals of a megger are connected to any two conductors.
- If the megger gives a zero reading it indicates short-circuit fault between these conductors.

Earth Fault
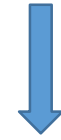
- When the conductor of a cable comes in contact with earth
- To identify this fault, one terminal of the megger is connected to the conductor and the other terminal connected to the earth.
- The megger indicates zero reading; it means the conductor is earthed.

Methods for the Detection of Faults

Online Method

- Online method uses and process the sampled current and voltages to determine the fault points.

Off-Line Method

- This method uses a special instrument to test out service of cable in the field. Offline method is classified into two methods such as tracer method and terminal method.

Terminal Methods

- Terminal method is used to detect the location of the fault in a cable from one end or both the ends without tracking. This method is used to find general areas of the fault to accelerate tracking on buried cable.

Tracer Method

- In this method fault of the cable can be detected by walking on the cable lines. Fault location is denoted from electromagnetic signal or audible signal. This method is used to find the fault location very accurately.

Introduction to Arduino Nano

- **Arduino Nano** is a small, compatible, flexible and breadboard friendly Microcontroller board, developed by Arduino.cc in Italy, based on ATmega328p (Arduino Nano V3.x) / Atmega168 (Arduino Nano V3.x).
- ❖ It comes with exactly the same functionality as in Arduino UNO but quite in small size.
- ❖ It comes with an operating voltage of 5V, however, the input voltage can vary from 7 to 12V.
- **Arduino Nano Pinout** contains 14 digital pins, 8 analog Pins, 2 Reset Pins & 6 Power Pins.
- ➢ Each of these Digital & Analog Pins are assigned with multiple functions but their main function is to be configured as input or output.
- ➢ They are acted as input pins when they are interfaced with sensors, but if you are driving some load then use them as output.
- ➢ Functions like pinMode() and digitalWrite() are used to control the operations of digital pins while analogRead() is used to control analog pins.
- ➢ The analog pins come with a total resolution of 10bits which measure the value from zero to 5V.
- ➢ Arduino Nano comes with a crystal oscillator of frequency 16 MHz. It is used to produce a clock of precise frequency using constant voltage.
- ➢ There is one limitation using Arduino Nano i.e. it doesn't come with DC power jack, means you can not supply external power source through a battery.
- ➢ This board doesn't use standard USB for connection with a computer, instead, it comes with Mini USB support.
- ➢ Tiny size and breadboard friendly nature make this device an ideal choice for most of the applications where a size of the electronic components are of great concern.
- ➢ Flash memory is 16KB or 32KB that all depends on the Atmega board i.e Atmega168 comes with 16KB of flash memory while Atmega328 comes with a flash memory of 32KB. Flash memory is used for storing code. The 2KB of memory out of total flash memory is used for a bootloader.

Arduino Nano Pinout
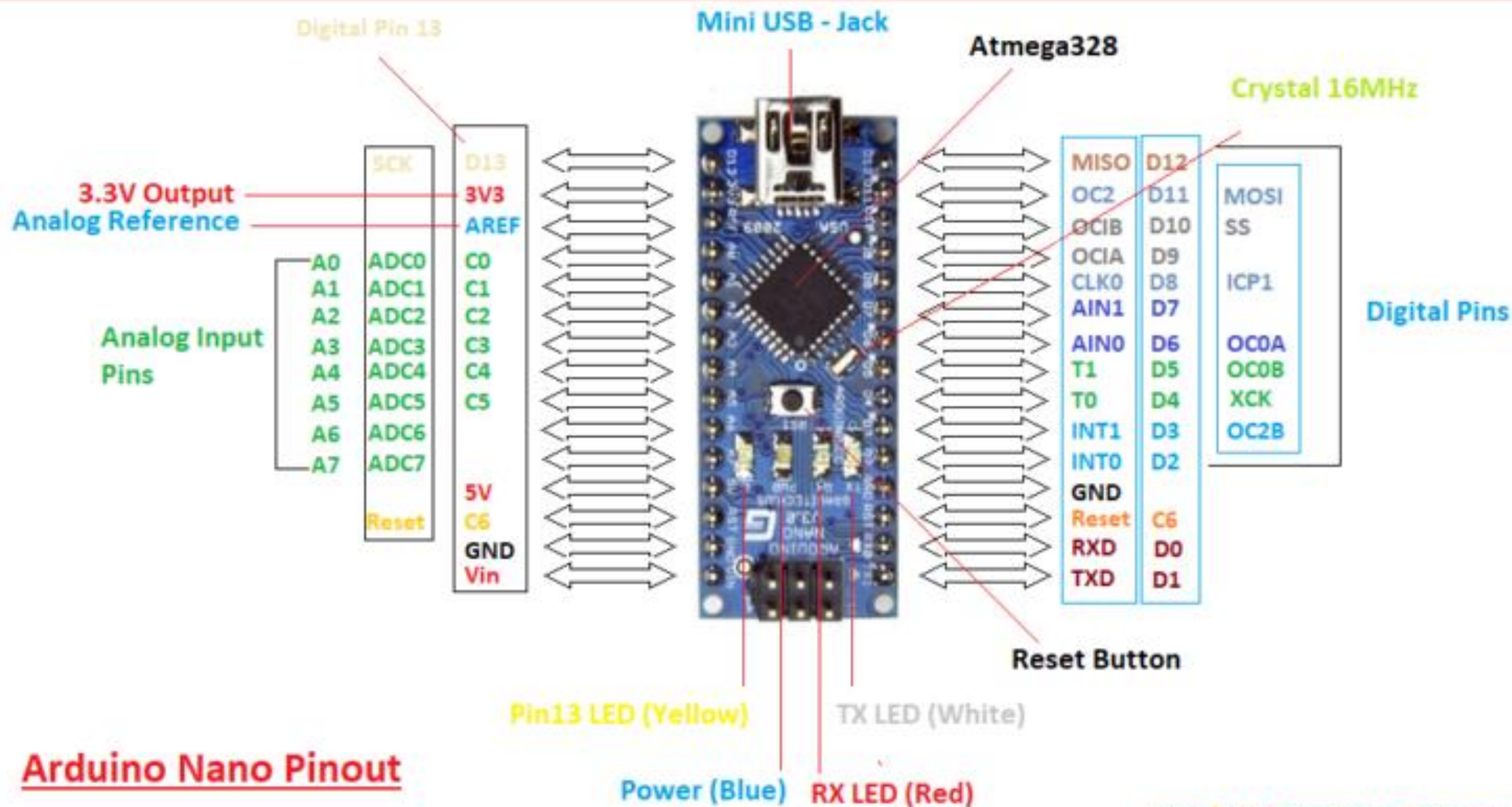
No.	Pin Number	Pin Description
1	D0 – D13	Digital Input / Output Pins.
2	A0 – A7	Analog Input / Output Pins.
3	Pin # 3, 5, 6, 9, 11	Pulse Width Modulation (PWM) Pins.
4	Pin # 0 (RX) , Pin # 1 (TX)	Serial Communication Pins.
5	Pin # 10, 11, 12, 13	SPI Communication Pins.
6	Pin # A4, A5	I2C Communication Pins.
7	Pin # 13	Built-In LED for Testing.
8	D2 & D3	External Interrupt Pins.

Liquid crystal display (LCD)

➢ Most common LCDs connected to the microcontrollers are 16x2 and 20x2 displays.

➢ This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.

➢ The standard is referred to as HD44780U, which refers to the controller chip which receives data from an external source (and communicates directly with the LCD.

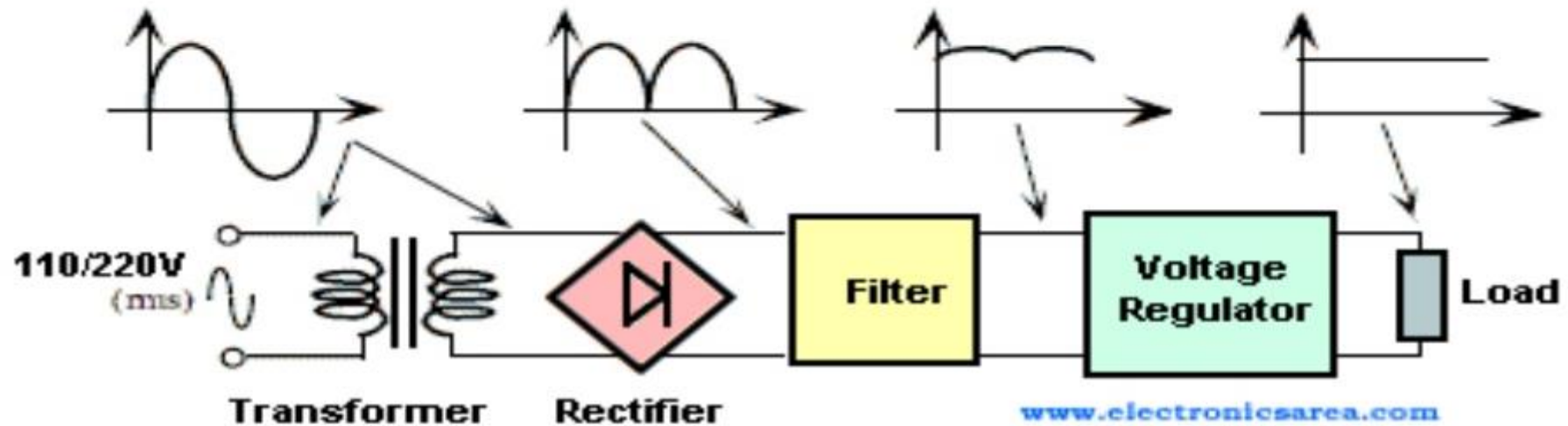| Pin	Symbol	Function
1	Vss	Ground
2	Vdd	Supply Voltage
3	Vo	Contrast Setting
4	RS	Register Select
5	R/W	Read/Write Select
6	En	Chip Enable Signal
7-14	DB0-DB7	Data Lines
15	A/Vee	Gnd for the backlight
16	K	Vcc for backlight

Introduction to circuit Diagram

- While a fault occurs for some reason, at that time the repairing process related to that particular cable is difficult due to not knowing the exact location of the cable fault. The proposed system is to find the exact location of the fault.

- This project uses Ohms Law concept, when a low voltage DC is applied to the feeder end through a series resistor, then the current would differ based on the location of fault occurred in the cable.

- In case is there any short circuit occurred from line to ground, then the voltage across series resistor alters accordingly, then it is fed to an analog to digital converter to develop exact data, which the pre programmed Arduino module will display in kilometres.

- The proposed system is designed with a set of resistors to signifying the length of a cable in kilometers, and the fault creation is designed with a set of switches at every known kilometer (KM) to cross check the exactness of the same. The fault happening at a specific distance and the particular phase is displayed on an LCD interfaced to the 8051 microcontroller.
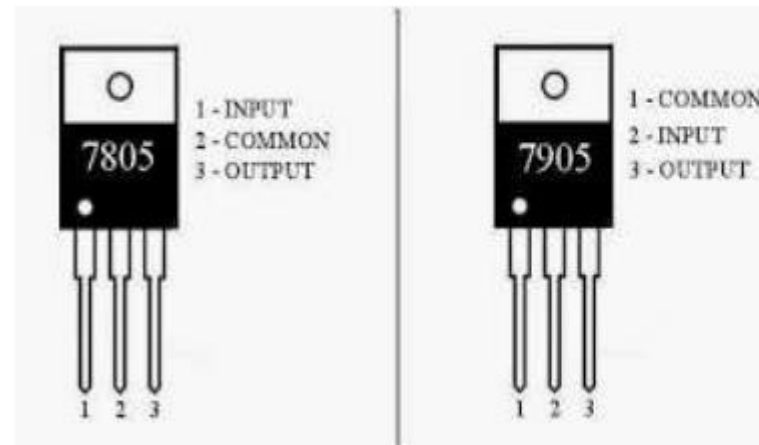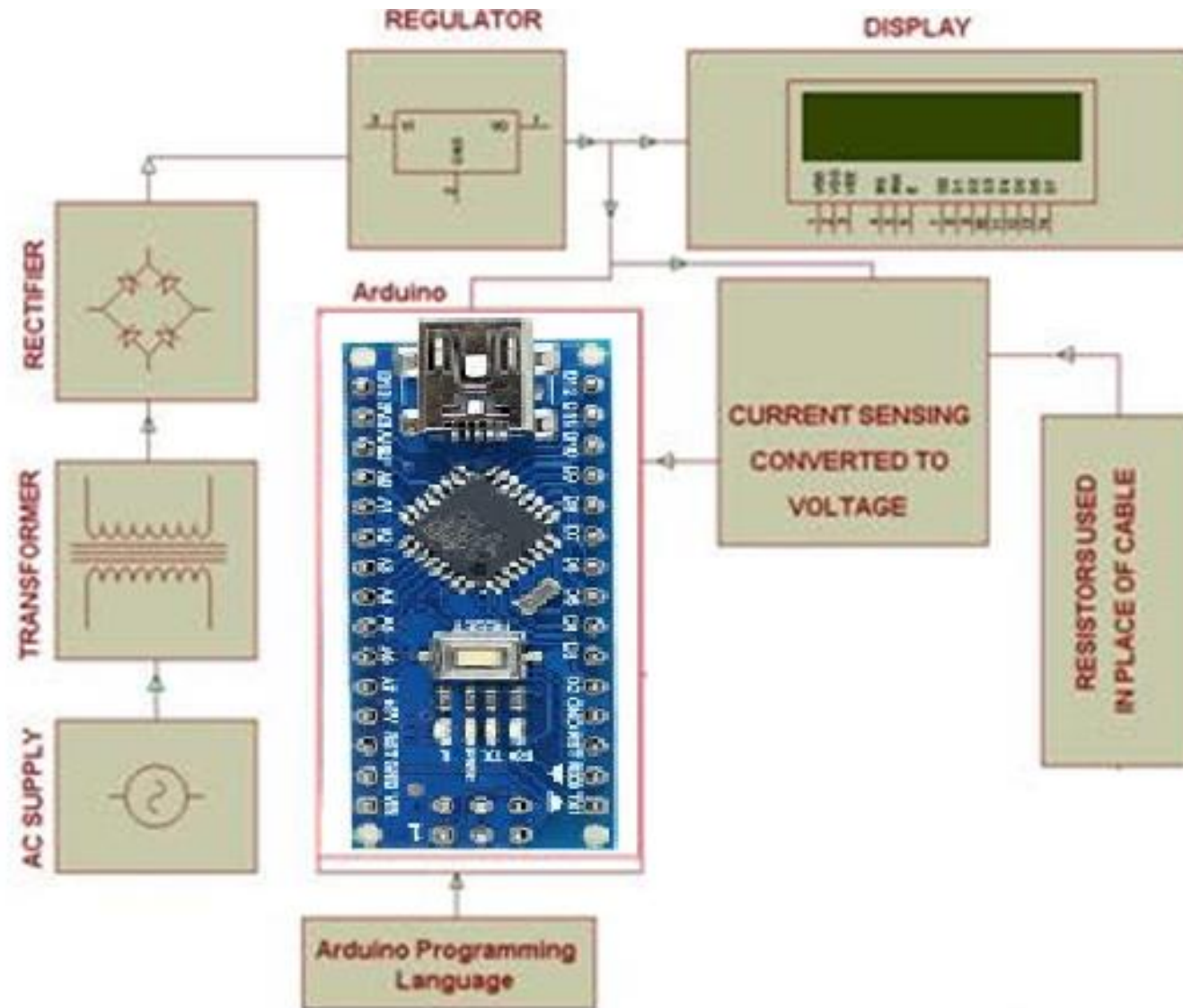
Block Diagram of Circuit
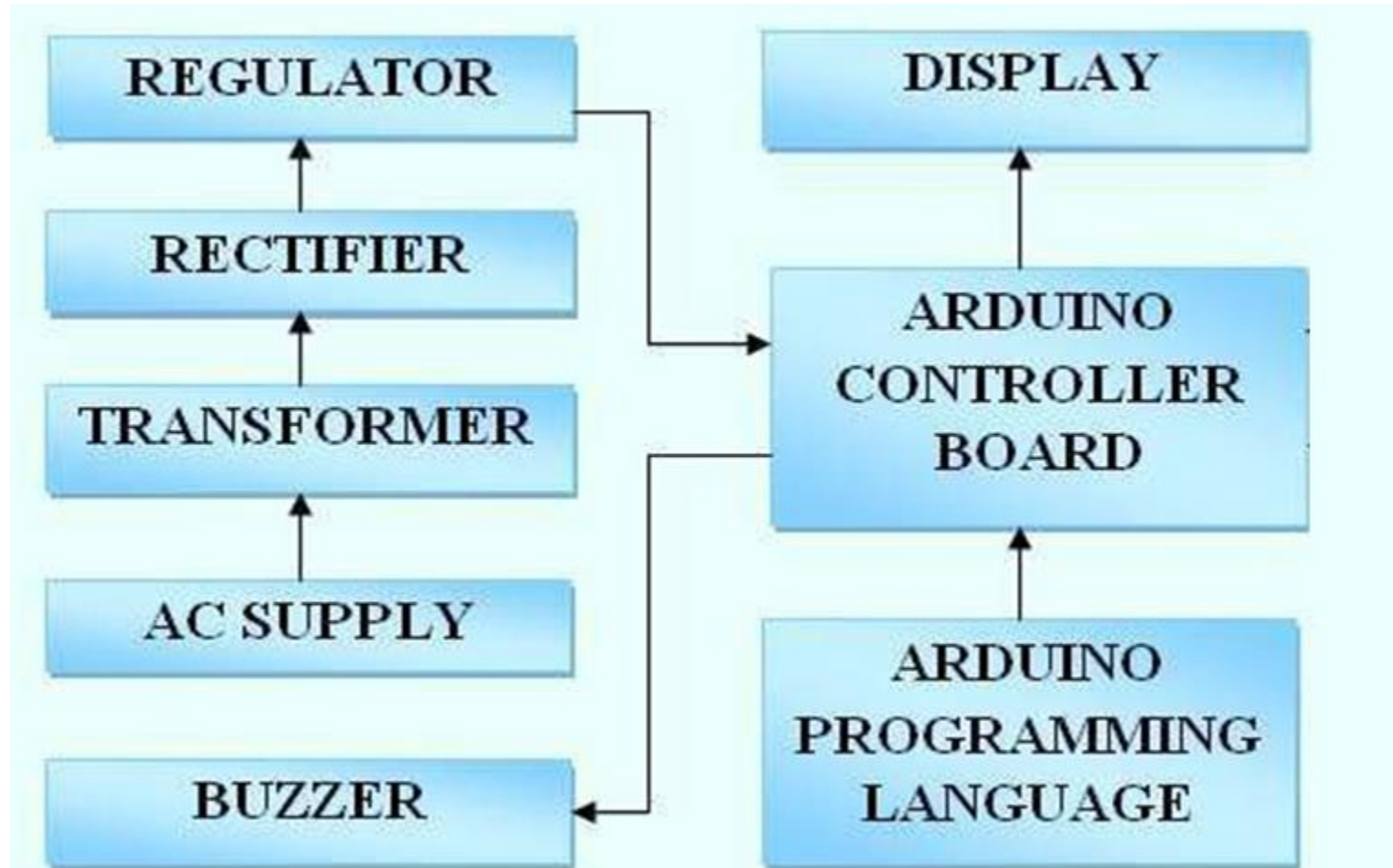
Power supply

All digital circuits require regulated power supply

Picture 1. Regulated power source Block Diagram

		1 - INPUT		1 - COMMON
7805		2 - COMMON	7905	2 - INPUT
		3 - OUTPUT		3 - OUTPUT

BLOCK DIAGRAM

Circuit Diagram

16x2 LCD

Brightness control

NANO

Pin 11

Pin 13

KNOWN RESISTOR

R1
3.3K

UNKNOWN RESISTOR

R R R R

R R R R

UNDERGROUND CABLE
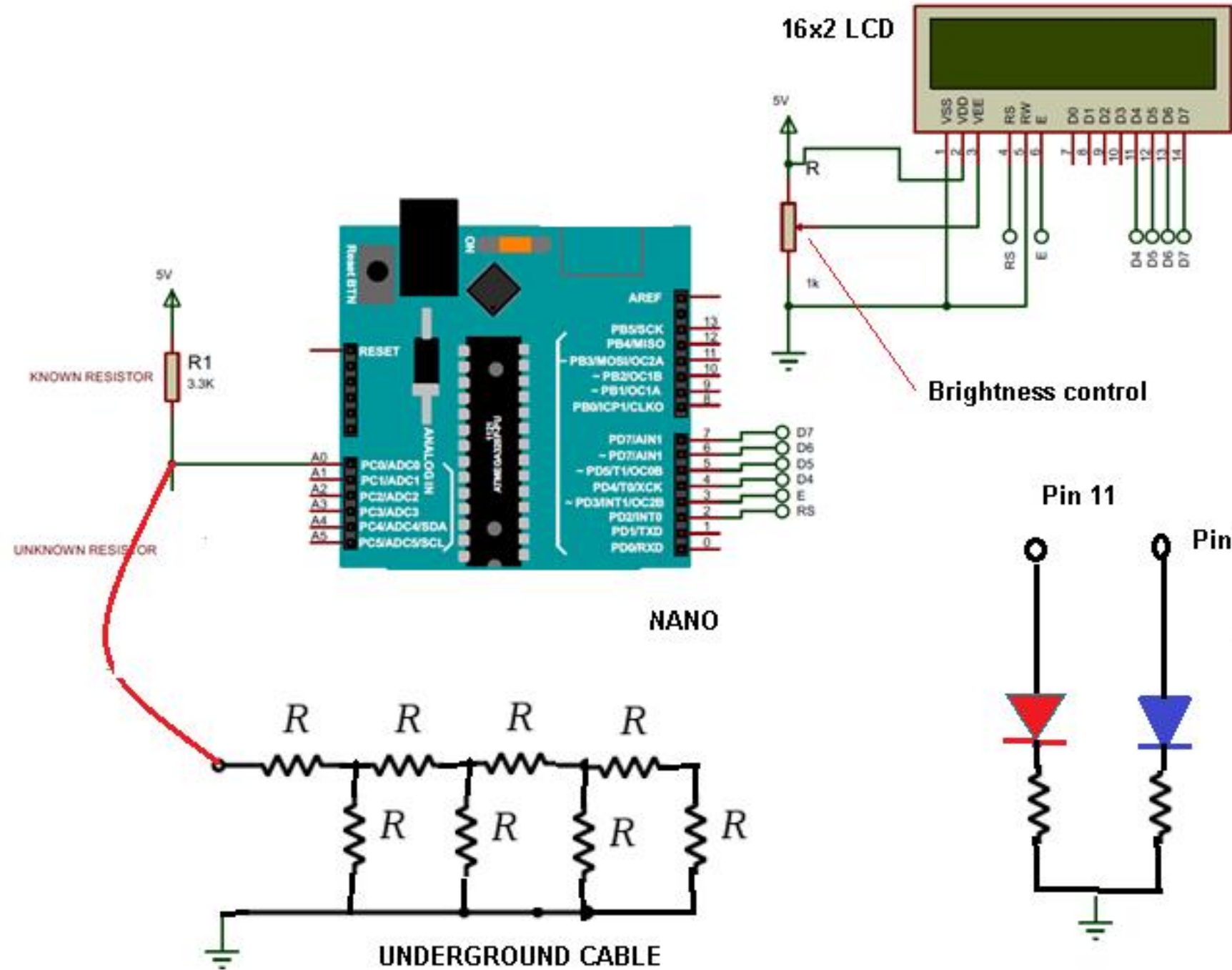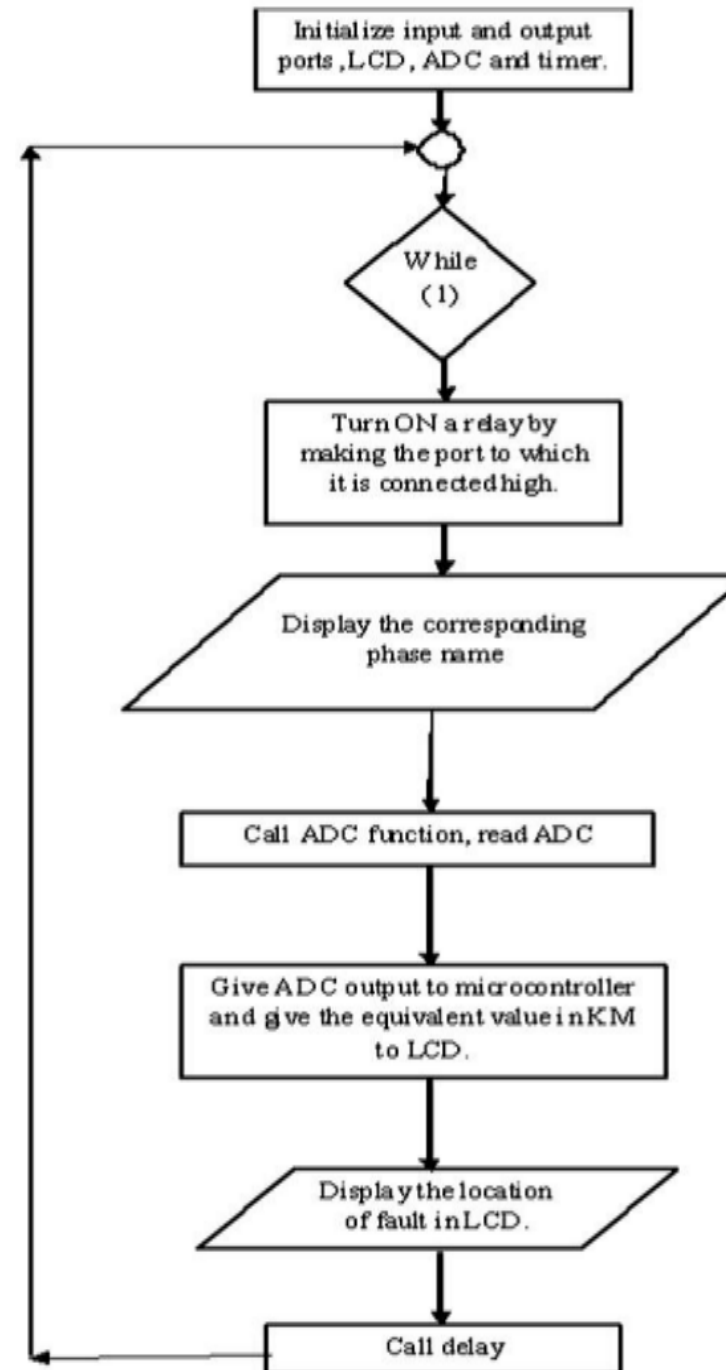
Working of Circuit

- The project is assembled with a set of resistors representing cable length in KM's. The fault occurring at a particular distance and the respective phase is displayed on a LCD interfaced to the Arduino board.

- This project uses a standard concept of Ohms law, i.e., when a low DC voltage is applied at the feeder end through series resistor (assuming them as cable lines), then the current would vary depending upon the location of the fault in the cable.

- In case of a short circuit (line to ground), the voltage across the series resistors changes which is then fed to an ADC, to develop a precise digital data that gets displayed on the LCD.

- Fault creation is made by a set of switches at every known KM to cross check the accuracy of the same.

- The fault occurring at a particular distance and the respective phase is displayed on a LCD interfaced to the Arduino board.

```
┌─────────────────────────┐
│ Initialize input and    │
│ output ports, LCD, ADC  │
│ and timer.              │
└─────────────────────────┘
           │
           ▼
         ◇◇◇
           │
           ▼
        ╱◇◇◇╲
       ◇ While ◇
        ╲ (1) ╱
         ◇◇◇
           │
           ▼
┌─────────────────────────┐
│ Turn ON a relay by      │
│ making the port to which│
│ it is connected high.   │
└─────────────────────────┘
           │
           ▼
  ╱────────────────────╲
 ╱ Display the          ╲
╱  corresponding         ╲
╲  phase name            ╱
 ╲──────────────────────╱
           │
           ▼
┌─────────────────────────┐
│ Call ADC function,      │
│ read ADC                │
└─────────────────────────┘
           │
           ▼
┌─────────────────────────┐
│ Give ADC output to      │
│ microcontroller and give│
│ the equivalent value in │
│ KM to LCD.              │
└─────────────────────────┘
           │
           ▼
  ╱────────────────────╲
 ╱ Display the location  ╲
╱  of fault in LCD.       ╲
 ╲──────────────────────╱
           │
           ▼
┌─────────────────────────┐
│ Call delay              │
└─────────────────────────┘
```

FLOW CHART

PROGRAM CODE

```cpp
#include<LiquidCrystal.h>

LiquidCrystal lcd(7,6,5,4,3,2);   //rs,e,d4,d5,d6,d7

int Vin=5;       //voltage at 5V pin of arduino
float Vout=0;    //voltage at A0 pin of arduino
int R1=10000;   //value of known resistance
int Rx=0;       //value of unknown resistance
int a2d_data=0;
float buffer=0;

float Rc = 0.01;   //Cable Resistance per meter its 0.01
Ohm/MtrPOHEDWWWERIP
void setup()
{
  Serial.begin(300);
 lcd.begin(16,2);
  pinMode(11, OUTPUT);
   pinMode(13, OUTPUT);
}

void loop()
{
  a2d_data=analogRead(A0);
  if(a2d_data)
  {
    buffer=a2d_data*Vin;
    Vout=(buffer)/1024.0;
    buffer=Vout/(Vin-Vout);
    Rx=R1*buffer*0.01;
Serial.print( A0);
  Serial.println(" analog value ----");
Serial.print( Rx);
  Serial.println(" ohms");

if ((Rx>0)&&(Rx<100))
 {
  digitalWrite(11, HIGH);
  digitalWrite(13, LOW);
  lcd.setCursor(0, 0); // set the cursor to column 0, line 1
  lcd.print("SC");
  lcd.setCursor(6, 0); // set the cursor to column 7, line 1

  lcd.print("R:");
  lcd.print(Rx);
  lcd.print(" Ohm");

  //Display Fault Location
  lcd.setCursor(0, 1); // set the cursor to column 10, line 2
  lcd.print("D:");
  lcd.print((Rx*0.01/Rc)/2);  //Find Location of Fault
  lcd.print(" KM");
  delay(1000);

}
if ((Rx>101)&&(Rx<439))
{
  digitalWrite(11, LOW);
  digitalWrite(13, LOW);
  lcd.setCursor(0, 0); // set the cursor to column 0, line 1
  lcd.print("NF");
//   lcd.setCursor(5, 1); // set the cursor to column 0, line 1
//  lcd.print("NO FAULT");
  lcd.setCursor(3, 0); // set the cursor to column 4, line 1
  lcd.print("          ");
lcd.setCursor(0, 1); // set the cursor to column 4, line 1
  lcd.print("          ");
```
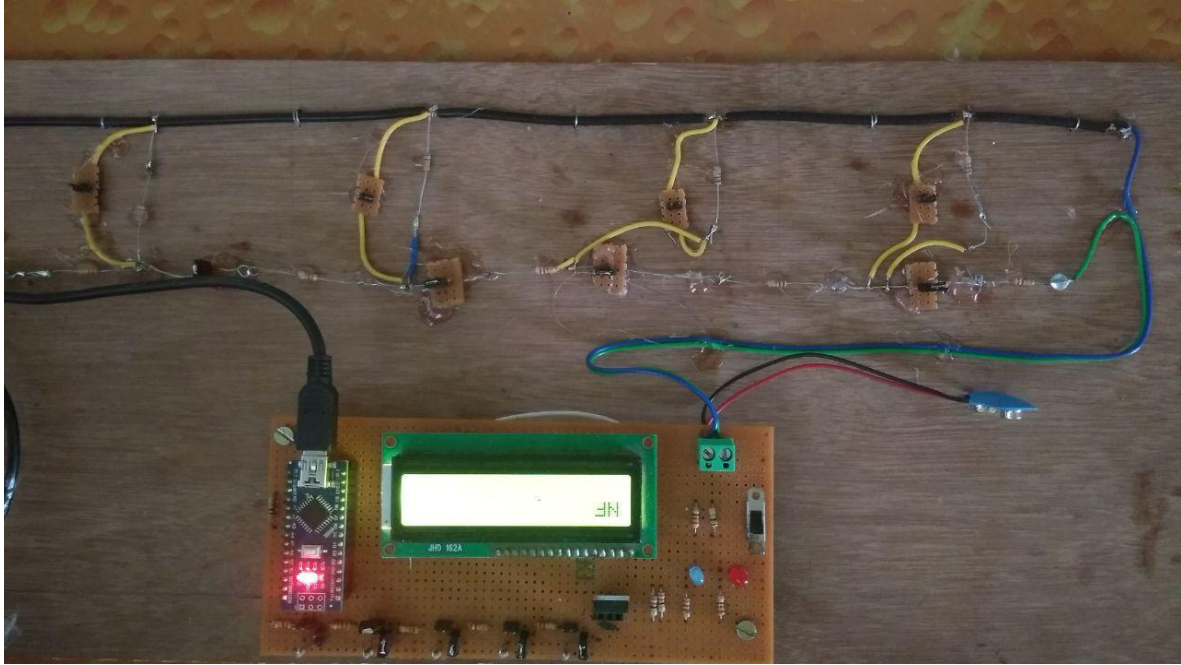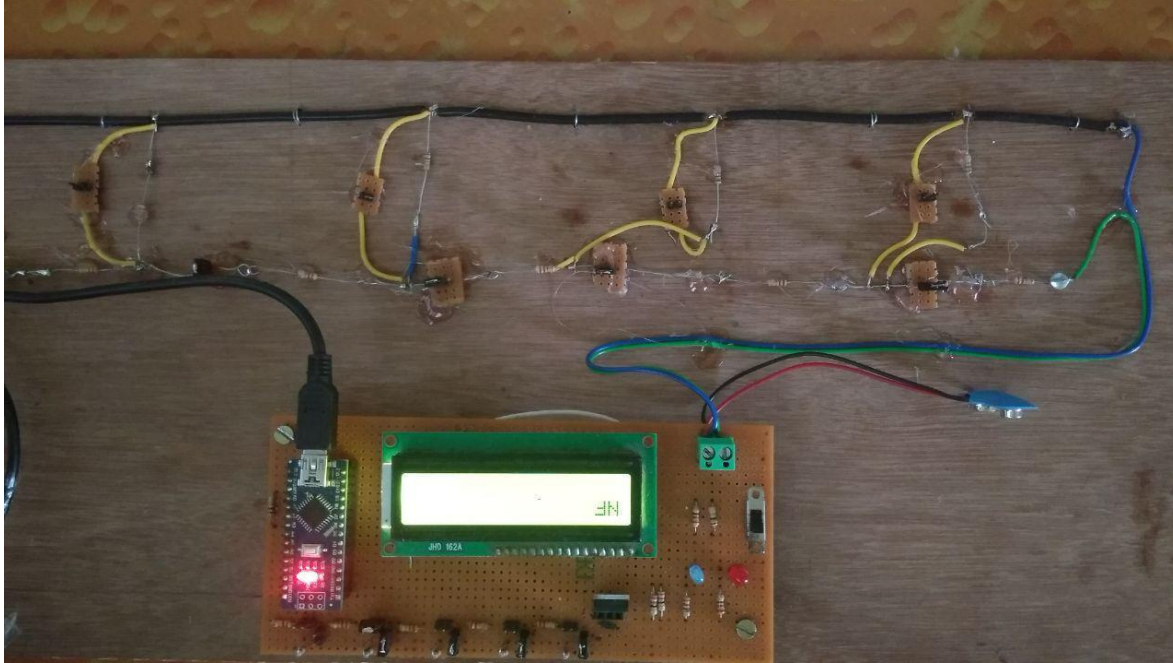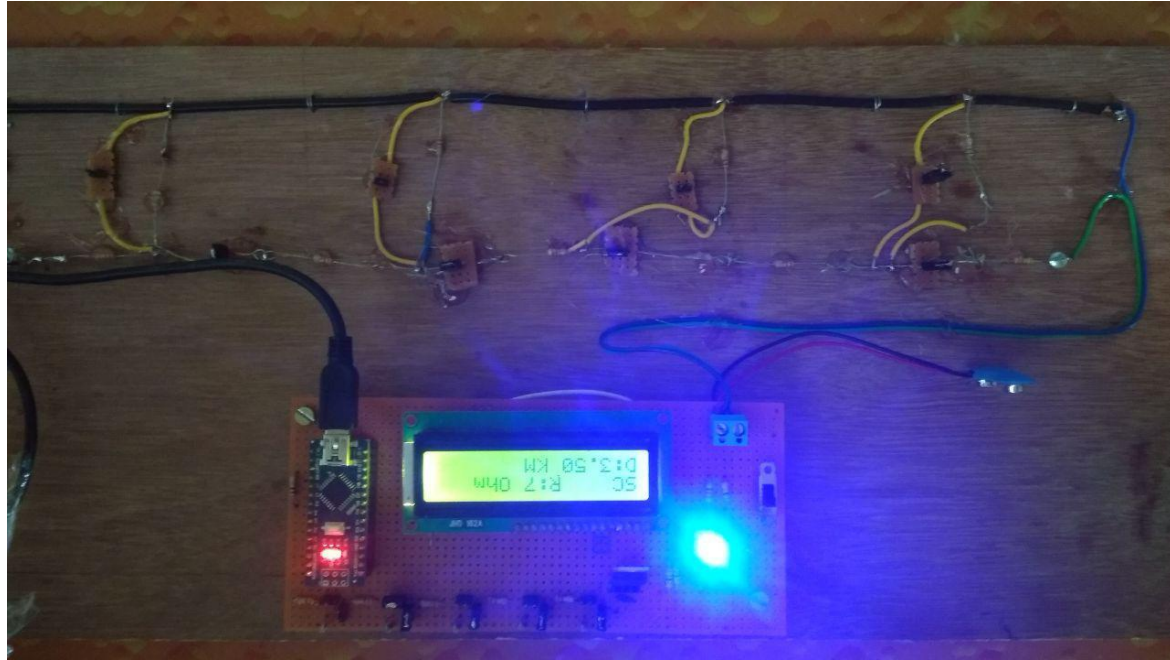
RESULT

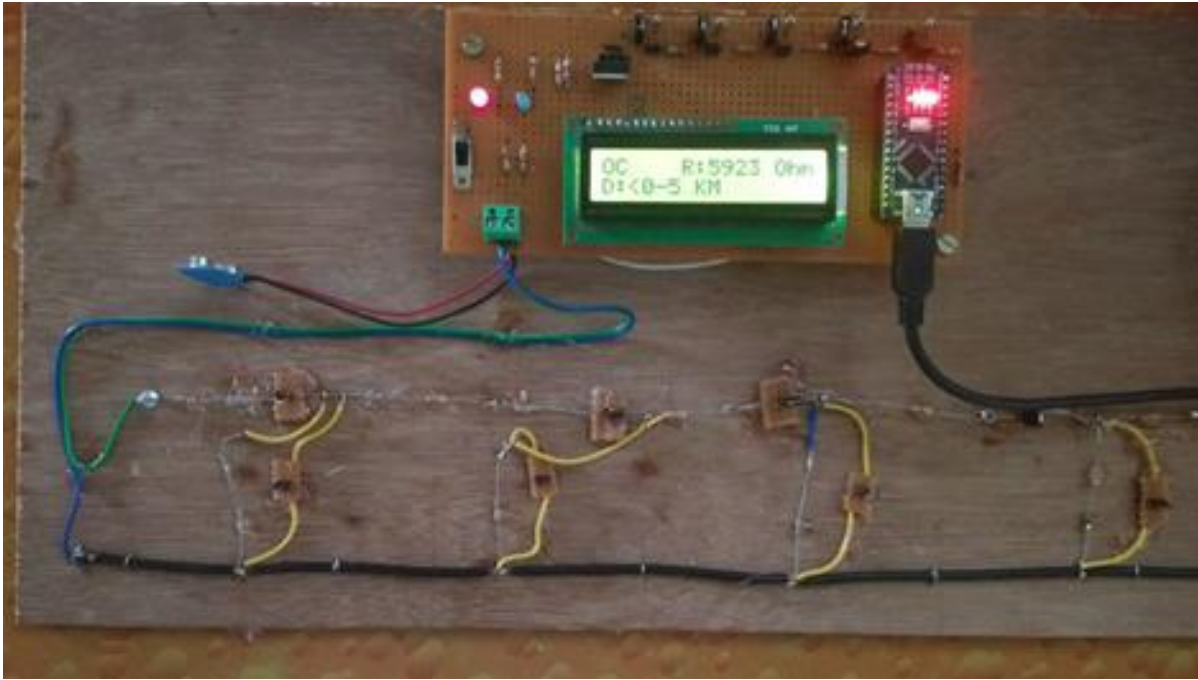Hardware implementation

NO FAULT CONDITION

power on condition
LCD Screen indicating NF (no fault condition)

Short circuit

BLUE Led indication

Open fault condition

RED Led condition

RESULT TABLE

	Short circuit			Open circuit		
	Distance (KM)	Resistance (Ω)	LED indication	Distance KM)	Resistance (Ω)	LED indication
Normal	NF (No Fault indication)		NO LED	NF (No Fault indication)		NO LED
Location 1	3.00	6	BLUE	0- 5KM	15000	RED
Location 2	6.50	13	BLUE	6.43	540	RED
Location 3	10.50	21	BLUE	10.18	488	RED
Location 4	15.00	30	BLUE	14.95	453	RED

Future scope

➢ In this project we detect only the location of short circuit fault in underground cable line, but we also detect the location of open circuit fault, to detect the open circuit fault capacitor is used in ac circuit which measure the change in impedance & calculate the distance of fault.

➢ It is used in neural network structure for fault section and fault location estimation.

conclusion

- The cable part is denoted by set of resistors along with switches

- Current sensing part of cable represented as set of resistors &switches are used as fault creators to indicate the fault at each location. This part senses the change in current by sensing the voltage drop.

- The controlling part which consist of analog to digital convertor which receives input from the current sensing circuit, converts this voltage into digital signal and feeds the microcontroller with the signal.

References

- http://www.edgefxkits.com

- https://www.elprocus.com

- http://electrical-engineering-portal.com

- http://ieeexplore.ieee.org

- http://www.electrical4u.com

Thank you