Experiment No: Date:

FAMILIARIZATION WITH ARDUINO/RASPBERRY PI AND PERFORM NECESSARY SOFTWARE INSTALLATION

AIM:

To get familiarization with Arduino/Raspberry Pi and to perform necessary software installation.

APPARATUS:

1. Arduino/Raspberry Pi Module
2. PC with Open Source Arduino/Raspberry Pi Software

THEORY:

Need to be written by Students by referring Text Books/Website.

PROCEDURE:

1. Connect the given three phase transformer as shown in the Circuit Diagram Fig1
2. After short-circuiting the low voltage side adjusts the voltage on high voltage side with the help of the autotransformer such that the rated current flows in the windings.
3. Note down the voltage and current.
4. From these readings determine the transformer positive sequence Impedance which is also equal to negative sequence impedance.
5. Determine the zero sequence impedance of the transformer by making connections as shown in Circuit DiagramFig2.
6. Note down the voltage and current.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT:

Experiment No: Date:

TO INTERFACE LED/BUZZER WITH ARDUINO/RASPBERRY PI AND WRITE A PROGRAM TO TURN ON LED FOR 1 SEC AFTER EVERY 2 SECONDS

AIM:

To interface LED/Buzzer with Arduino and write a program to turn ON LED for 1 sec after every 2 seconds.

APPARATUS:

1.	Arduino/Raspberry Pi Module	--- 1No
2.	PC with Open Source Arduino/Raspberry Pi Software	--- 1No
3.	IOT Development Board	--- 1No
4.	Male to Female connectors	--- As per requirement
5.	LED/Buzzer	--- 1No

THEORY:

Need to be written by Students by referring Text Books/Website.

PROGRAM:

```
int LED = 7;
void setup ()
{
  pinMode (LED,OUTPUT);
}
void loop()
{
  digitalWrite (LED,HIGH);
  delay(1000);
  digitalWrite (LED,LOW);
  delay(2000);
}
```

PROCEDURE:

1. Open the Arduino software and type the program and then save it.
2. Click for verification of code by clicking on verify button.
3. Click for uploading the code to Arduino board by clicking on Upload button.
4. Now place the Arduino board on to the IOT development board and give required connections.
5. Connect the LED/Buzzer terminals to the Arduino Digital input terminals as mentioned in the Program.
6. On the IOT development board and verify for the Output.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT:

Experiment No: Date:

TO INTERFACE PUSH BUTTON/DIGITAL SENSOR (IR/LDR) WITH ARDUINO/RASPBERRY PI AND WRITE A PROGRAM TO TURN ON LED WHEN PUSH BUTTON IS PRESSED OR AT SENSOR DETECTION

AIM:

To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.

APPARATUS:

1.	Arduino/Raspberry Pi Module	--- 1No
2.	PC with Open Source Arduino/Raspberry Pi Software	--- 1No
3.	IOT Development Board	--- 1No
4.	Male to Female connectors	--- As per requirement
5.	LED/Buzzer	--- 1No
6.	Push Button	--- 1No

THEORY:

Need to be written by Students by referring Text Books/Website.

PROGRAM:

```
const int led = 2;
const int input_switch = 3;
void setup() {
pinMode(led, OUTPUT);
pinMode(input_switch, INPUT_PULLUP);
}
void loop() {
if(digitalRead(input_switch) == LOW)
{
digitalWrite(led, HIGH);
delay(1000);
}
else
digitalWrite(led, LOW);
}
```

PROCEDURE:

1. Open the Arduino software and type the program and then save it.
2. Click for verification of code by clicking on verify button.
3. Click for uploading the code to Arduino board by clicking on Upload button.
4. Now place the Arduino board on to the IOT development board and give required connections.
5. Connect the LED/Buzzer and Push Button terminals to the Arduino Digital input terminals as mentioned in the Program.
6. On the IOT development board and verify for the Output.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT:

Experiment No: Date:

TO INTERFACE TEMPERATURE SENSOR WITH ARDUINO/RASPBERRY PI AND WRITE A PROGRAM TO PRINT TEMPERATURE AND HUMIDITY READINGS

AIM:

To interface temperature sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings.

APPARATUS:

1.	Arduino/Raspberry Pi Module	--- 1No
2.	PC with Open Source Arduino/Raspberry Pi Software	--- 1No
3.	IOT Development Board	--- 1No
4.	Male to Female connectors	--- As per requirement
5.	Temperature Sensor	--- 1No
6.	LCD Display	--- 1No

THEORY:

Need to be written by Students by referring Text Books/Website.

PROGRAM:

```
#include <dht.h>
#include<LiquidCrystal.h>
#define DHT11_PIN 8
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
dht DHT;
void setup()
{
 lcd.begin(16, 2);
}
void loop()
{
 int chk = DHT.read11(DHT11_PIN);
 lcd.setCursor(0,0);
 lcd.print("TMP:          ");
 lcd.setCursor(0,1);
 lcd.print("Hum:          ");
 lcd.setCursor(4,0);
lcd.print(DHT.temperature);
lcd.setCursor(4,1);
lcd.print(DHT.humidity);
delay(500);
}
```

PROCEDURE:

1. Open the Arduino software and type the program and then save it.
2. Click for verification of code by clicking on verify button.
3. Click for uploading the code to Arduino board by clicking on Upload button.
4. Now place the Arduino board on to the IOT development board and give required connections.
5. Connect the Temperature Sensor and LCD Display terminals to the Arduino Digital input terminals as mentioned in the Program.
6. On the IOT development board and observe for the Output on the LCD Screen.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT:

Experiment No: Date:

TO INTERFACE ORGANIC LIGHT EMITTING DIODE (OLED) WITH ARDUINO/RASPBERRY PI

AIM:

To interface Organic Light Emitting Diode (OLED) with Arduino/Raspberry Pi.

APPARATUS:

1.	Arduino/Raspberry Pi Module	--- 1No
2.	PC with Open Source Arduino/Raspberry Pi Software	--- 1No
3.	IOT Development Board	--- 1No
4.	Male to Female connectors	--- As per requirement
5.	OLED	--- 1No

THEORY:

Need to be written by Students by referring Text Books/Website.

PROGRAM:

```
/* Starting with Arduino OLED coding
 *  for " arduino oled i2c tutorial : 0.96" 128 X 32 for beginners"
 *  subscribe for more arduino Tuorials and Projects
https://www.youtube.com/channel/UCM6rbuieQBBLFsxszWA85AQ?sub_confirmation=1
 */
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define OLED_RESET 4
Adafruit_SSD1306 display(OLED_RESET);
void setup()
{
  display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
  display.clearDisplay();
}
void loop()
{
  display.setTextSize(2);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println("  DHANEKULA");
  display.display();
}
```

PROCEDURE:

1. Open the Arduino software and type the program and then save it.
2. Click for verification of code by clicking on verify button.
3. Click for uploading the code to Arduino board by clicking on Upload button.
4. Now place the Arduino board on to the IOT development board and give required connections.
5. Connect the OLED terminals to the Arduino Analog input terminals as mentioned in the Program.
6. On the IOT development board and verify for the Output displayed on the OLED screen.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT:

Experiment No: Date:

TO INTERFACE 7 SEGMENT DISPLAY WITH ARDUINO/RASPBERRY PI

AIM:

To interface 7 Segment Display with Arduino/Raspberry Pi.

APPARATUS:

1.	Arduino/Raspberry Pi Module	--- 1No
2.	PC with Open Source Arduino/Raspberry Pi Software	--- 1No
3.	IOT Development Board	--- 1No
4.	Male to Female connectors	--- As per requirement
5.	7 Segment Display	--- 1No

THEORY:

Need to be written by Students by referring Text Books/Website.

PROGRAM:

```
void setup()
{
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
}
void loop()
{
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, LOW);
    delay(1500);
    digitalWrite(2, LOW);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, LOW);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, LOW);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, LOW);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, LOW);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, LOW);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, LOW);
    digitalWrite(3, LOW);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, LOW);
    delay(1500);
    digitalWrite(2, LOW);
    digitalWrite(3, HIGH);
    digitalWrite(4, HIGH);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, HIGH);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, LOW);
    digitalWrite(6, HIGH);
    digitalWrite(7, HIGH);
    digitalWrite(8, HIGH);
    delay(1500);
    digitalWrite(2, LOW);
    digitalWrite(3, LOW);
    digitalWrite(4, LOW);
    digitalWrite(5, LOW);
    digitalWrite(6, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, LOW);
}
```

PROCEDURE:

1. Open the Arduino software and type the program and then save it.
2. Click for verification of code by clicking on verify button.
3. Click for uploading the code to Arduino board by clicking on Upload button.
4. Now place the Arduino board on to the IOT development board and give required connections.
5. Connect the 7 Segment terminals to the Arduino Digital input terminals as mentioned in the Program.
6. On the IOT development board and verify for the Output displayed on the 7 segment display.

PRECAUTIONS:

1. Connections are to be made tightly and according to the circuit diagram.
2. While making or breaking connections supply has to switched OFF.

RESULT: