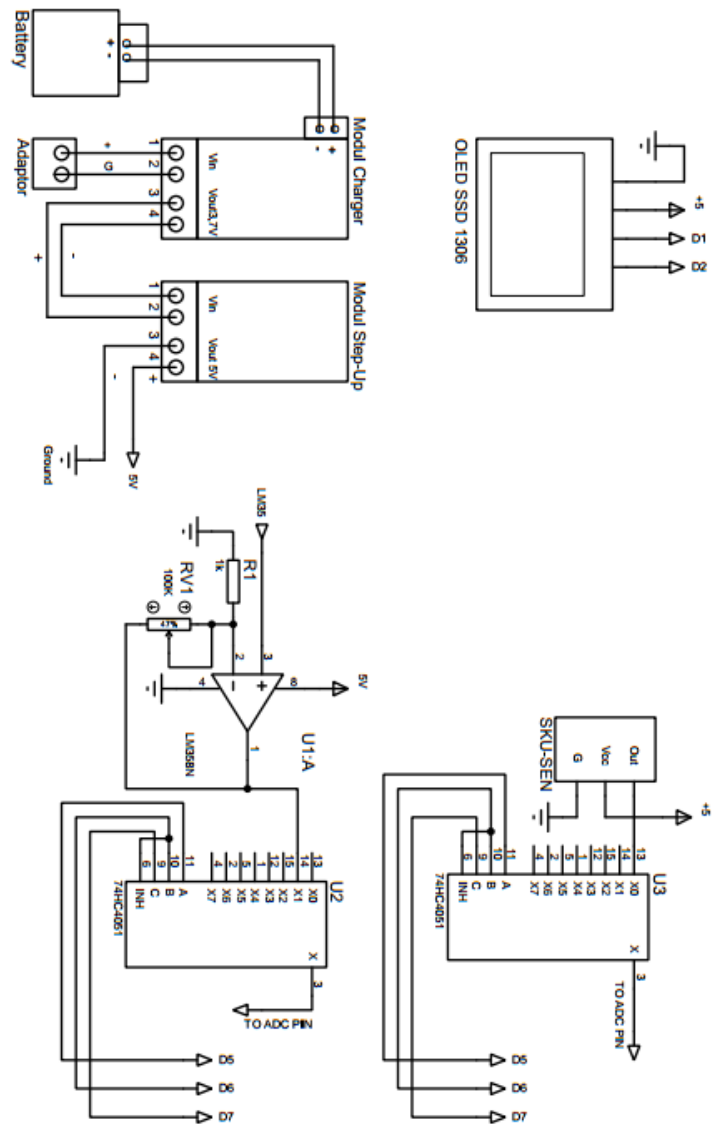


LAMPIRAN

1. Gambar Rangkaian Keseluruhan



2. Coding Program Arduino

```
#include <declare.h>
//Wifi
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
//OLED
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET D5
/* Object named display, of the class Adafruit_SSD1306 */
Adafruit_SSD1306 display(OLED_RESET);

#if (SSD1306_LCDHEIGHT != 64)
#error("Height incorrect, please fix Adafruit_SSD1306.h!");
#endif

//analog input
const int analogPin = A0;
//LM35 config
float data;
float teg;
float suhu;
unsigned int adc;
int suhu1_integer;
String suhu1_string;
//dbmeee
float db;
float data0;
float teg0;
unsigned int adc0;
//multiplexer
int s0 = 5;
int s1 = 6;
int s2 = 7;

void inputMultiplekser(int y)
{
  if (y==0){
    digitalWrite(s0,LOW);
    digitalWrite(s1,LOW);
  }
}
```

```

        digitalWrite(s2,LOW); }
    else if (y==1){
        digitalWrite(s0,HIGH);
        digitalWrite(s1,LOW);
        digitalWrite(s2,LOW); }
    }

void temperature()
{
    adc=0;
    inputMultiplekser(1);
    for (int i=0; i<30; i++)
    {
        adc += analogRead(analogPin);
    }
    data = adc /30;
    teg = data * (3.3 / 1023);
    suhu = teg * 16.6 ;
    suhu1_integer =(int)suhu;
    //suhu1_string =(String)suhu1_integer;
    delay (250);
}

void kebisingan()
{
    adc0= 0;
    inputMultiplekser(0);
    for(int i=0;i<50;i++)
    {
        adc0 +=analogRead(analogPin);
    }
    data0 =(adc0 / 50);
    teg0 =(data0*3.3) / 1023;
    db =(11.98*teg0)+38.9;
    //teg=teg/100;
    //konversi_data();
    //delay(250);
}

void showTemp(float suhu,float db)
{
    display.clearDisplay();

```

```

display.setTextSize(2);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println("  Temp  ");

display.setTextSize(3);           // Size4 = 5 digit , size3 = 7
digits
//display.setTextColor(BLACK, WHITE);           // 'inverted' text
display.setTextColor(WHITE);
display.setCursor(0,24);

display.print(suhu,1);
display.println("C");
display.display();
delay(1000);
}

void showdB(float suhu,float db)
{

display.clearDisplay();
display.setTextSize(2);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println("  dB  ");

display.setTextSize(3);
display.setTextColor(WHITE);
display.setCursor(0,24);

display.print(db,1);
display.println("V");
display.display();
delay(1000);
}

void setup()
{
// put your setup code here, to run once:

Serial.begin(9600); //Serial begin

pinMode(s0, OUTPUT);

```

```

pinMode(s1, OUTPUT);
pinMode(s2, OUTPUT);

pinMode (BUILTIN_LED, OUTPUT);
digitalWrite(BUILTIN_LED, HIGH);

display.begin(SSD1306_SWITCHCAPVCC, 0x3C);

// Clear the buffer.
display.clearDisplay();

display.setTextSize(2);
display.setTextColor(WHITE);
//display.setTextColor(BLACK, WHITE); // 'inverted' text
display.setCursor(0,0);
display.println("Monitoring");

display.setTextSize(2);          // Size4 = 5 digit , size3 = 7 digits
//display.setTextColor(BLACK, WHITE); // 'inverted' text
display.setTextColor(WHITE);
display.setCursor(0,18);
display.println("  Suhu  ");

display.setTextSize(2);
display.setTextColor(WHITE);
//display.setTextColor(BLACK, WHITE); // 'inverted' text
display.setCursor(0,36);
display.println("  dB  ");

display.display();
delay(3000);

//conect to wifi
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("connecting");

display.clearDisplay(); // Clear the buffer.

display.setTextSize(2);
display.setTextColor(WHITE);
//display.setTextColor(BLACK, WHITE); // 'inverted' text
display.setCursor(0,0);
display.println("Connecting");

```

```
while (WiFi.status() != WL_CONNECTED) // Wait for connection
{
  Serial.print(".");
  display.print(".");
  display.display();
  delay(500);
}
```

```
Serial.println("");
Serial.print("Connected to ");
//Serial.println(ssid);
Serial.print("IP address: ");
Serial.println(WiFi.localIP());
```

```
display.clearDisplay();
display.setTextSize(2);
display.setTextColor(WHITE);
display.setCursor(0,0);
display.println(WiFi.SSID());
display.setCursor(9,18);
display.println(WiFi.localIP());
//display.setCursor(0,36);
//display.println(WiFi.localIP());
```

```
display.display();
delay(2000);
```

```
Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
}
```

```
void loop()
{
  // put your main code here, to run repeatedly:
  unsigned long currentMillis = millis();

  if(currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    digitalWrite(BUILTIN_LED,LOW);

    temperature();
    kebisingan();
  }
}
```

```

Serial.print("Temperature: ");
Serial.print(suhu);
Serial.print(" *C\t");
Serial.print("dB: ");
Serial.print(db);
Serial.print(" V \n");

if (showState == LOW)
{
  showState = HIGH;
  showTemp(suhu,db);
}
else
{
  showState = LOW;
  showdB(suhu,db); // show temp
}

lp++;
if(lp >= lp_time){
  lp=0;
  digitalWrite(BUILTIN_LED,LOW);
  Firebase.setFloat ("temperature",suhu1_integer);
  Firebase.setFloat ("dB",db);
  // handle error
if (Firebase.failed()) {
  Serial.print("Firebase Pushing /dbmeiot-abf85/failed:");
  Serial.println(Firebase.error());
  display.clearDisplay();
  display.setTextSize(2);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println(" Firebase ");

  display.setTextSize(3);
  display.setTextColor(WHITE);
  display.setCursor(0,24);
  display.println("ERROR");
  display.display();

  digitalWrite(BUILTIN_LED,HIGH);
  delay(2000);
  lp = 2; // repeat to sent again

```

```

return; }
else
{
  Serial.print("Firebase Pushed /dbmeiot-abf85 ");
  Serial.println("Done");

  display.clearDisplay();
  display.setTextSize(2);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println(" Firebase ");

  display.setTextSize(2);
  display.setTextColor(WHITE);
  display.setCursor(0,24);

  display.println("Update OK");
  display.display();
  digitalWrite(BUILTIN_LED,HIGH);
  delay(500);
}
}
digitalWrite(BUILTIN_LED,HIGH);
}}

```

3. Coding Program MIT App Inventor

The image shows two blocks for initializing global variables and a larger block for handling Firebase data changes.

- Block 1:** initialize global Temp to create empty list
- Block 2:** initialize global dBme to create empty list
- Block 3:** when FirebaseDB1 .DataChanged
 - do
 - if get tag = "Temp" then set global Temp to get value
 - if get tag = "dBme" then set global dBme to get value


```

when Clock1.Timer
do
  call readdata
  if is number? get global Temp and get global Temp > 37
  then
    set status1.Text to "suhu tidak normal"
    set hasilsuhu.BackgroundColor to red
    call Sound1.Vibrate
      milliseconds 200
    call Sound1.Play
  else
    set status1.Text to "Suhu Normal"
    set hasilsuhu.BackgroundColor to gray
  end if
  if is number? get global dBme and get global dBme > 60
  then
    set status2.Text to "Nilai dB tidak normal"
    set hasildb.BackgroundColor to red
    call Sound1.Play
    call Sound1.Vibrate
      milliseconds 200
  else
    set status2.Text to "dB Normal"
    set hasildb.BackgroundColor to gray
  end if
end do

```

```

when Button1.Click
do
  close application
end do

```

```

to readdata
do
  if is number? get global Temp
  then
    set hasilsuhu.Text to join
      format as decimal number get global Temp
      places 1
      " °C"
  end if
  if is number? get global dBme
  then
    set hasildb.Text to join
      format as decimal number get global dBme
      places 1
      " dB"
  end if
end do
end to

```