

PRAKTIKUM INTERNET OF THINGS

**“PROJECT PIR, IMU, ULTRASONIC SENSOR MODUL DENGAN
BLYNK”**



KELOMPOK 1

M. Arya Izzulhaq (1123800001)

Daffa Syah A (1123800013)

Utari Sanaba (1122800021)

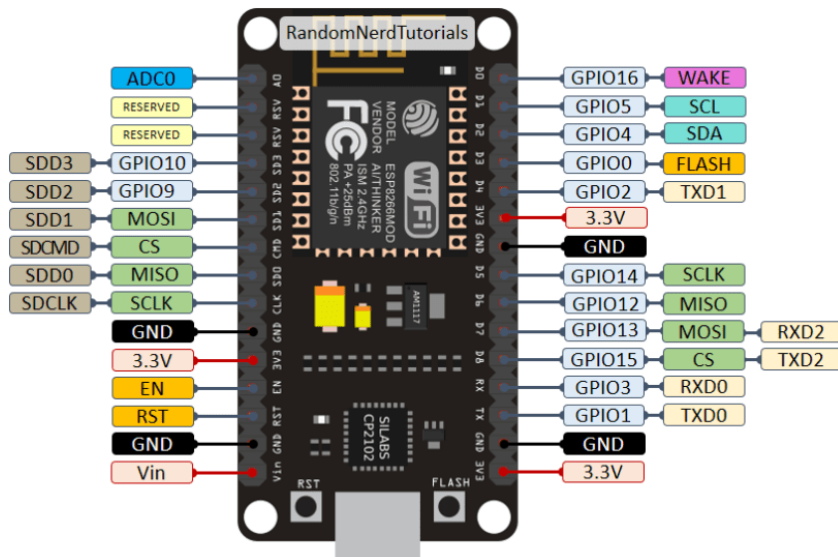
PROGRAM STUDI MAGISTER TERAPAN TEKNIK ELEKTRO

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

TAHUN AJARAN 2023/2024

A. MIKROKONTROLLER

Mikrokontroller yang digunakan pada project ini yaitu NodeMCU ESP 8266. NodeMCU adalah perangkat seperti Arduino. Komponen utamanya adalah ESP8266, memiliki pin yang dapat diprogram dan memiliki WiFi bawaan. Serta bisa mendapatkan daya melalui port micro-usb. Berikut board ESP8266 yang digunakan beserta pinoutnya.



Gambar 1. Pinout ESP8266

B. SENSOR

Terdapat 3 sensor yang digunakan pada project ini yaitu sensor PIR, Ultrasonic, dan IMU

1) Sensor PIR (Passive Infrared Sensor)

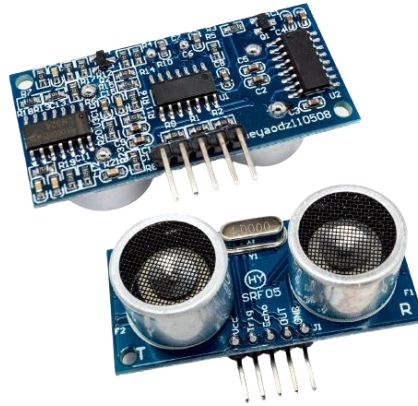
Sebuah sensor yang digunakan untuk mengetahui adanya pergerakan atau tidaknya suatu objek yang berada didepannya.



Gambar 2. Sensor PIR

2) Sensor Ultrasonic

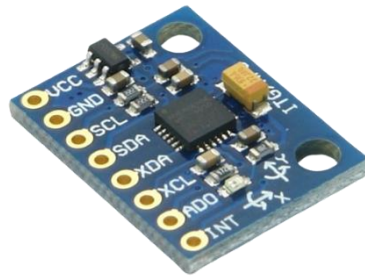
Sebuah sensor yang digunakan untuk mendeteksi objek atau mengukur jarak dengan cara mengirimkan gelombang ultrasonik ke objek lalu dipantulkan kembali.



Gambar 3. Sensor Ultrasonic

3) Sensor IMU (Inertial Measurement Unit)

Sebuah sensor yang digunakan untuk mengukur gerakan dan orientasi suatu objek dalam tiga dimensi (biasanya disebut sebagai tiga sumbu: sumbu x, y, dan z).



Gambar 4. Sensor IMU

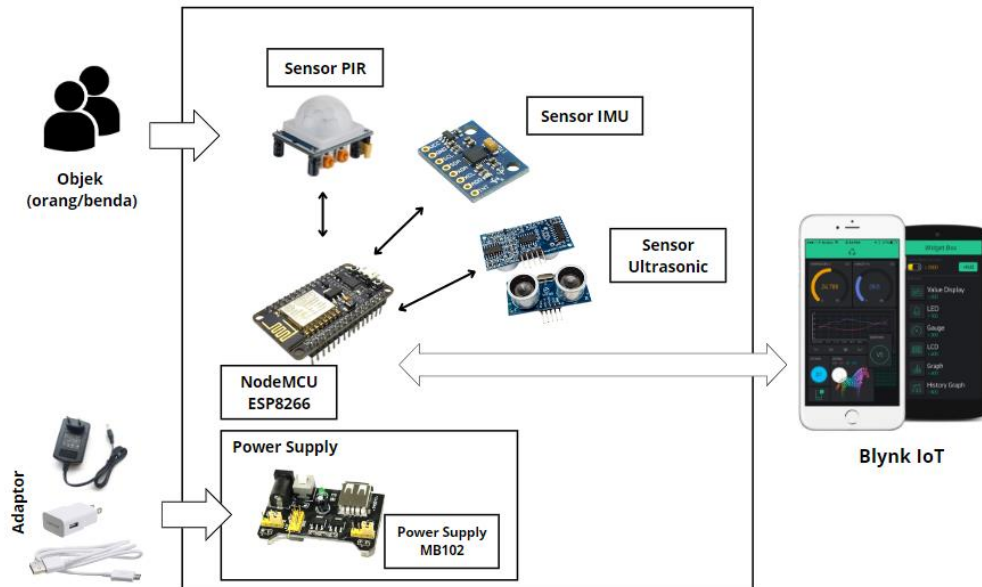
C. KOMPONEN YANG DIGUNAKAN

Berikut komponen yang digunakan pada proyek ini:

- Kabel jumper
- Breadboard
- Power supply MB102
- Sensor ultrasonic
- Sensor IMU
- Sensor PIR
- ESP8266

D. DESAIN SISTEM

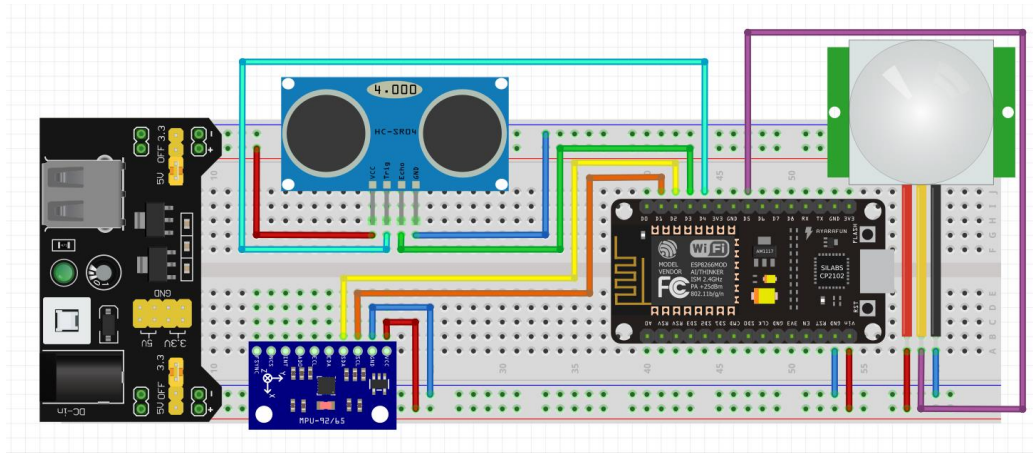
Berikut desain sistem yang dibuat pada project ini.



Gambar 5. Desain Sistem

E. RANGKAIAN SISTEM

Berikut rangkaian sistem pada project ini yang menghubungkan komponen satu dengan yang lainnya.



Gambar 6. Rangkaian Sistem

F. KODE PROGRAM

Berikut kode program dari project ini menggunakan Bahasa C yang decompile pada Arduino IDE

```
#define BLYNK_TEMPLATE_ID "TMPL6xJNCYqdm"
#define BLYNK_TEMPLATE_NAME "EMU PIR Ultrasonic ESP8266"
#define BLYNK_AUTH_TOKEN "fo6CT7wdfjJ0cCSy4ljUfgke6EemqoX"

#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <MPU9250_asukiaaa.h>

char ssid[] = "alakazookas";
char pass[] = "oogaboogazu";

const int trigPin = D4;
const int echoPin = D3;
const int pirPin = D5;

MPU9250_asukiaaa mySensor;
float aX, aY, aZ, aSqrt, gX, gY, gZ, mDirection, mX, mY, mZ, second;

BlynkTimer timer;

void setup() {
  Serial.begin(115200);
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);

  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(pirPin, INPUT);

  timer.setInterval(1000L, sendData);

#ifdef _ESP32_HAL_I2C_H_ // For ESP32
  Wire.begin(SDA_PIN, SCL_PIN);
  mySensor.setWire(&Wire);
#endif
}
```

```

mySensor.beginAccel();
mySensor.beginGyro();
mySensor.beginMag();

// Pengaturan offset untuk magnetometer
// mySensor.magXOffset = -50;
// mySensor.magYOffset = -55;
// mySensor.magZOffset = -10;
}

void sendData() {
  int result;

  // Pembacaan gerakan oleh PIR Sensor
  int pirValue = digitalRead(pirPin);
  if (pirValue == HIGH) {
    Blynk.virtualWrite(V0, "Motion detected!");
    Serial.println("Motion detected!");
    // Pengiriman notifikasi saat terdeteksi gerakan
    //Blynk.notify("Motion detected!");
  } else {
    Blynk.virtualWrite(V0, "No motion detected");
    Serial.println("No motion detected");
  }

  // Pembacaan jarak oleh HC-SR04 Ultrasonic Sensor
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration / 2) / 29.1; //Pembagian 29.1 ke centimeter

  if (distance >= 2 && distance <= 400) {
    Blynk.virtualWrite(V1, distance);
    Serial.print("Distance: ");
    Serial.print(distance);
    Serial.println(" cm");
  }
}

```

```

    } else {
        Blynk.virtualWrite(V1, "Out of range");
        Serial.println("Out of range");
    }

    // Pembacaan accelometer, gryroscope dan magnetometer oleh MPU9250 EMU
    sensor
    result = mySensor.accelUpdate();
    if (result == 0) {
        aX = mySensor.accelX();
        aY = mySensor.accelY();
        aZ = mySensor.accelZ();
        aSqrt = mySensor.accelSqrt();
        Serial.println("accelX: " + String(aX));
        Serial.println("accelY: " + String(aY));
        Serial.println("accelZ: " + String(aZ));
        Serial.println("accelSqrt: " + String(aSqrt));
    } else {
        Serial.println("Cannot read accel values " + String(result));
    }

    result = mySensor.gyroUpdate();
    if (result == 0) {
        gX = mySensor.gyroX();
        gY = mySensor.gyroY();
        gZ = mySensor.gyroZ();
        Serial.println("gyroX: " + String(gX));
        Serial.println("gyroY: " + String(gY));
        Serial.println("gyroZ: " + String(gZ));
    } else {
        Serial.println("Cannot read gyro values " + String(result));
    }

    result = mySensor.magUpdate();
    if (result != 0) {
        Serial.println("Cannot read mag, calling beginMag() again");
        mySensor.beginMag();
        result = mySensor.magUpdate();
    }
    if (result == 0) {

```

```
    mX = mySensor.magX();
    mY = mySensor.magY();
    mZ = mySensor.magZ();
    mDirection = mySensor.magHorizDirection();
    Serial.println("magX: " + String(mX));
    Serial.println("magY: " + String(mY));
    Serial.println("magZ: " + String(mZ));
    Serial.println("horizontal direction: " + String(mDirection));
  } else {
    Serial.println("Cannot read mag values " + String(result));
  }
}
```

```
second = millis() / 1000.0; // Konversi millis ke detik
Serial.println("at " + String(second, 3) + " seconds");
```

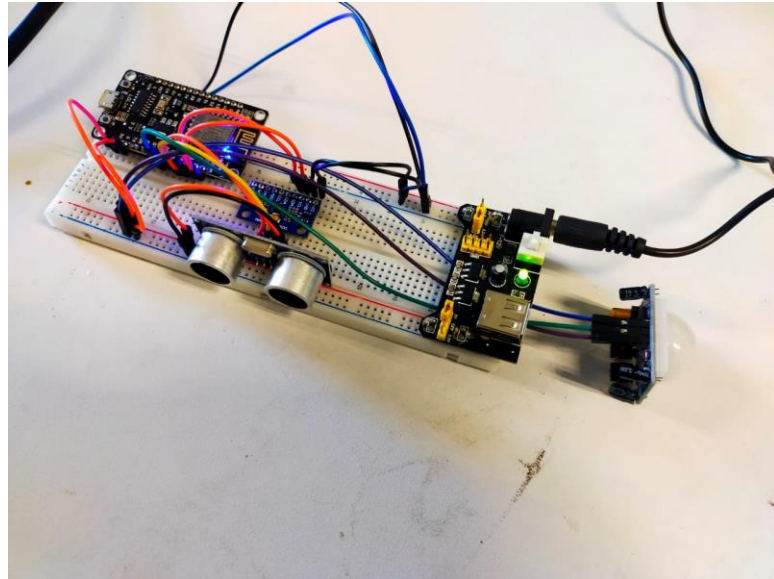
```
// Pengiriman data sensor ke Blynk IoT
Blynk.virtualWrite(V2, aX);
Blynk.virtualWrite(V3, aY);
Blynk.virtualWrite(V4, aZ);
Blynk.virtualWrite(V5, gX);
Blynk.virtualWrite(V6, gY);
Blynk.virtualWrite(V7, gZ);
Blynk.virtualWrite(V8, mX);
Blynk.virtualWrite(V9, mY);
Blynk.virtualWrite(V10, mZ);
Blynk.virtualWrite(V11, mDirection);
```

```
Serial.println("");
}
```

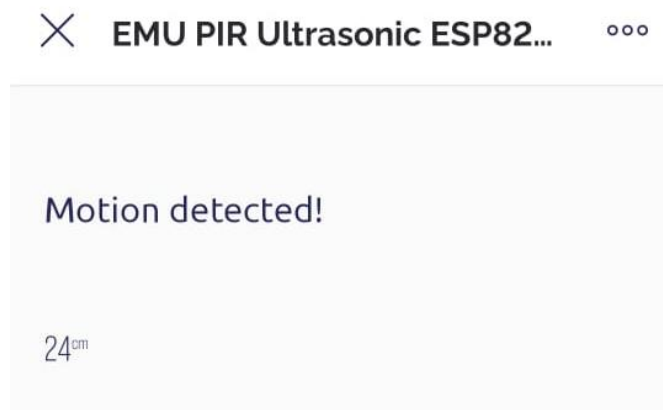
```
void loop() {
  Blynk.run();
  timer.run();
}
```


G. HASIL PROJECT

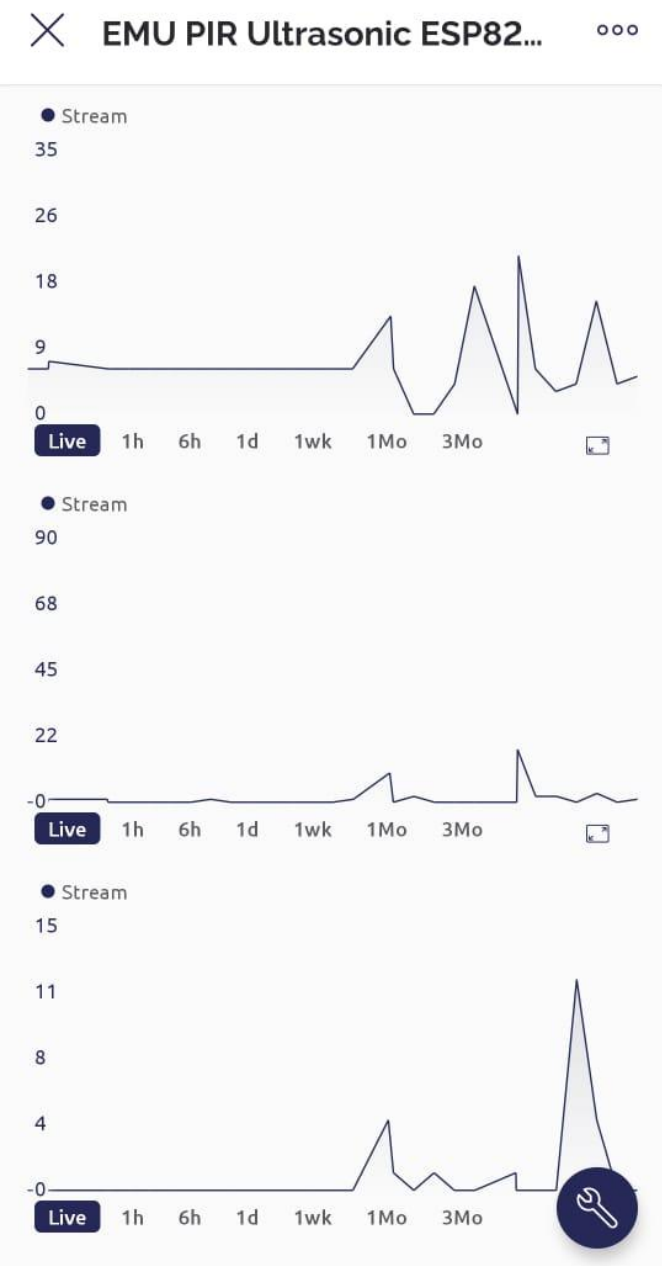
Berikut hasil project yang telah dilakukan terdiri dari proses pembacaan sensor, pengiriman data ke Blynk, dan data sensor ditampilkan pada dashboard.



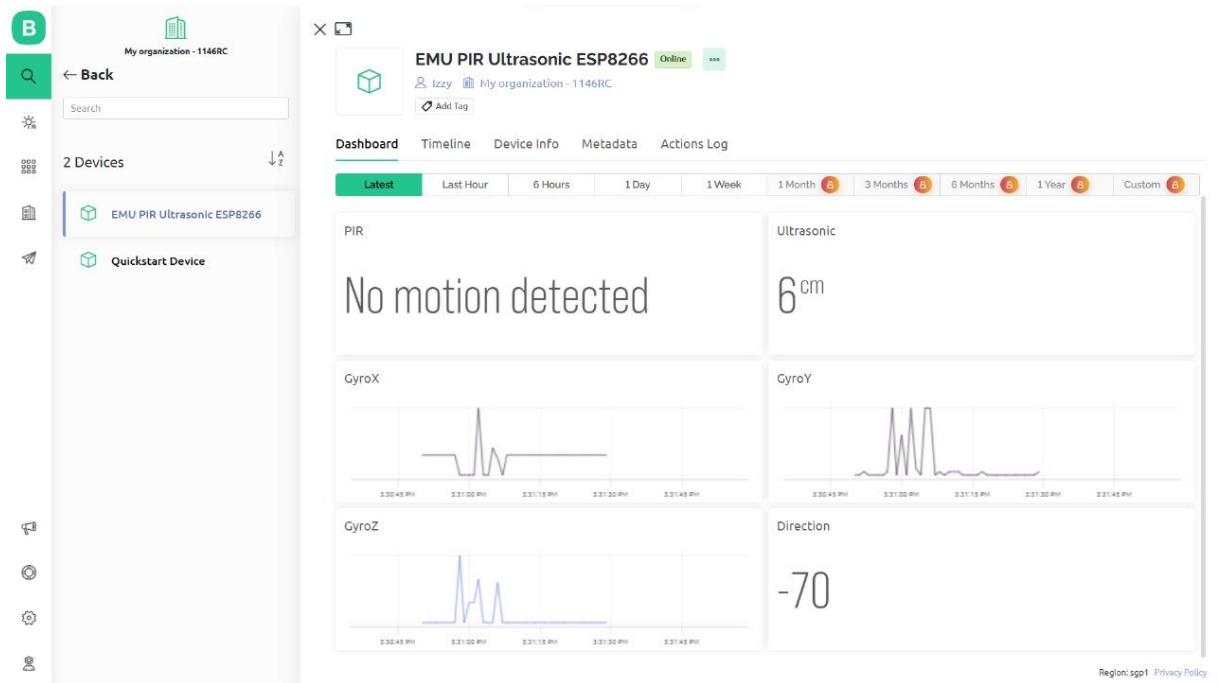
Gambar 7. Sensor melakukan proses pembacaan data



Gambar 8. Tampilan dashboard dari android Blynk untuk data PIR



Gambar 9. Tampilan grafik dari data sensor yang terbaca



Gambar 10. Tampilan dashboard website untuk semua data sensor