**Koding untuk Transmitter (ESP32)**

1. Library yang dibutuhkan:

* Wire.h untuk komunikasi I2C
* MPU6050.h untuk sensor MPU6050
* WiFi.h untuk koneksi Wi-Fi
* WiFiUdp.h untuk komunikasi UDP (transmisi data)

1. Kode Transmitter (ESP32)

#include <Wire.h>

#include <MPU6050.h>

#include <WiFi.h>

#include <WiFiUdp.h>

const char\* ssid = "Your\_SSID"; // Masukkan SSID Wi-Fi

const char\* password = "Your\_PASSWORD"; // Masukkan Password Wi-Fi

const char\* udpAddress = "192.168.1.100"; // IP address ESP Receiver

const int udpPort = 12345;

MPU6050 mpu;

WiFiUDP udp;

void setup() {

Serial.begin(115200);

Wire.begin();

mpu.begin();

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.println("Connecting to WiFi...");

}

Serial.println("Connected to WiFi");

}

void loop() {

mpu.update();

String sensorData = String(mpu.getAccX()) + "," +

String(mpu.getAccY()) + "," +

String(mpu.getAccZ()) + "," +

String(mpu.getGyroX()) + "," +

String(mpu.getGyroY()) + "," +

String(mpu.getGyroZ());

udp.beginPacket(udpAddress, udpPort);

udp.print(sensorData);

udp.endPacket();

delay(100); // delay to control transmission speed

}

**Koding untuk Receiver (ESP32)**

1. Library yang dibutuhkan:

* WiFi.h untuk koneksi Wi-Fi
* WiFiUdp.h untuk komunikasi UDP (transmisi data)
* SPIFFS.h untuk menyimpan data ke file CSV

1. Kode Receiver

#include <WiFi.h>

#include <WiFiUdp.h>

#include "SPIFFS.h"

const char\* ssid = "Your\_SSID";

const char\* password = "Your\_PASSWORD";

const int udpPort = 12345;

WiFiUDP udp;

char incomingPacket[255];

void setup() {

Serial.begin(115200);

if (!SPIFFS.begin(true)) {

Serial.println("Failed to mount file system");

return;

}

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.println("Connecting to WiFi...");

}

Serial.println("Connected to WiFi");

udp.begin(udpPort);

createCSVFile();

}

void loop() {

int packetSize = udp.parsePacket();

if (packetSize) {

int len = udp.read(incomingPacket, 255);

if (len > 0) {

incomingPacket[len] = 0;

}

Serial.printf("Received: %s\n", incomingPacket);

logData(incomingPacket);

}

}

void createCSVFile() {

File file = SPIFFS.open("/log.csv", FILE\_WRITE);

if (!file) {

Serial.println("Failed to create file");

return;

}

file.println("AccX,AccY,AccZ,GyroX,GyroY,GyroZ");

file.close();

}

void logData(String data) {

File file = SPIFFS.open("/log.csv", FILE\_APPEND);

if (!file) {

Serial.println("Failed to open file for appending");

return;

}

file.println(data);

file.close();

}