

## LAMPIRAN

Source Code Pengiriman Data NodeMCU ESP8266 :

```
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <SoftwareSerial.h>
#include <ArduinoJson.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClient.h>
WiFiClient client;
ESP8266WiFiMulti WiFiMulti;

String apiKey = "BYJ77I5HD3HHASZ0"; // Apikey Thingspeak

SoftwareSerial s(D6, D5);

const char *ssid = "Mi 8"; // Setting WiFi yang akan digunakan
const char *pass = "12345678";
const char* server = "api.thingspeak.com";

void setup()
{
  Serial.begin(9600); // Initializes serial port
  s.begin(9600);

  while ( !Serial ) ;
  delay(1000);

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

  WiFi.begin(ssid, pass);
  if (WiFi.status() != WL_CONNECTED)
    return;

  Serial.println("WiFi connected");
}

void loop() {
  if(WiFi.status() == WL_CONNECTED){
    //Arduino Json v5
    StaticJsonBuffer<1000> jsonBuffer;
    JsonObject& root = jsonBuffer.parseObject(s);
    if (root == JsonObject::invalid())
```

```

return;

int sensor_lc = root["sensor_LcGram"];
int sensor_lcToPersen = root["sensor_LcToPersen"];
int sensor_mq6 = root["sensor_mq6"];

Serial.print("sensor MQ-6: ");
Serial.println(sensor_mq6);

Serial.print("sensor LC: ");
Serial.print(sensor_lc);
Serial.println(" Gram");

Serial.print("sensor LC Persen: ");
Serial.print(sensor_lcToPersen);
Serial.println(" %");

    if (isnan(sensor_lc) || isnan(sensor_lcToPersen) )
    {
        Serial.println("Failed to read ");
        return;
    }

    if (client.connect(server,80))
    {

        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(sensor_mq6);
        postStr += "&field2=";
        postStr += String(sensor_lcToPersen);
        postStr += "&field3=";
        postStr += String(sensor_lc);
        postStr += "\r\n\r\n";

        client.print("POST /update HTTP/1.1\n");
        client.print("Host: api.thingspeak.com\n");
        client.print("Connection: close\n");
        client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
        client.print("Content-Type:application/x-www-form
urlencoded\n");
        client.print("Content-Length: ");
        client.print(postStr.length());
        client.print("\n\n");
        client.print(postStr);

```

```

        Serial.println("Mengirim data ke ThingSpeak");

    }
    client.stop();
    Serial.println("\n");
    Serial.println("Waiting...");
} else {
    Serial.println("Koneksi Gagal");
}
}
delay(15000);
}

```

#### Source code Kalibrasi Loadcell HX711

```

#include "HX711.h"
#define DOUT D5
#define CLK D6
HX711 scale;
float calibration_factor = 650;
int GRAM;

void setup() {
    Serial.begin(9600);
    scale.begin(DOUT, CLK);
    Serial.println("tekan a,s,d,f untuk menaikkan calibration_factor ke
10,100,1000,10000");
    Serial.println("tekan z,x,c,v untuk menurunkan calibration_factor ke
10,100,1000,10000");
    Serial.println("Tekan T untuk Tare");
    scale.set_scale();
    scale.tare();
    long zero_factor = scale.read_average();
    Serial.print("Zero factor: ");
    Serial.println(zero_factor);
    delay(1000);
}

void loop() {
    scale.set_scale(calibration_factor);
    GRAM = scale.get_units(), 4;
    Serial.print("Reading: ");
    Serial.print(GRAM);
    Serial.print(" Gram");
    Serial.print(" calibration_factor: ");
    Serial.print(calibration_factor);
    Serial.println();
}

```

```
if (Serial.available()) {
  char temp = Serial.read();
  if (temp == '+' || temp == 'a')
    calibration_factor += 0.1;
  else if (temp == '-' || temp == 'z')
    calibration_factor -= 0.1;
  else if (temp == 's')
    calibration_factor += 10;
  else if (temp == 'x')
    calibration_factor -= 10;
  else if (temp == 'd')
    calibration_factor += 100;
  else if (temp == 'c')
    calibration_factor -= 100;
  else if (temp == 'f')
    calibration_factor += 1000;
  else if (temp == 'v')
    calibration_factor -= 1000;
  else if (temp == 't')
    scale.tare();
}
}
```