



Membuat Aplikasi IoT (Internet of Things) dg IoT Starter Kit



MEMBUAT APLIKASI IoT dg IoT STARTER KIT

[Apa itu IoT ?](#)

[Contoh Aplikasi IoT](#)

[Sistem Pada IoT](#)

[Membuat aplikasi IoT dengan Nodemcu V3](#)

[Modul IoT Starter Kit Inkubatek](#)

[Project IoT 1 : Kontrol On Off LED dengan WiFi](#)

[Project IoT 2 : Kontrol 2 LED via Internet](#)

[Project IoT 3 : Kontrol 4 Lampu via Internet](#)

[Project IoT 4 : Monitoring Suhu dg Tampilan Grafik di Internet](#)

[Project IoT 5 : Monitoring Suhu Kelembaban di Thingspeak](#)

[Project IoT 6 : Kontrol LED dengan Android](#)

[Project IoT 7 : Mengirim Notifikasi ke Telegram](#)

[Project IoT 8 : Sistem Keamanan Rumah dg Sensor PIR & Telegram](#)

[Project IoT 9 : Kontrol Lampu via Telegram](#)

[Project IoT 10 : Membuat Aplikasi Adnroid Kontrol LED dg Blynk](#)

[Project IoT 11 : Kontrol 4 Relay dg Aplikasi Android](#)

[Project IoT 12 : Monitoring DHT11 dg Tampilan Grafis](#)

[Project IoT 13 : Mengirim Notifikasi ke Email](#)

[Project IoT 14 : Early Warning System Kebocoran Gas](#)

[Project IoT 15 : Kontrol LED RGB via Android](#)

[Project IoT 16 : Kontrol Buzzer via Android](#)

[Project IoT 17 : Monitoring Suhu LM35 Tampilan Chart di Android](#)

[Project IoT 18 : Monitoring Suhu kelembaban Tampilan Chart Android](#)

[Project IoT 19 : Smart Home dengan Android](#)

I. Apa itu IoT ?

IoT (*Internet of Things*) saat ini memang lagi marak dipakai di dunia IT. Lantas apakah arti dari IoT tersebut ? Berikut definisi IoT yang saya ambil dari Wikipedia : IoT merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari koneksi internet yang tersambung secara terus-menerus. Adapun kemampuan seperti berbagi data, *remote control*, dan sebagainya, termasuk juga pada benda di dunia nyata. Contohnya bahan pangan, elektronik, peralatan apa saja, termasuk benda hidup yang semuanya tersambung ke jaringan lokal dan global melalui sensor yang tertanam dan selalu aktif.

II. Contoh Aplikasi IoT

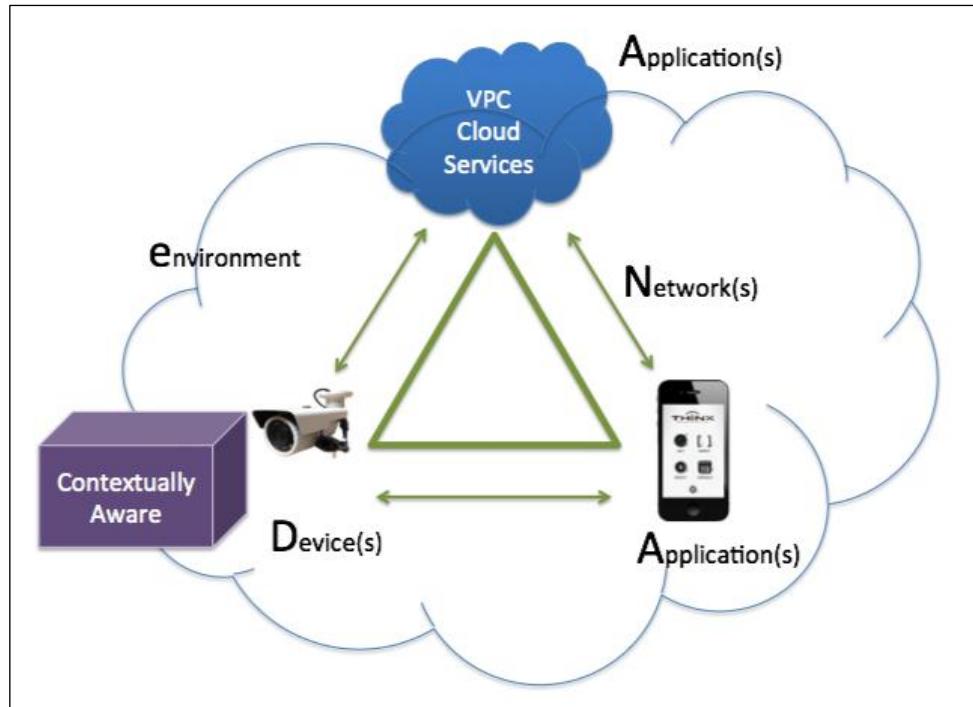
Banyak sekali contoh penerapan teknologi IoT, beberapa contohnya :

- Smart Home (system keamanan rumah berbasis internet, dapat mengetahui kedaan rumah serta mengontrol peralatan rumah tangga melalui jaringan internet)
- Internet Industri (monitoring dan controlling peralatan serta proses pada industry)
- Pertanian (monitoring dan pengendalian peralatan pertanian)
- Kesehatan (monitoring kondisi kesehataan seseorang)
- Transportasi (manajemen dan informasi lalu lintas)

III. Sistem pada IoT

Sistem dasar dari IoT terdiri dari 3 hal :

- Hardware/fisik (*Things*)
- Koneksi internet
- *Cloud* data center sebagai tempat untuk menyimpan atau menjalankan aplikasinya.



IV. Membuat Aplikasi IoT dengan Nodemcu V3

- **Konfigurasi Nodemcu V3**

NodeMCU adalah sebuah board elektronik yang berbasis chip ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (WiFi). Terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi monitoring maupun controlling pada proyek IoT.

Bentuk fisik dari modul NodeMCU V3 tampak pada gambar berikut :

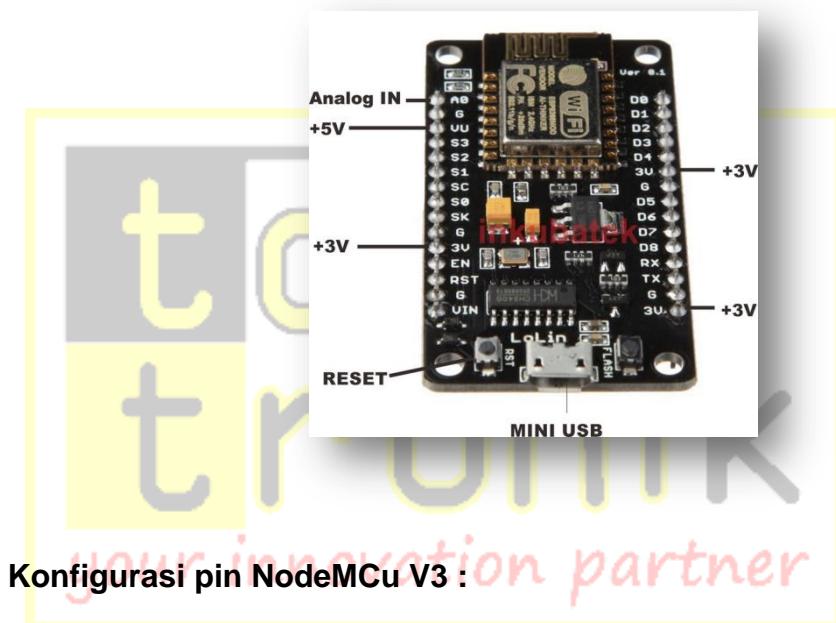


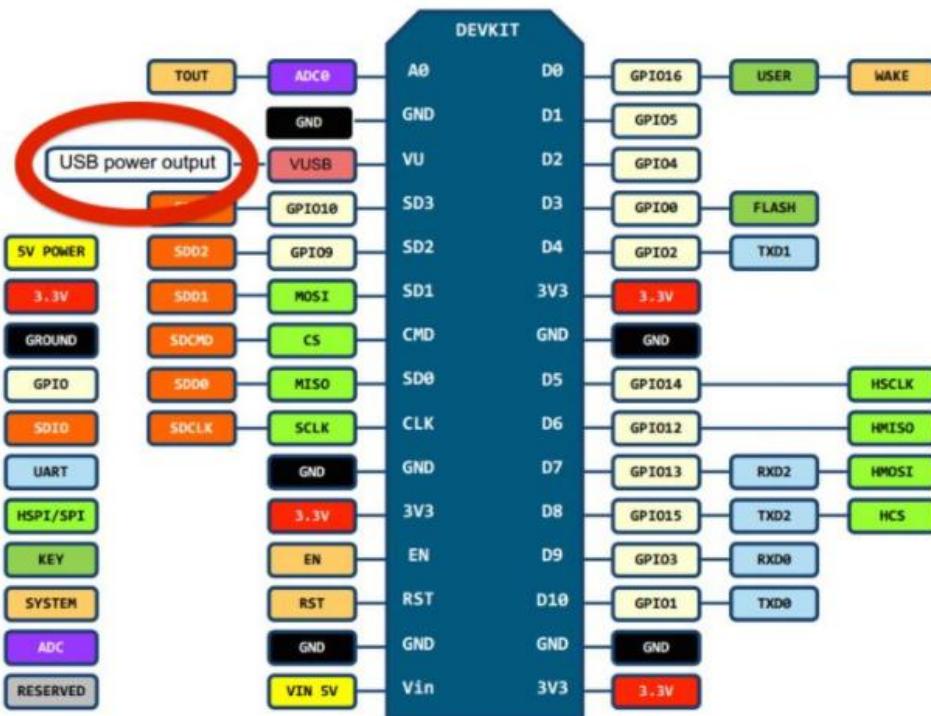
IoT Starter Kit Inkubatek

Perhatikan pada NodeMCU V3 ini terdapat port USB (mini USB) sehingga akan memudahkan dalam pemrogramannya nanti.

Spesifikasi :

- Tegangan kerja : 3,3 V
- Flash memori : 16 MB
- Terintegrasi dengan protocol TCP/IP
- Processor : Tensilica L106 32 bit
- Kecepatan : 80 – 160 Mhz
- Jumlah pin Digital I/O : 11 (D0 – D10)





- **Pemrograman NodeMCU V3**

NodeMCU V3 dapat diprogram dengan compiler-nya Arduino, menggunakan Arduino IDE. Tentu saja platform pemrogramannya memakai bahasa C.

Bagi anda yang belum pernah sama sekali memprogram Arduino, sebaiknya pelajari dahulu dasar – dasar pemrograman Arduino, karena nanti akan sangat menunjang pemahaman dan pengembangan aplikasi IoT dengan modul NodeMCU ini.

Secara sederhana, terdapat 2 fungsi di dalam setiap pemrogramannya :

- **Void init()**
- **Void loop()**

Penulisan huruf juga harus teliti, karena case sensitive, huruf besar berbeda dengan huruf kecil.

Ekspresi bilangan, logika pemrograman, operasi aritmatika sama dengan pemrograman Arduino. Banyak artikel sudah membahasnya, silakan anda pelajari terlebih dahulu.

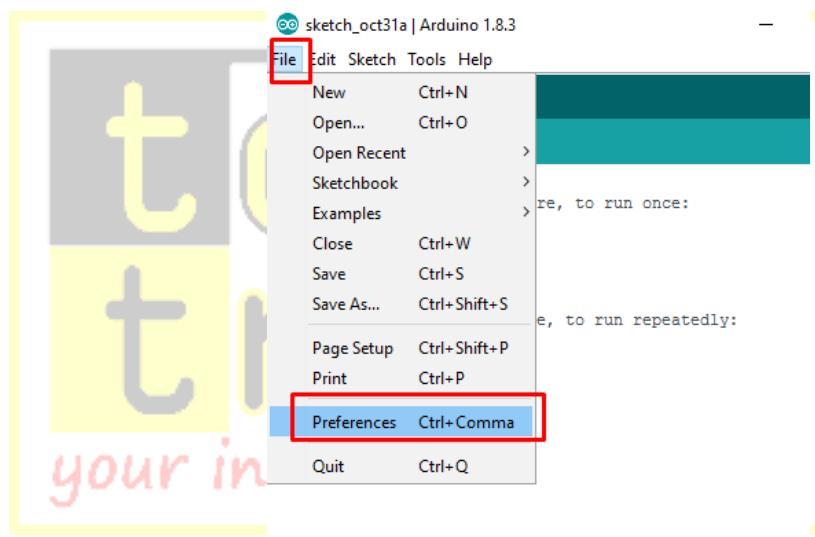
- **Instalasi Driver NodeMCU V3**

Board NodeMCU V3 support Win XP, Vista, Win 7/8/10. Prosedur instalasi silakan baca di CD.

- **Seting Arduino IDE untuk Nodemcu V3**

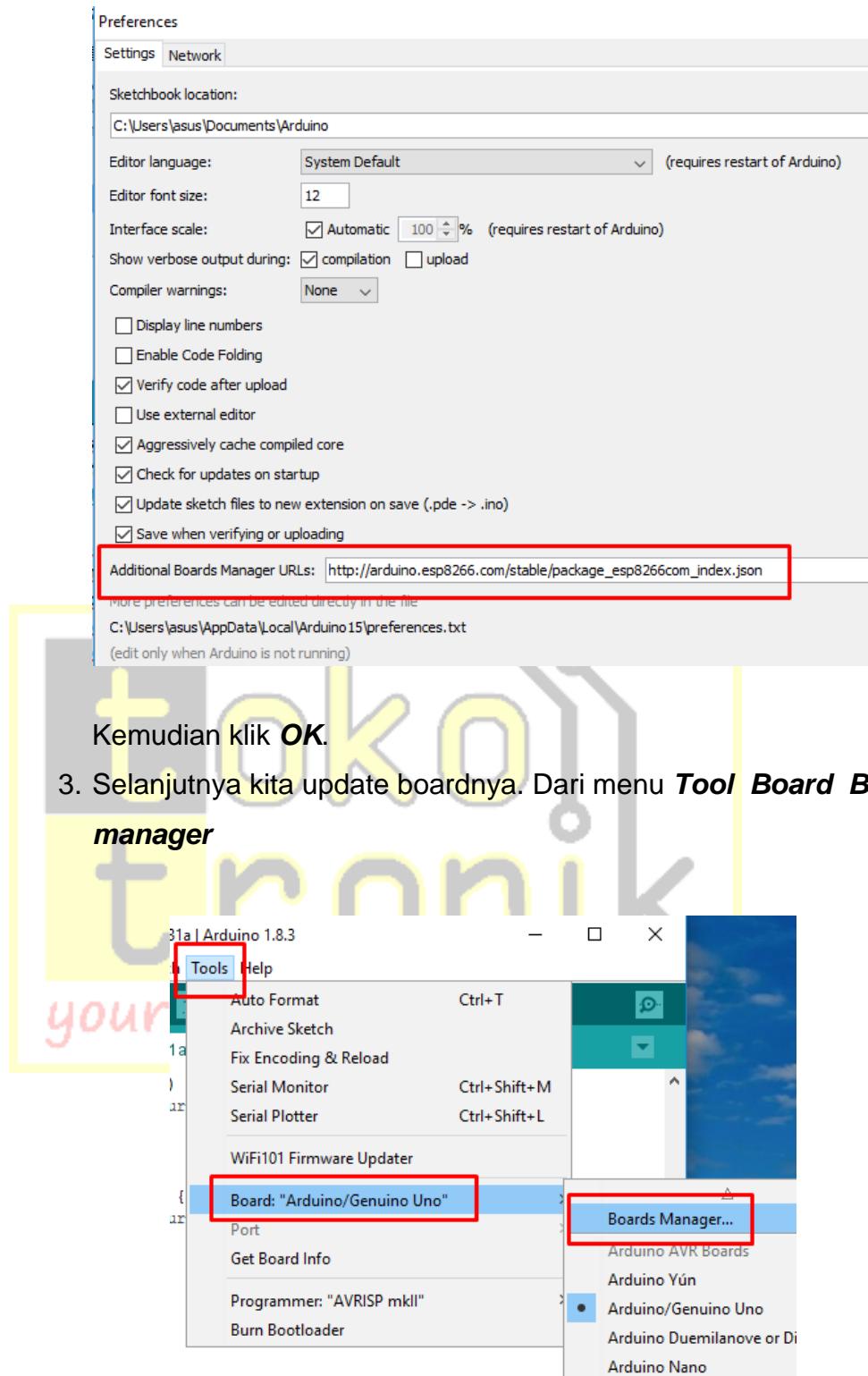
Untuk pemrogramannya kita akan memakai Arduino IDE, jika belum diseting (default program Arduino IDE belum terdapat board NodeMCU) maka perlu kita seting terlebih dahulu. File master Arduino IDE ada di CD. Cukup anda copy paste aja 1 folder ke folder computer anda. Buka foldernya dan cari file Arduino.exe. Klik 2x untuk menjalankannya. Selanjutnya kita seting Arduino IDE agar dapat dipakai untuk NodeMCu V3 kita.

1. Jalankan Arduino IDE. Dari menu **File Preference**.



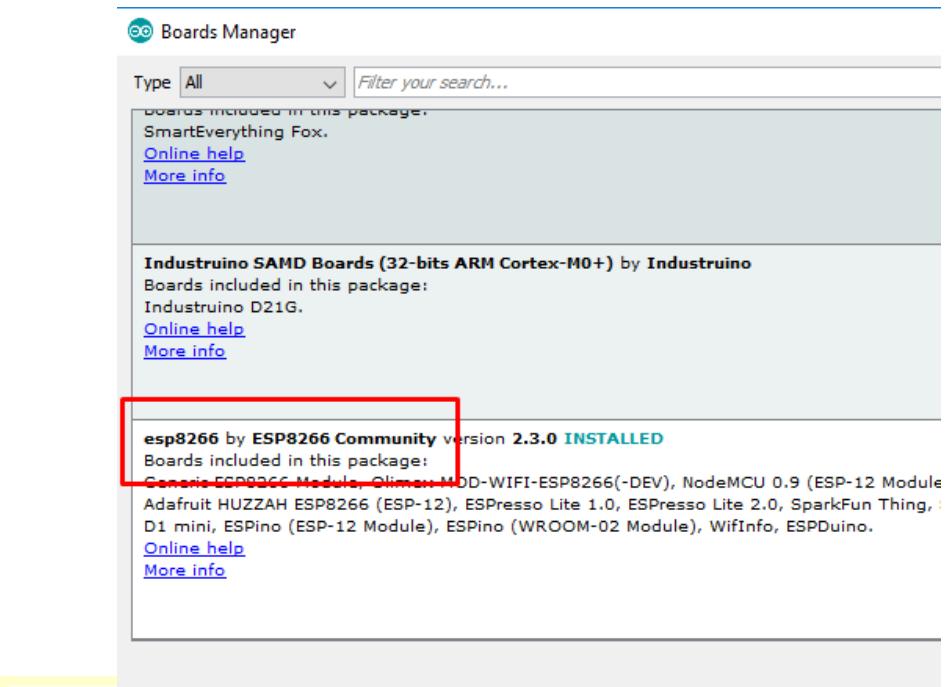
2. Pada jendela Preference, di bagian bawah terdapat kolom **Additional Boards Manager URLs** kemudian copy-paste link berikut :
http://arduino.esp8266.com/stable/package_esp8266com_index.json

IoT Starter Kit Inkubatek

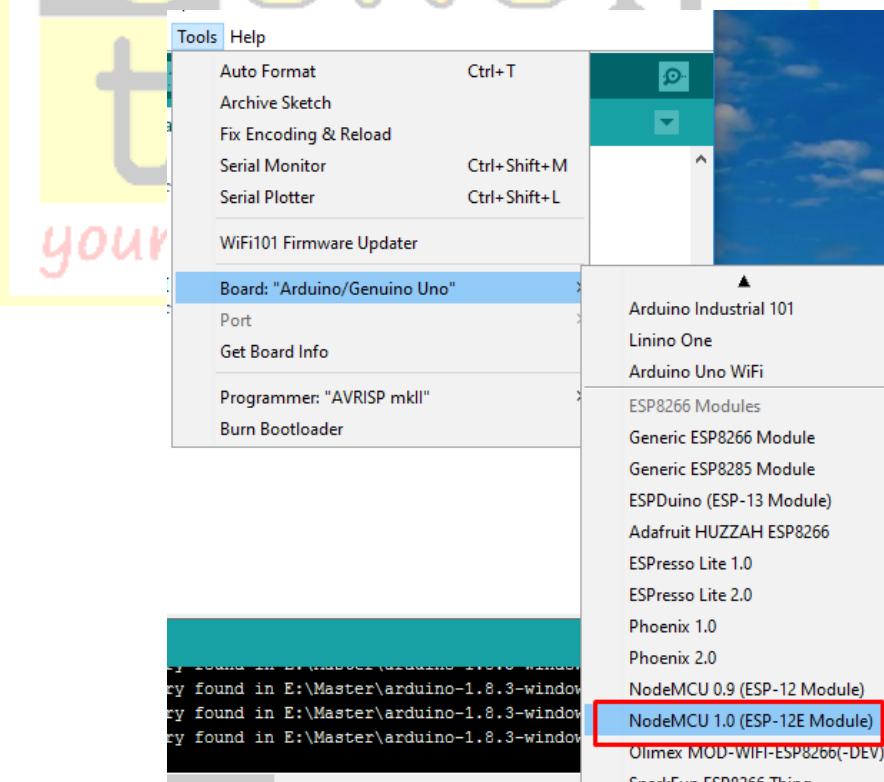


Akan muncul jendela **Boards Manager**. Pada bagian bawah cari esp8266 kemudian klik Install.

IoT Starter Kit Inkubatek



4. Ok, sekarang kita cek apakah NodeMCU sudah terinstal di Arduino IDE kita atau belum. Dari menu **Tools Board NodeMCU** seperti pada gambar.



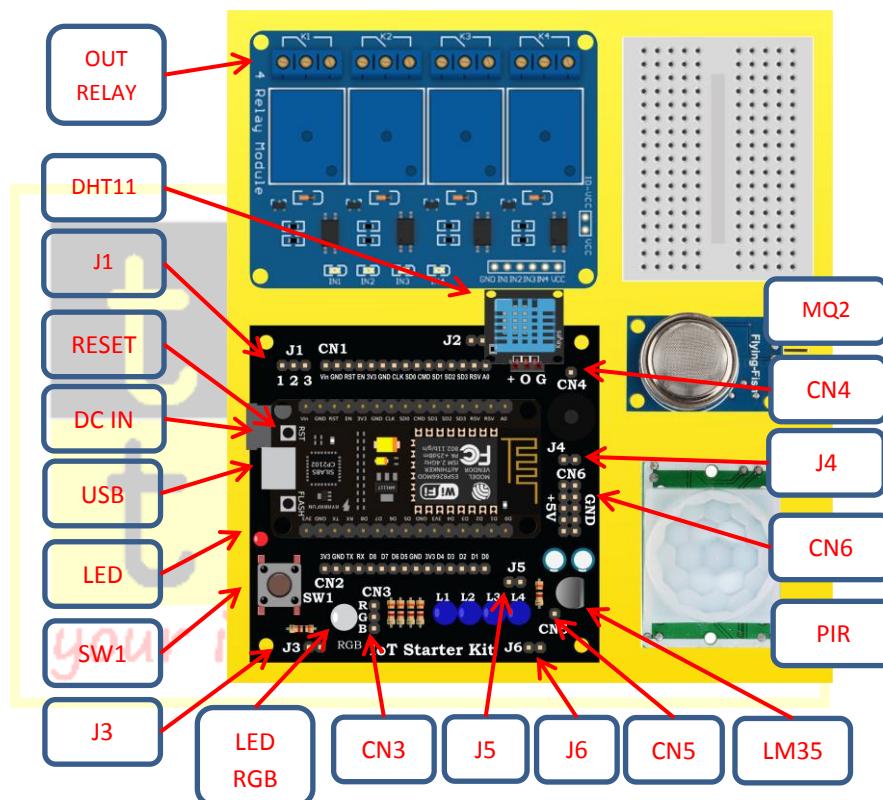
- Library Arduino untuk Aplikasi IoT

IoT Starter Kit Inkubatek

Sebelum kita menggunakan Arduino IDE untuk membuat program aplikasi IoT, beberapa library perlu kita tambahkan. File library dan cara menambahkan library silakan buka di CD.

V. Modul IoT Starter Kit Inkubatek

Modul **IoT Starter Kit** made in Inkubatek terdiri dari board NodeMCU V3 dan beberapa peripheral penunjang aplikasi IoT.



Keterangan :

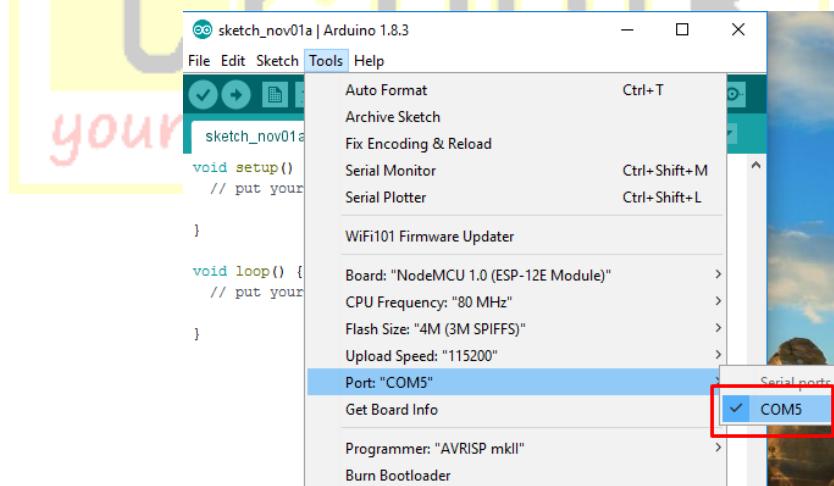
Nama	Keterangan
USB	Konektor micro USB
DC IN	Jack DC input 7-9 VDC
J1	Jika tegangan dari DC in maka J1 = 1+2 Jika tegangan dari USB computer maka J1=2+3
J2	ON = Sensor DHT11 aktif (dapat tegangan) OFF=Sensor DHT11 disable (tidak dapat tegangan)

J3	ON = SW1 (push button) aktif dan terhubung ke pin D2 (aktif HIGH)
J4	ON = Buzzer <i>enable</i> terhubung ke pin D1 (aktif HIGH)
J5	ON = LED 1 – LED 4 <i>enable</i> , terhubung ke pin D5 – D8 (aktif HIGH)
J6	ON = LM35 <i>enable</i> (dapat tegangan)
CN3	Koneksi LED RGB
CN4	Output sensor DHT11
CN5	Output sensor LM35

Tes modul Nodemcu V3

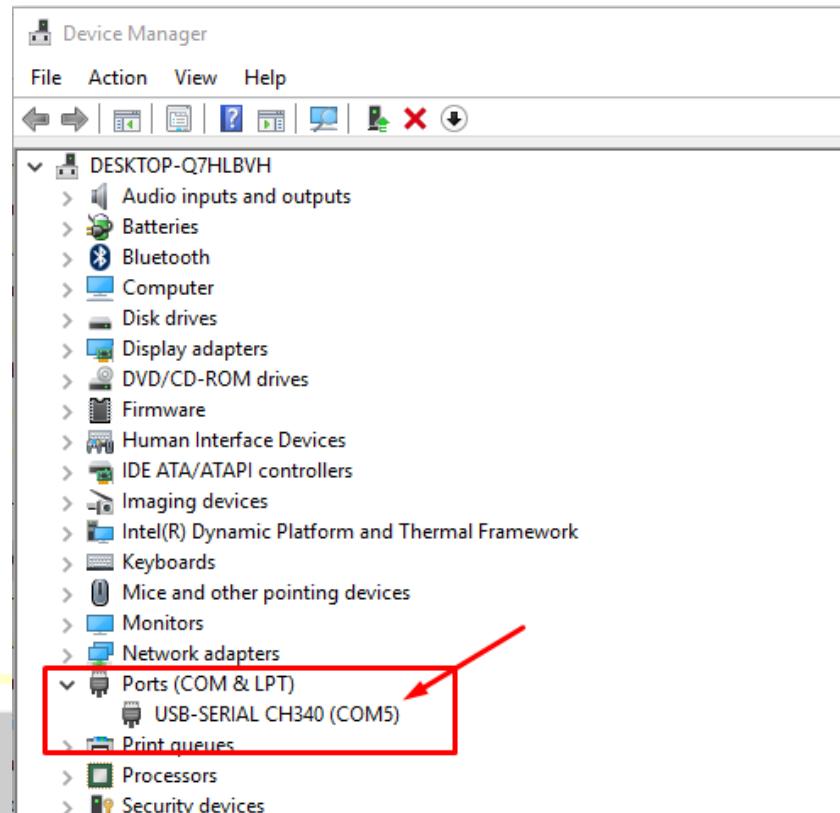
Sekarang saatnya kita coba modul NodeMCU. Koneksikan modul NodeMCU V3 ke port computer/laptop melalui kabel USB. Cara instalasi drivernya silakan lihat di CD.

Pilih board : NodeMCU 1.0 (ESP-12E Module). Sesuaikan nomor port dengan nomor com NodeMCU V3 yang terdeteksi di computer anda.

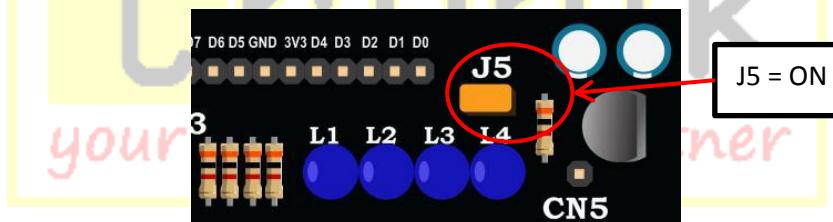


Untuk nomor port-nya silakan cek di *Device Manager* komputer anda.

IoT Starter Kit Inkubatek



Kita coba program pertama yaitu **LED Blink**. Set (pasang) jumper J5 di board IoT.



Tulis program berikut di sketch Arduino. (nama program : LED Blink).

```
#define LED_pin D5
void setup() {
    pinMode(LED_pin, OUTPUT);
}

void loop() {
    digitalWrite(LED_pin, LOW);
    delay(1000);
    digitalWrite(LED_pin, HIGH);
    delay(1000);
}
```

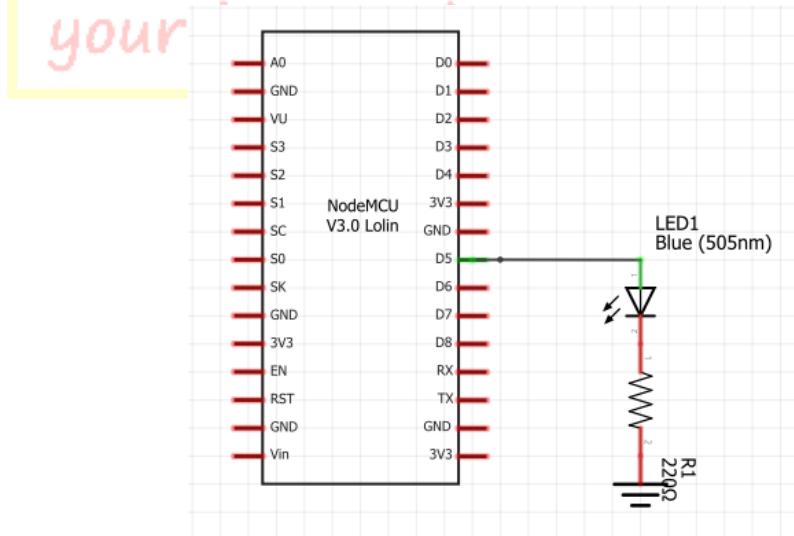
IoT Starter Kit Inkubatek

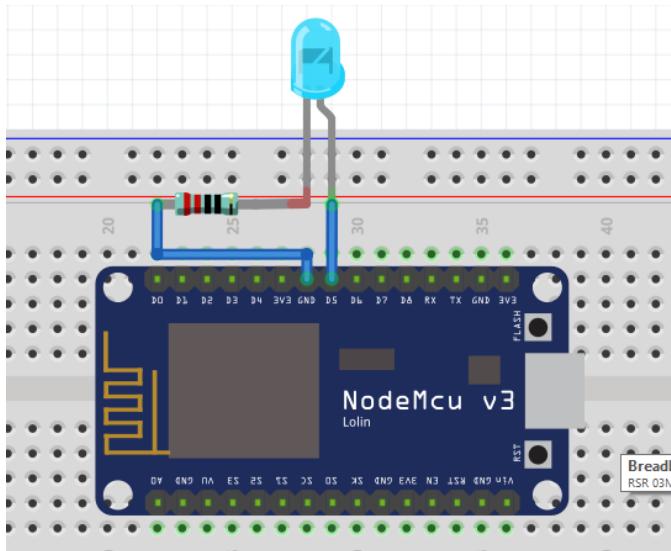
Simpan (**Save**) kemudian **Upload** ke board NodeMCU V3. Klik **Upload**.

```
LED_Blink | Arduino 1.8.3
File Edit Sketch Tools Help
LED_Blink
1. Save
2. Upload
* IoT Starter
* Tes modul dengan LED Blink
* www.inkubator-teknologi.com
* www.tokotronik.com
* 1 November 2017
* OK
*****
#define LED_pin 14
void setup() {
    pinMode(LED_pin, OUTPUT);
}

void loop() {
    digitalWrite(LED_pin, LOW);
    delay(1000);
    digitalWrite(LED_pin, HIGH);
    delay(1000);
}
< >
Done Saving.
```

Programnya sederhana, menghidupkan LED 1 yang ada di board berkedip (*blink*) dengan delay 1 detik. Rangkaian LED 1 tampak pada gambar berikut ini. LED akan nyala jika pada kaki anoda (pin D5) berlogika '*HIGH*'.





Program ini untuk testing apakah NodeMCU V3 anda dapat diprogram dan berjalan dengan Arduino IDE.

Ok sekarang kita **Upload** ke modul NodeMCU V3. Klik tombol **Upload**, tunggu sampai proses compiling s/d upload selesai.

Hasilnya tampak LED 1 pada board akan berkedip dengan delay 1 detik.

Jika belum berhasil silakan cek lagi programnya. Jika ketika proses **Upload** mengalami masalah coba cek lagi koneksi NodeMCU V3 ke port USB (pastikan tidak kendor) dan instalasi driver berhasil (cek di *Device Manager*)

Tes koneksi Nodemcu V3 dengan WiFi

Selanjutnya kita akan coba koneksi NodeMCU V3 ini dengan jaringan internet WiFi. Sebelumnya siapkan hotspot (WiFi) anda. Dapat memakai modem atau HP.

Catatan : nama jaringan WiFi dan passwordnya, nantinya kita gunakan untuk menulis program.

Misalnya :

- o **Nama : FirnasAbe**
- o **Password : podowingi**

Selanjutnya bikin programnya, sekali lagi ganti nama WiFi dan passwordnya sesuai dengan jaringan anda (nama program : Tes Koneksi WiFi).

```
*****
```

* Program : Tes Koneksi WiFi NodeMCU V3

* Input : -

* Output : Serial Monitor

* IoT Starter Kit Inkubatek

* www.tokotronik.com

* ******/

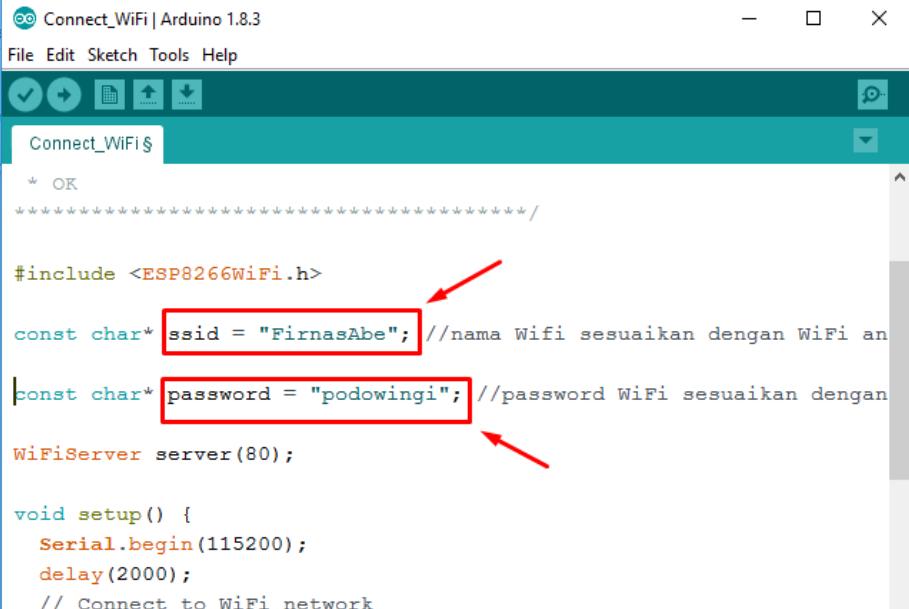
```
#include <ESP8266WiFi.h>
const char* ssid = "FirnasAbe"; //nama Wifi sesuaikan dengan WiFi anda
const char* password = "podowingi"; //password WiFi sesuaikan dengan WiFi anda
WiFiServer server(80);

void setup() {
    Serial.begin(115200);
    delay(2000);
    // Connect to WiFi network
    Serial.println();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
}

void loop() {
```

Jangan lupa untuk mengganti nama jaringan (ssid) dan password WiFi anda).

IoT Starter Kit Inkubatek



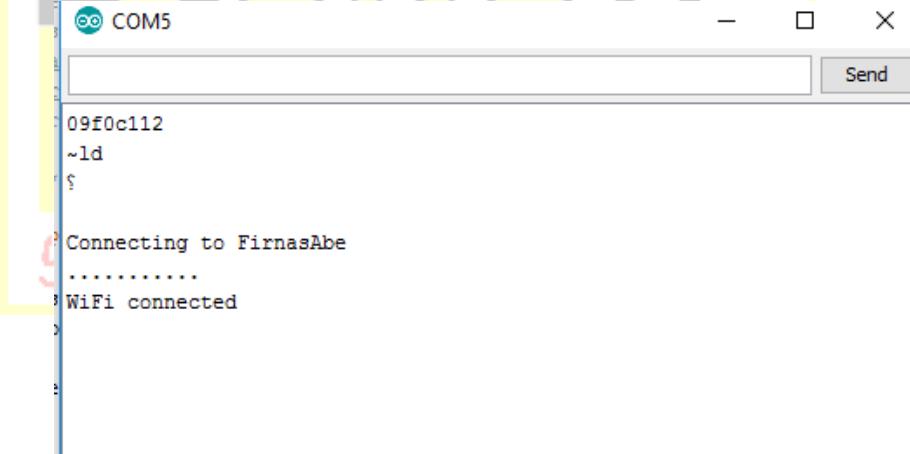
```
#include <ESP8266WiFi.h>

const char* ssid = "FirnasAbe"; //nama Wifi sesuaikan dengan WiFi an
const char* password = "podowinggi"; //password WiFi sesuaikan dengan

WiFiServer server(80);

void setup() {
  Serial.begin(115200);
  delay(2000);
  // Connect to WiFi network
```

Simpan (**Save**) kemudian **Upload**. Tunggu sampai selesai, setelah itu cek di Serial Monitor (**Tools Serial Monitor**) maka akan tampil status koneksi NodeMCU dengan WiFi. Seting Serial Monitor pada **Baudrate 115200**.



```
09f0c112
~ld
?

Connecting to FirnasAbe
.....
WiFi connected
>
:
```

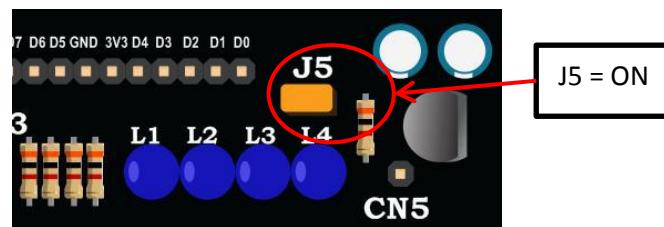
Jika tidak tampil, silakan anda reset dengan menekan tombol RST di NodeMCU.

VI. Project IoT 1 : Kontrol On Off LED dengan WiFi

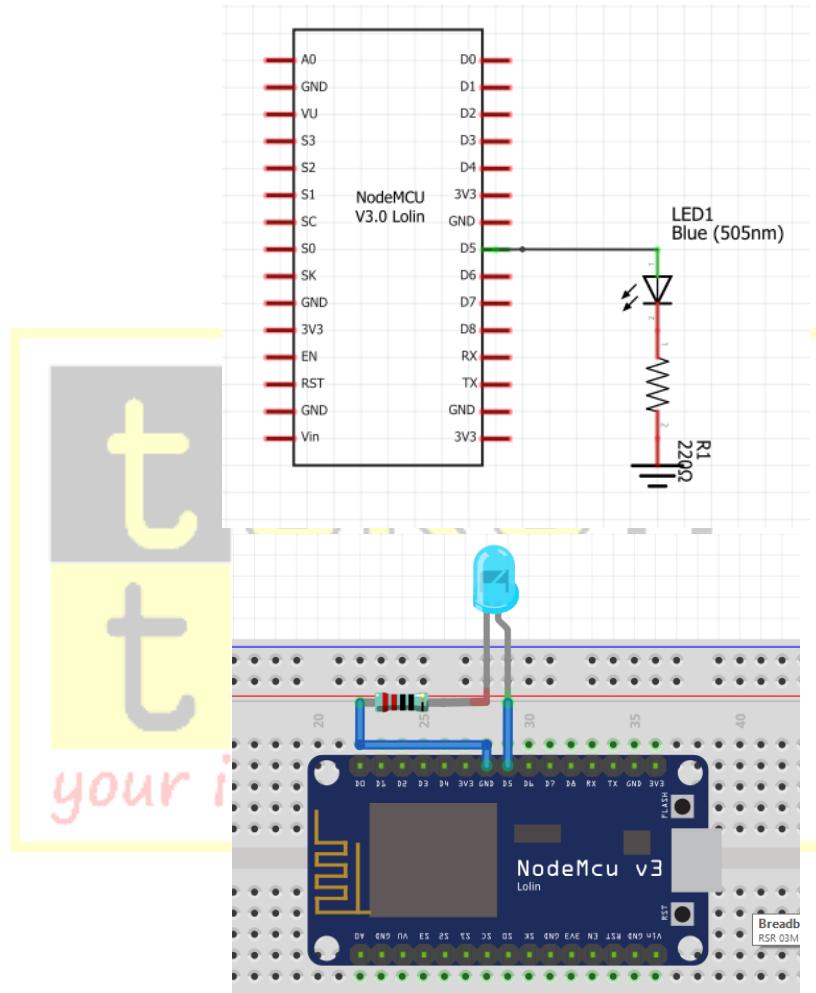
Pada proyek pertama ini kita akan membuat proyek IoT berupa kontrol On – Off LED melalui jaringan WiFi.

Pastikan jumper **J5** pada board **IoT Starter Kit** dalam posisi “ON” (terpasang).

IoT Starter Kit Inkubatek



Sehingga rangkaian pada board IoT Starter Kit menjadi seperti ini :



Buat program berikut : (anda juga dapat membuka langsung program yang ada di CD). Jangan lupa untuk mengganti SSID dan password WiFi nya.

```
*****
* Program : Project 1 Kontrol LED
* Input : -
* Output : LED1
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****/
```

```
#include <ESP8266WiFi.h>
const char* ssid = "FirnasAbe";
const char* password = "podowingi";

int ledPin = D5;
WiFiServer server(80);

void setup() {
    Serial.begin(115200);
    delay(10);
    pinMode(ledPin, OUTPUT);
    digitalWrite(ledPin, LOW);
    Serial.println();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    server.begin();
    Serial.println("Server started");
    Serial.print("Use this URL to connect: ");
    Serial.print("http://");
    Serial.print(WiFi.localIP());
    Serial.println("/");
}

void loop() {
    WiFiClient client = server.available();
    if (!client) {
        return;
    }

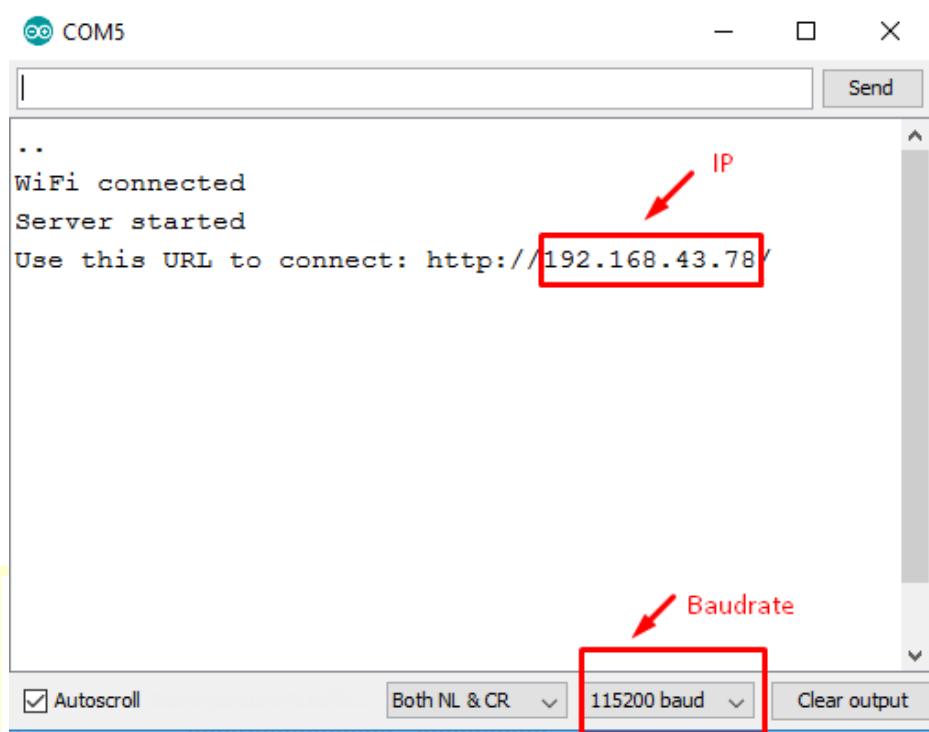
    Serial.println("new client");
    while(!client.available()){


```

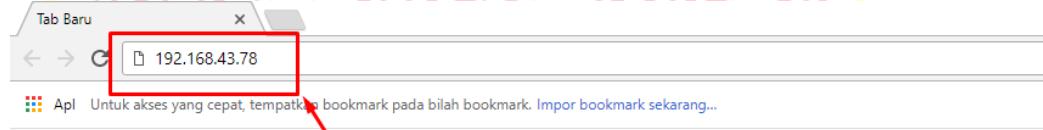
```
delay(1);  
}  
  
String request = client.readStringUntil('\r');  
Serial.println(request);  
client.flush();  
  
int value = LOW;  
if (request.indexOf("/LED=ON") != -1) {  
    digitalWrite(ledPin, HIGH);  
    value = HIGH;  
}  
if (request.indexOf("/LED=OFF") != -1) {  
    digitalWrite(ledPin, LOW);  
    value = LOW;  
}  
  
client.println("HTTP/1.1 200 OK");  
client.println("Content-Type: text/html");  
client.println(""); // do not forget this one  
client.println("<!DOCTYPE HTML>");  
client.println("<html>");  
  
client.print("Led pin is now: ");  
  
if(value == HIGH){  
    client.print("On");  
} else {  
    client.print("Off");  
}  
client.println("<br><br>");  
client.println("<a href=\"/LED=ON\"><button>Turn On </button></a>");  
client.println("<a href=\"/LED=OFF\"><button>Turn Off </button></a><br>");  
client.println("</html>");  
delay(1);  
Serial.println("Client disconnected");  
Serial.println("");  
}
```

IoT Starter Kit Inkubatek

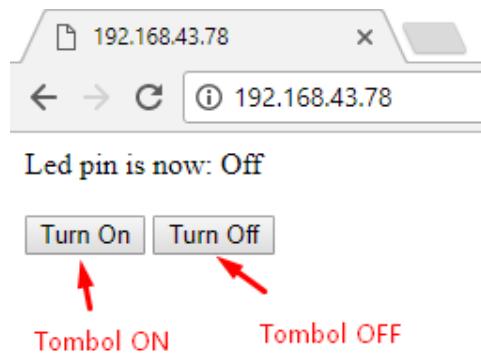
Setelah disimpan (**Save**) kemudian **Upload** (pastikan tidak ada eror). Buka Serial Monitor dengan Baud rate 115200.



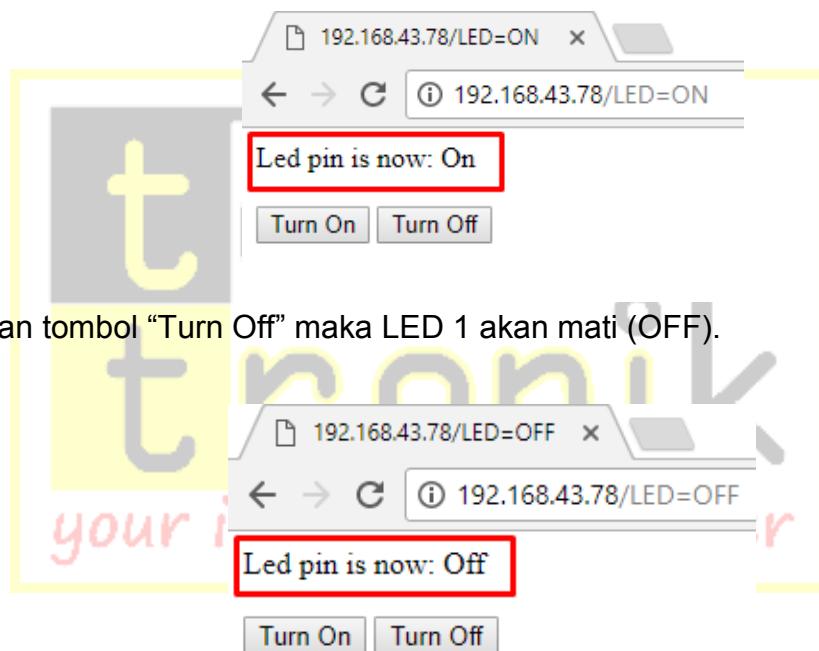
Setelah device terkoneksi dengan WiFi, pada Serial Monitor muncul nomor IP. Copy alamat IP nya, pada contoh diatas alamatnya : 192.168.43.78 kemudian paste di web browser anda.



Kemudian enter maka akan muncul tampilan seperti ini :



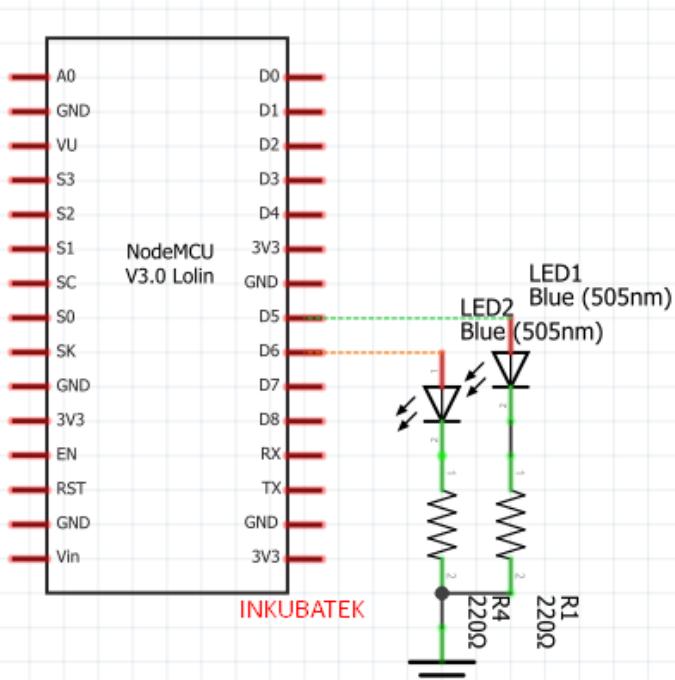
Tekan tombol “Turn On” kemudian perhatikan LED 1, jika berjalan normal maka LED 1 akan nyala (ON).



VII. Project IoT 2 : Kontrol 2 LED

Selanjutnya kita akan membuat aplikasi IoT untuk mengontrol 2 LED. Jumper J5 tetap dalam keadaan terpasang ya ..

IoT Starter Kit Inkubatek



Kita buat programnya. Jangan lupa untuk mengganti SSID dan password WiFi nya.

```
*****
* Program : Project 2: Kontrol 2 LED
* Input : -
* Output : LED1 + LED 2
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****
```

```
#include <ESP8266WiFi.h>
const char* ssid = "FirnasAbe";
const char* password = "podowingi";
#define LED1 D5
#define LED2 D6

WiFiServer server(80);

void setup() {
  Serial.begin(115200);
  delay(10);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
```

```
digitalWrite(LED1, LOW);
digitalWrite(LED2, LOW);

// Connect to WiFi network
Serial.println();
Serial.println();
Serial.print("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");

server.begin();
Serial.println("Server started");

Serial.print("Use this URL to connect: ");
Serial.print("http://");
Serial.print(WiFi.localIP());
Serial.println("/");
}

void loop() {
    WiFiClient client = server.available();
    if (!client) {
        return;
    }

    Serial.println("new client");
    while(!client.available()){
        delay(1);
    }

    String request = client.readStringUntil('\r');
    Serial.println(request);
    client.flush();
```

```
if (request.indexOf("/led1on") > 0) {
    digitalWrite(LED1, HIGH);

}

if (request.indexOf("/led1off") > 0) {
    digitalWrite(LED1, LOW);
}

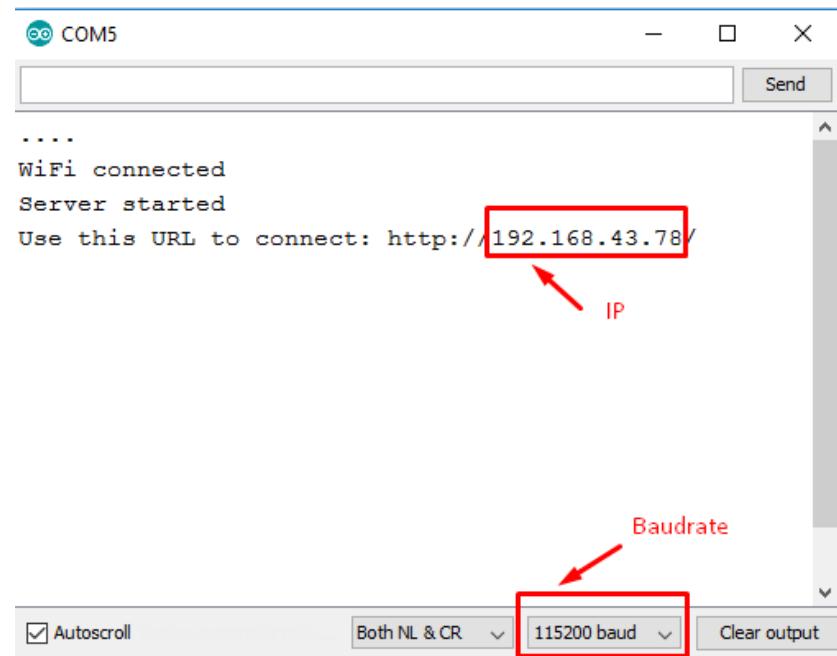
if (request.indexOf("/led2on") > 0) {
    digitalWrite(LED2, HIGH);
}

if (request.indexOf("/led2off") > 0) {
    digitalWrite(LED2, LOW);
}

// Return the response
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("");
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.println("<head>");
client.println("<meta name='apple-mobile-web-app-capable' content='yes' />");
client.println("<meta name='apple-mobile-web-app-status-bar-style' content='black-translucent' />");
client.println("</head>");
client.println("<body bgcolor = '#f7e6ec'>");
client.println("<hr/><hr>");
client.println("<h4><center> IoT Starter Kit : Kontrol 2 LED </center></h4>");
client.println("<hr/><hr>");
client.println("<br><br>");
client.println("<br><br>");
client.println("<center>");
client.println("LED 1");
client.println("<a href=\"/led1on\"><button>Turn On </button></a>");
client.println("<a href=\"/led1off\"><button>Turn Off </button></a><br />");
client.println("</center>");
client.println("<br><br>");
client.println("<center>");
client.println("LED 2");
```

```
client.println("<a href=\"/led2on\"><button>Turn On </button></a>");  
client.println("<a href=\"/led2off\"><button>Turn Off </button></a><br />");  
client.println("</center>");  
client.println("<br><br>");  
client.println("<center>");  
client.println("<table border=\"5\">");  
client.println("<tr>");  
//=====  
if (digitalRead(LED1))  
{  
    client.print("<td>LED 1 = ON</td>");  
}  
else  
{  
    client.print("<td>LED 1 = OFF</td>");  
}  
client.println("<br />");  
//=====  
if (digitalRead(LED2))  
{  
    client.print("<td>LED 2 = ON</td>");  
}  
else  
{  
    client.print("<td>LED 2 = OFF</td>");  
}  
client.println("</tr>");  
//=====  
client.println("<tr>");  
client.println("</table>");  
client.println("</center>");  
client.println("</html>");  
delay(1);  
Serial.println("Client disconnected");  
Serial.println("");  
}
```

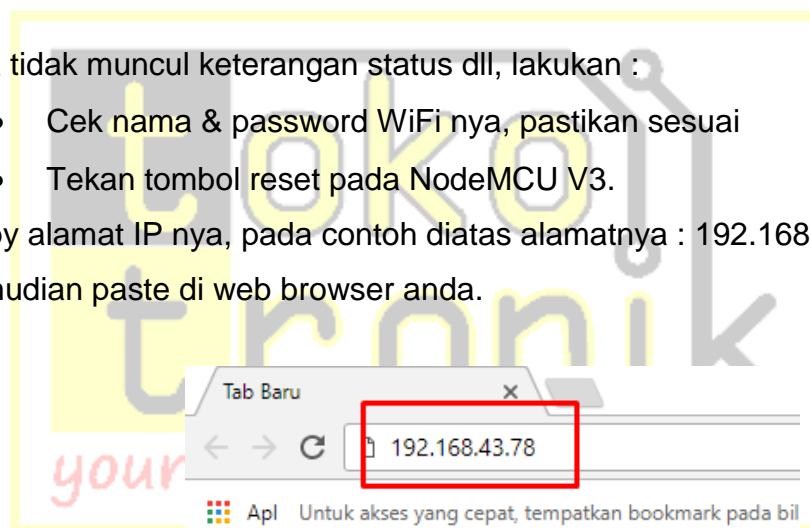
Save kemudian **Upload**. Buka Serial Monitor (**Tools Serial Monitor**) setting baud rate pada nilai 115200. Maka akan muncul tampilan status koneksi dengan WiFi dan nomor IP jaringannya.



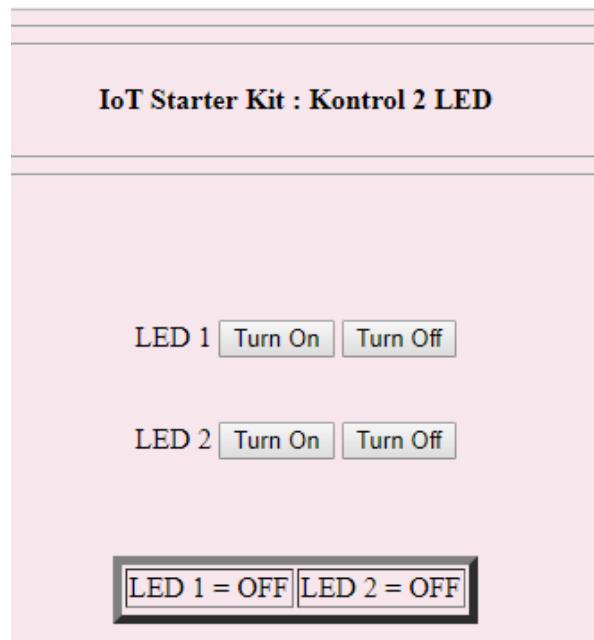
Jika tidak muncul keterangan status dll, lakukan :

- Cek nama & password WiFi nya, pastikan sesuai
- Tekan tombol reset pada NodeMCU V3.

Copy alamat IP nya, pada contoh diatas alamatnya : 192.168.43.78
kemudian paste di web browser anda.



Kemudian ENTER maka akan tampil :

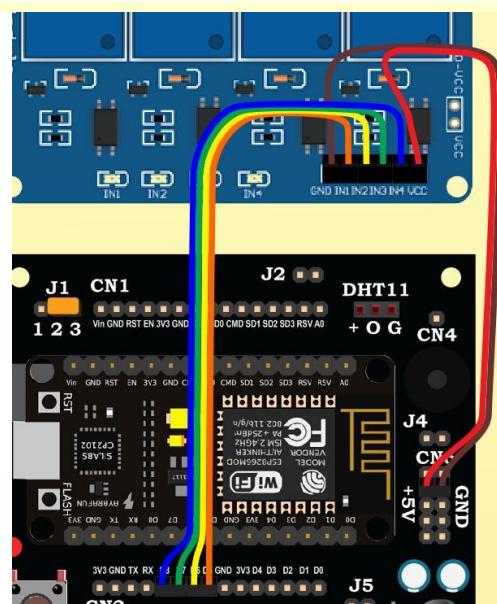


Tekan tombol Turn On pada LED 1 dan perhatikan LED 1 akan ON, demikian juga pada LED 2.

VIII. Project IoT 3 : Kontrol 4 Lampu

Pada proyek IoT ke-3 ini kita akan mengontrol lampu AC sebanyak 4 buah. Untuk *interfacingnya* kita pakai relay yang sudah ada di board **IoT Starter Kit** Inkubatek

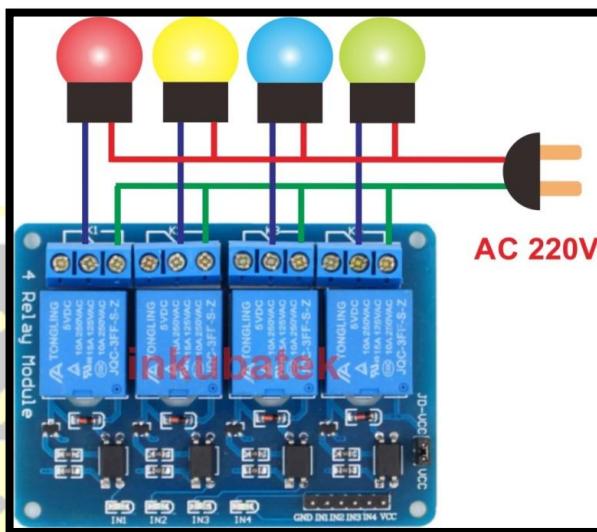
Buatlah rangkaian modul relay dengan board IoT Starter Kit seperti pada rangkaian.



IoT Starter Kit Inkubatek

Relay modul	Board IoT Starter Kit
IN1	D5
IN2	D6
IN3	D7
IN4	D8
VCC	+5V
GND	GND

Siapkan 4 buah lampu AC beserta kabel dan stecker (jack), kemudian buatlah rangkaian seperti ini :



Hati – hati dan perhatikan dengan teliti karena kita akan berhubungan dengan tegangan AC 220V. Anda dapat juga menggantinya dengan lampu DC. Selanjutnya buatlah program berikut, jangan lupa untuk mengganti **SSID** dan **password WiFi** nya.

```
*****  
* Program : Project 3 : Kontrol 4 Lampu  
* Input : -  
* Output : Relay 4 unit.  
* IoT Starter Kit Inkubatek  
* www.tokotronik.com  
*****/  
#include <ESP8266WiFi.h>  
const char* ssid = "FirnasAbe";
```

```
const char* password = "podowingi";  
  
#define Relay1 D5  
  
#define Relay2 D6  
  
#define Relay3 D7  
  
#define Relay4 D8  
  
;//  
  
WiFiServer server(80);  
  
void setup() {  
    Serial.begin(115200);  
    delay(10);  
    pinMode(Relay1, OUTPUT);  
    pinMode(Relay2, OUTPUT);  
    pinMode(Relay3, OUTPUT);  
    pinMode(Relay4, OUTPUT);  
    digitalWrite(Relay1, LOW);  
    digitalWrite(Relay2, LOW);  
    digitalWrite(Relay3, LOW);  
    digitalWrite(Relay4, LOW);  
    Serial.println();  
    Serial.println();  
    Serial.print("Connecting to ");  
    Serial.println(ssid);  
  
    WiFi.begin(ssid, password);  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(500);  
        Serial.print(".");  
    }  
    Serial.println("");  
    Serial.println("WiFi connected");  
    server.begin();  
    Serial.println("Server started");
```

```
Serial.print("Use this URL to connect: ");
Serial.print("http://");
Serial.print(WiFi.localIP());
Serial.println("/");
}

void loop() {
    WiFiClient client = server.available();
    if (!client) {
        return;
    }
    Serial.println("new client");
    while(!client.available()){
        delay(1);
    }
    String request = client.readStringUntil('\r');
    Serial.println(request);
    client.flush();
    if (request.indexOf("/relay1on") > 0) {
        digitalWrite(Relay1, HIGH);
    }
    if (request.indexOf("/relay1off") > 0) {
        digitalWrite(Relay1, LOW);
    }

    if (request.indexOf("/relay2on") > 0) {
        digitalWrite(Relay2, HIGH);
    }

    if (request.indexOf("/relay2off") > 0) {
        digitalWrite(Relay2, LOW);
    }
}
```

```
}

if (request.indexOf("/relay3on") > 0) {
    digitalWrite(Relay3, HIGH);

}

if (request.indexOf("/relay3off") > 0) {
    digitalWrite(Relay3, LOW);

}

if (request.indexOf("/relay4on") > 0) {
    digitalWrite(Relay4, HIGH);

}

if (request.indexOf("/relay4off") > 0) {
    digitalWrite(Relay4, LOW);

}

client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println(""); // do not forget this one
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.println("<head>");
client.println("<meta name='apple-mobile-web-app-capable' content='yes' />");
client.println("<meta name='apple-mobile-web-app-status-bar-style' content='black-translucent' />");
client.println("</head>");
client.println("<body bgcolor = '#f7e6ec'>");
client.println("<hr/><hr>");
```

```
client.println("<h4><center> IoT Starter Kit : Kontrol 4 Lampu</center></h4>");

client.println("<h4><center> www.tokotronik.com </center></h4>");
client.println("<hr><hr>");
client.println("<br><br>");
client.println("<br><br>");
client.println("<center>");

client.println("Lampu 1");
client.println("<a href=\"/relay1on\"><button>Turn On </button></a>");

client.println("<a href=\"/relay1off\"><button>Turn Off </button></a><br/>");
client.println("</center>");

client.println("<br><br>");
client.println("<center>");

client.println("Lampu 2");
client.println("<a href=\"/relay2on\"><button>Turn On </button></a>");

client.println("<a href=\"/relay2off\"><button>Turn Off </button></a><br/>");
client.println("</center>");

client.println("<br><br>");
client.println("<center>");

client.println("Lampu 3");
client.println("<a href=\"/relay3on\"><button>Turn On </button></a>");

client.println("<a href=\"/relay3off\"><button>Turn Off </button></a><br/>");
client.println("</center>");

client.println("<br><br>");
client.println("<center>");

client.println("Lampu 4");
client.println("<a href=\"/relay4on\"><button>Turn On </button></a>");

client.println("<a href=\"/relay4off\"><button>Turn Off </button></a><br/>");
```

```
client.println("</center>");  
client.println("<br><br>");  
client.println("<center>");  
client.println("<table border=\"5\">");  
client.println("<tr>");  
=====  
if (digitalRead(Relay1))  
{  
    client.print("<td>Lampu 1 = ON</td>");  
}  
else  
{  
    client.print("<td>Lampu 1 = OFF</td>");  
}  
client.println("<br />");  
=====  
if (digitalRead(Relay2))  
{  
    client.print("<td>Lampu 2 = ON</td>");  
}  
else  
{  
    client.print("<td>Lampu 2 = OFF</td>");  
}  
client.println("</tr>");  
=====  
client.println("<tr>");  
  
if (digitalRead(Relay3))  
{  
    client.print("<td>Lampu 3 = ON</td>");  
}
```

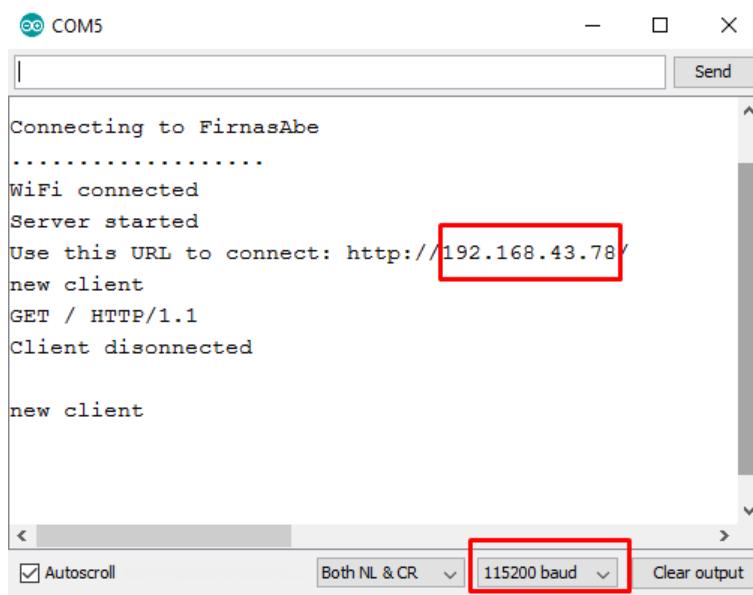
```
else
{
    client.print("<td>Lampu 3 = OFF</td>");
}

if(digitalRead(Relay4))
{
    client.print("<td>Lampu 4 = ON</td>");
}
else
{
    client.print("<td>Lampu 4 = OFF</td>");
}

client.println("</tr>");
client.println("</table>");
client.println("</center>");
client.println("</html>");
delay(1);
Serial.println("Client disconnected");
Serial.println("");
}
```

Save kemudian **Upload**. Buka Serial Monitor (**Tools Serial Monitor**) seting baud rate pada nilai 115200. Maka akan muncul tampilan status koneksi dengan WiFi dan nomor IP jaringannya.

IoT Starter Kit Inkubatek



The screenshot shows the Arduino Serial Monitor window titled "COM5". The text output is as follows:

```
Connecting to FirnasAbe
.....
WiFi connected
Server started
Use this URL to connect: http://192.168.43.78
new client
GET / HTTP/1.1
Client disconnected

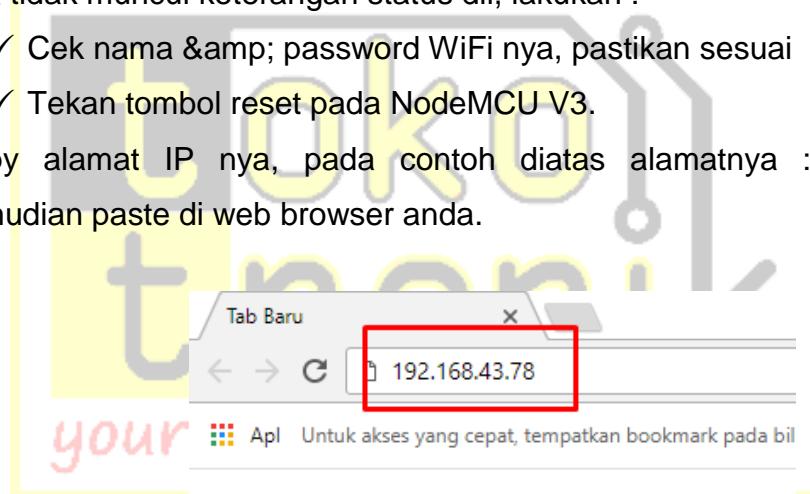
new client
```

The IP address "192.168.43.78" is highlighted with a red box. The baud rate is set to "115200 baud", which is also highlighted with a red box.

Jika tidak muncul keterangan status dll, lakukan :

- ✓ Cek nama & password WiFi nya, pastikan sesuai
- ✓ Tekan tombol reset pada NodeMCU V3.

Copy alamat IP nya, pada contoh diatas alamatnya : 192.168.43.78
kemudian paste di web browser anda.



Kemudian ENTER maka akan tampil :

IoT Starter Kit Inkubatek



Tekan tombol Turn On atau Turn Off pada masing – masing lampu maka lampu yang bersesuaian akan nyala atau padam.

IX. Project IoT 4 : Monitoring Suhu dengan Tampilan Grafik di Internet

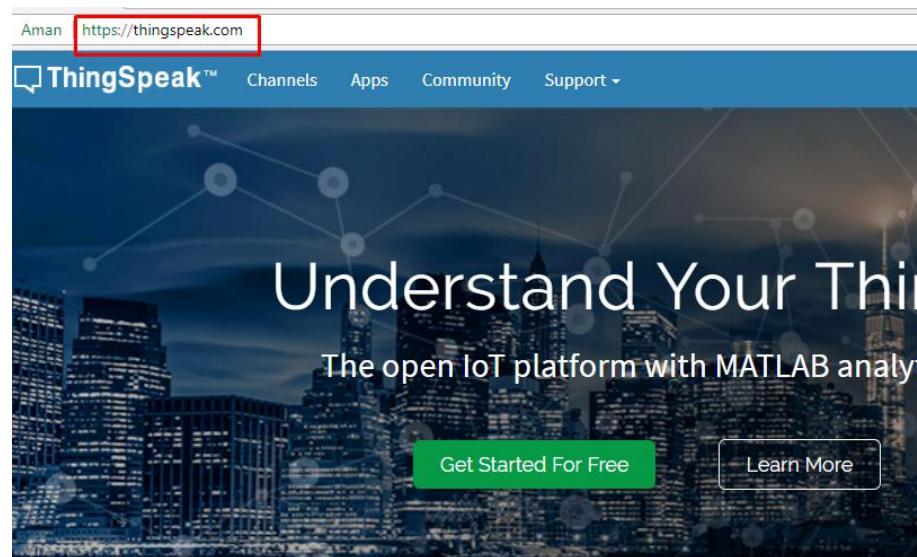
Pada proyek ke-4 IoT Starter Kit Inkubatek ini kita akan membaca nilai temperatur (suhu) lingkungan yang dibaca sensor LM35 kemudian menampilkan hasilnya di sebuah web server dalam bentuk grafik.

Kita akan menggunakan server **ThingSpeak**, jika belum mempunyai akun di **ThingSpeak** silakan mendaftar dulu.

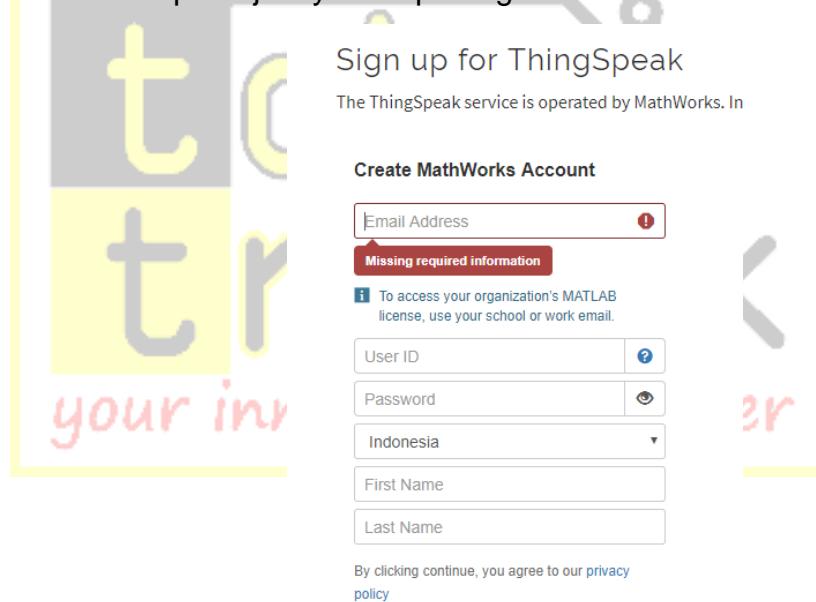


- Buka : <https://thingspeak.com>

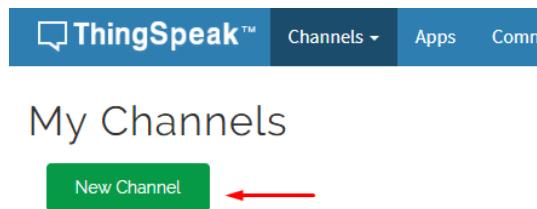
IoT Starter Kit Inkubatek



- Daftarkan akun anda. Klik “**Sign Up**” yang ada di pojok kanan atas.
- Siapkan sebuah akun email kemudian isi data – data yang diperlukan serta ikuti petunjuknya sampai registrasi berhasil.



- Selanjutnya kita buat channel. Klik “**New Channel**”

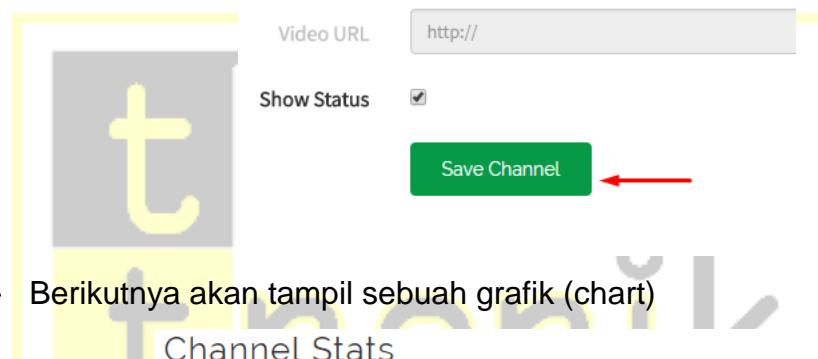


- Berikutnya isi kolom **Name, Description dan Field 1**.

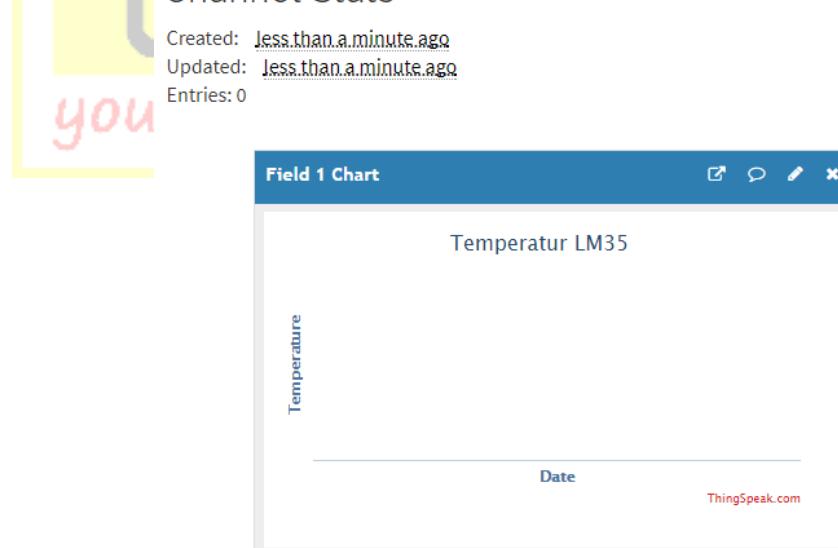
New Channel

Name	Temperatur LM35
Description	Monitoring Suhu LM35
Field 1	Temperature <input checked="" type="checkbox"/>
Field 2	<input type="checkbox"/>
Field 3	<input type="checkbox"/>
Field 4	<input type="checkbox"/>
Field 5	<input type="checkbox"/>

- OK, selanjutnya klik “**Save Channel**”

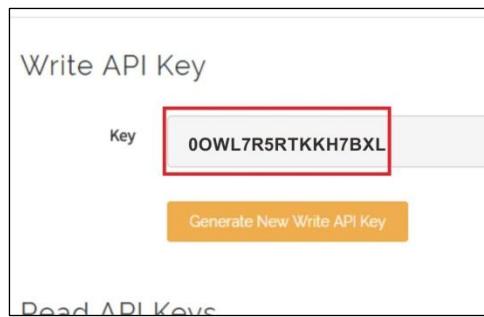


- Berikutnya akan tampil sebuah grafik (chart)



- Langkah selanjutnya kita cek API , klik tab **API Keys**.

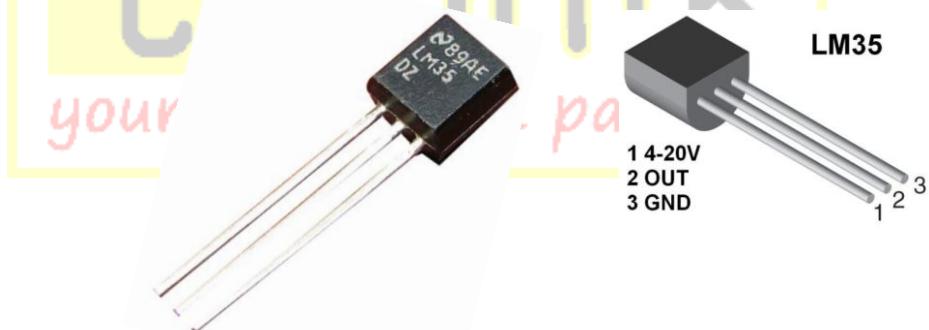
IoT Starter Kit Inkubatek



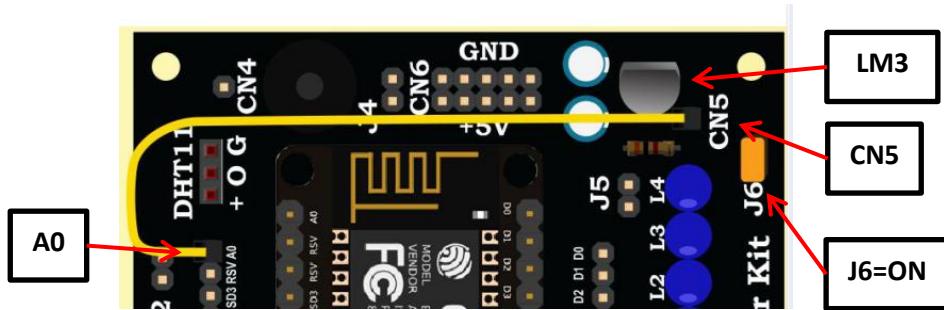
- Perhatikan dan catat kode API Keys-nya , nanti kita pakai untuk membuat programnya.
- Untuk pembuatan akun *Thingspeak* sampai disini dulu ya.. kita lanjutkan ke hardware-nya.

Siapkan sebuah sensor suhu LM35 (sudah ada di board IoT Starter Kit Inkubatek).

Sensor suhu LM35 sudah sangat umum dipakai untuk membaca temperatur lingkungan, mulai dari 0° – 100° Celcius. Rangkaianya juga mudah, tinggal beri tegangan 5VDC dan keluaran sensor akan liner dengan persamaan $10\text{mV}/1^{\circ}\text{C}$.



Pasang jumper “**J6**” agar tegangan LM35 terhubung ke 5V dan hubungkan **CN5** ke A0 dari NodeMCU V3 (output LM35 terhubung ke A0).



Selanjutnya kita siapkan programnya, buka lembar kerja baru (**File New**) di Arduino IDE, kemudian buat source code berikut (di CD juga sudah ada, tinggal anda buka dan sesuaikan WiFi-nya serta API Keys-nya *Thingspeak*.

```
/****************************************************************************
 * Program :Project 4 Monitoring Suhu Thingspeak
 * Input : LM35 di Ao
 * Output : Grafik Thingspeak
 * IoT Starter Kit Inkubatek
 * www.tokotronik.com
 * *****/
int outputpin= Ao;
int analogValue;
float millivolts,celsius;
#include <ESP8266WiFi.h>

// ganti dengan API Keys anda di Thingspeak
String apiKey = "oOWL7R5RTKKH7BXL";
const char* ssid = "FirnasAbe";
const char* password = "podowingi";
const char* server = "api.thingspeak.com";
WiFiClient client;
```

```
//=====
void setup() {
    Serial.begin(115200);
    delay(10);
    WiFi.begin(ssid, password);
    Serial.println();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
}

//=====

void loop() {
    analogValue = analogRead(outputpin);
    millivolts = (analogValue/1024.0) * 3300;
    celsius = millivolts/10;
    if (client.connect(server,80)) {
        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(celsius);
        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.print("Temperature: ");
Serial.print(celsius);
Serial.print(" Celcius ");
Serial.println(" send to Thingspeak");
}

client.stop();
Serial.println("Waiting...");
delay(20000); //minimal nunggu 15 detik update Thingspeak
}
```

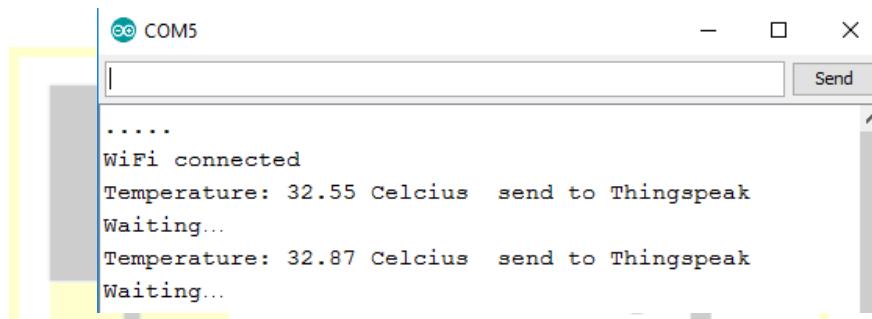
Sesuaikan beberapa variable berikut :

- ✓ **API Keys** Cek di akun Thingspeak, copy paste ke program untuk variable apiKey
- ✓ **Nama jaringan WiFi** , isikan ke ssid
- ✓ **Password WiFi** , isikan ke password

IoT Starter Kit Inkubatek

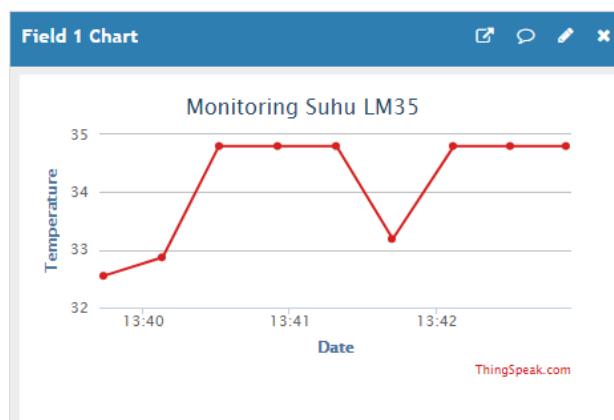


Save kemudian **Upload**. Pastikan tidak ada error. Buka Serial Monitor di Arduino IDE (**Tools Serial Monitor**) maka akan tampil :



Buka web **thingspeak.com** dan masuk ke akun anda dengan user name + password yang sudah dibuat.

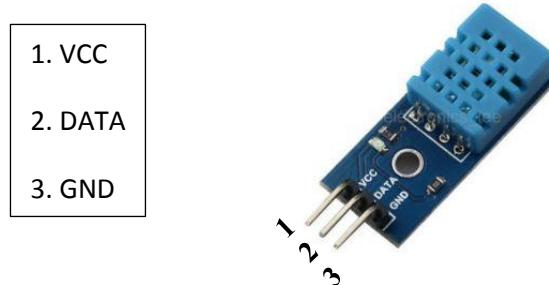
Perhatikan tampilan grafik akan sesuai dengan nilai yang dibaca oleh sensor suhu LM35.



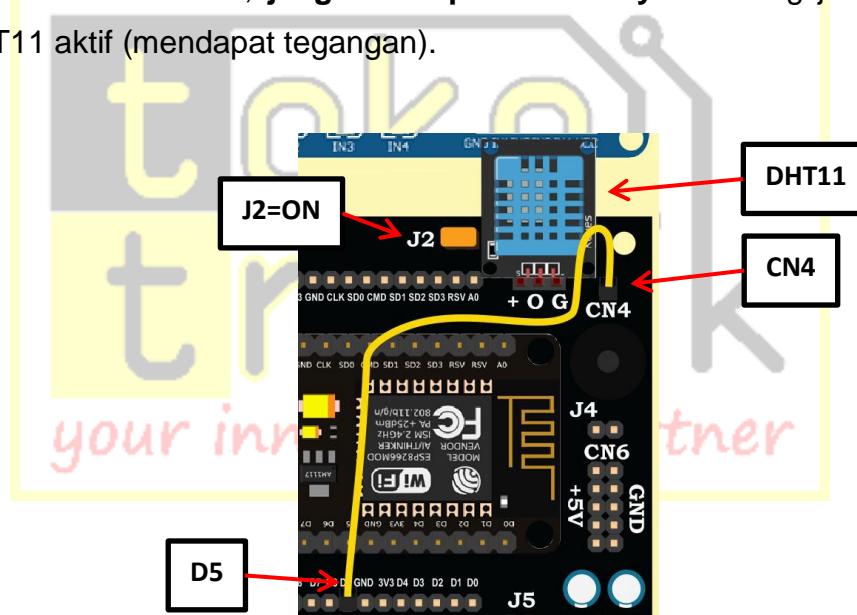
X. Project IoT 5 : Monitoring Suhu Kelembaban di Thingspeak

IoT Starter Kit Inkubatek

Pada project IoT ke-5 ini hampir sama dengan project ke-4. Ada tambahan satu nilai lagi yaitu kelembaban. Kita akan memanfaatkan sensor suhu kelembaban DHT11. Sensor DHT11 terdiri dari 3 kaki, dengan kaki output berupa digital (berbeda dengan LM35).



Pasang sensor DHT11 ke konektor female 3 pin yang ada di board IoT Starter Kit Inkubatek, **jangan sampai terbalik ya**. Pasang jumper J2 agar DHT11 aktif (mendapat tegangan).



Koneksikan CN4 (output sensor DHT11) dengan pin D5. Sekarang kita buat Channel lagi di *Thingspeak*.

My Channels

New Channel

Name
Monitoring Suhu LM35

Private Public Settings Sharing API Keys Data Import / Export

Klik **New Channel** kemudian isi data - datanya. Centang Field 2.

New Channel

Name Monitoring Suhu Kelembaban

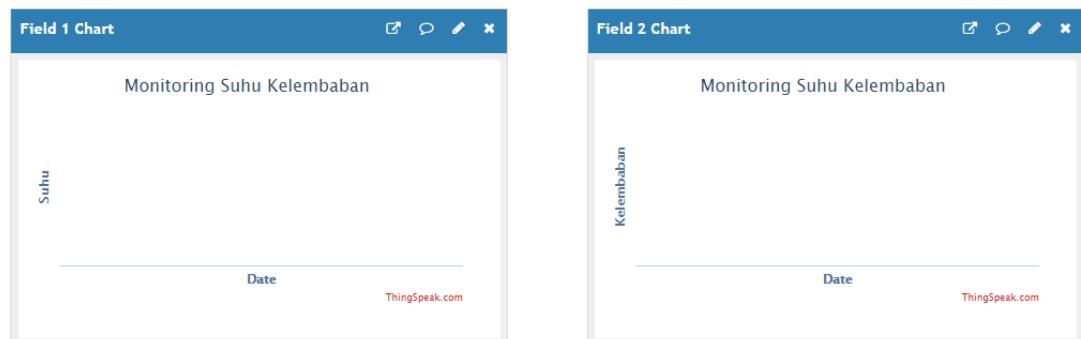
Description Suhu kelembaban dengan DHT11

Field 1 Suhu

Field 2 Kelembaban

Field 3

Klik “Save” yang ada di bagian bawah. Kemudian akan muncul 2 tempat untuk menampilkan data berupa grafik suhu dan juga grafik kelembaban.



Ok, sekarang kita lihat API Keys-nya. Klik tab API Keys :

Private View Public View Channel Settings Sharing API Keys

Write API Key

Key 8SYIT7FN65P5HD

Generate New Write API Key

IoT Starter Kit Inkubatek

Perhatikan **API Key** ini nanti akan kita masukkan ke program NodeMCU V3. Selesai dulu untuk *Thingspeak*, sekarang kita masuk lagi ke board **IoT Starter Kit Inkubatek**.

Buat program ini ya..

```
*****
* Program :Project 5 Monitoring Suhu Kelembaban Thingspeak
* Input : DHT11 di pin D5
* Output : Grafik Thingspeak
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****/




#include <DHT.h>
#include <ESP8266WiFi.h>

// ganti dengan API Keys anda di Thingspeak
String apiKey = "8SYIT7FN65P5HD";
const char* ssid = "FirnasAbe";
const char* password = "podowingi";
const char* server = "api.thingspeak.com";
#define DHTPIN 5 // DHT11 terhubung dengan PIN D5 NODEMCU


DHT dht(DHTPIN, DHT11,15);
WiFiClient client;
=====

void setup() {
  Serial.begin(115200);
  delay(10);
  dht.begin();
  WiFi.begin(ssid, password);
```

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

WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
}

//=====
void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();
    if (isnan(h) || isnan(t)) {
        Serial.println("DHT11 tidak terbaca... !");
        return;
    }
    if (client.connect(server,80)) {
        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(t);
        postStr += "&field2=";
        postStr += String(h);
        postStr += "\r\n\r\n";
        client.print("POST /update HTTP/1.1\r\n");
        client.print("Host: api.thingspeak.com\r\n");
        client.print("Connection: close\r\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);

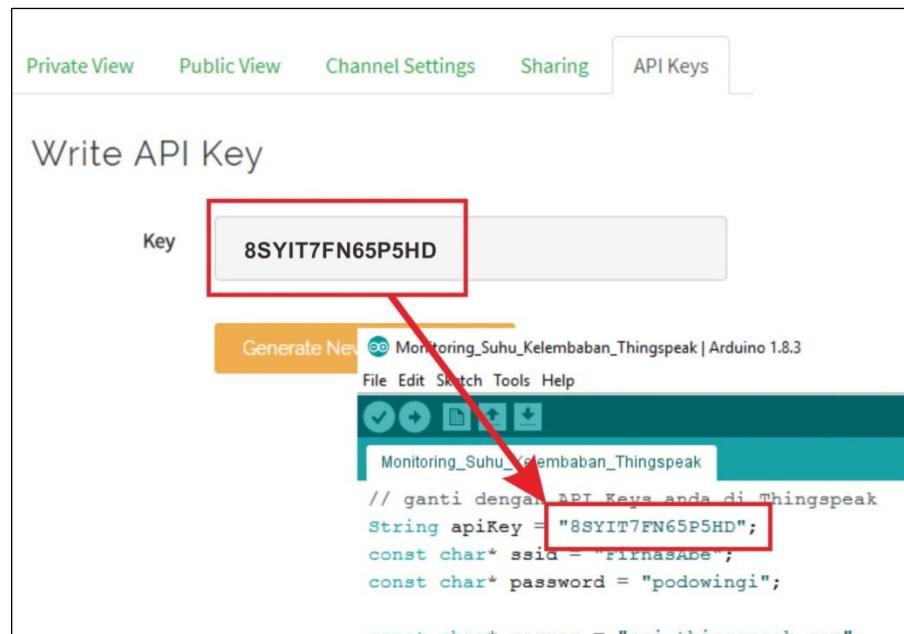
Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" Celcius Humidity: ");
Serial.print(h);
Serial.println("% send to Thingspeak");
}

client.stop();
Serial.println("Waiting...");
delay(2000); //minimal nunggu 15 detik update Thingspeak
}
```

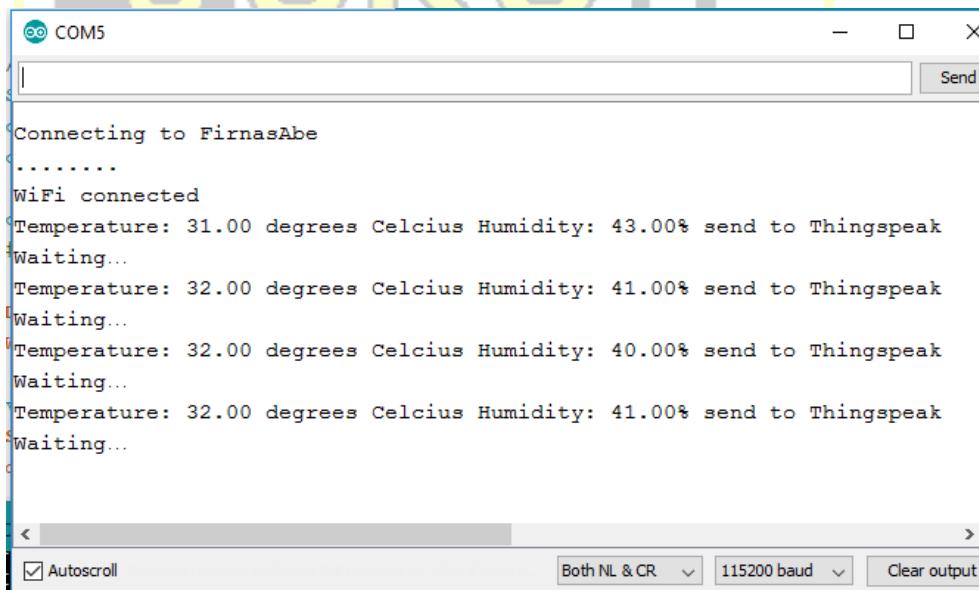
Sesuaikan beberapa variable berikut :

- 
- ✓ API Keys Cek di akun *Thingspeak*, copy paste ke program untuk variable apiKey
 - ✓ Nama jaringan WiFi , isikan ke ssid
 - ✓ Password WiFi , isikan ke password

IoT Starter Kit Inkubatek

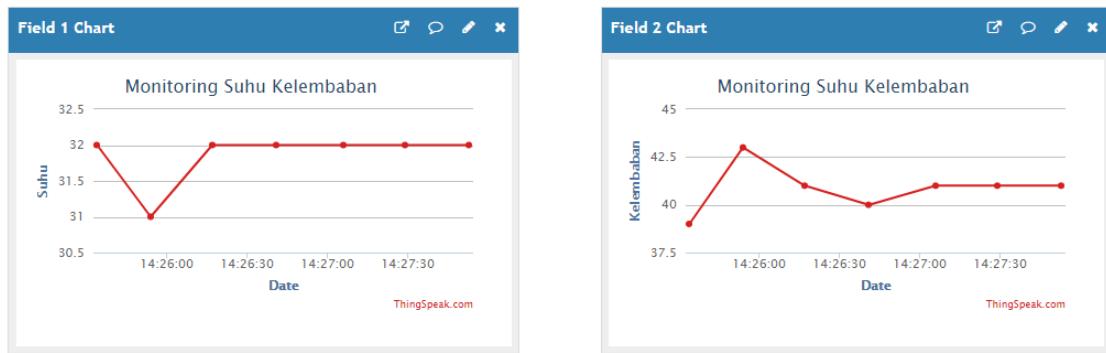


Save kemudian **Upload**. Pastikan tidak ada error. Buka Serial Monitor di Arduino IDE (**Tools Serial Monitor**) maka akan tampil :



Cek di Channel *Thingspeak* anda :

IoT Starter Kit Inkubatek

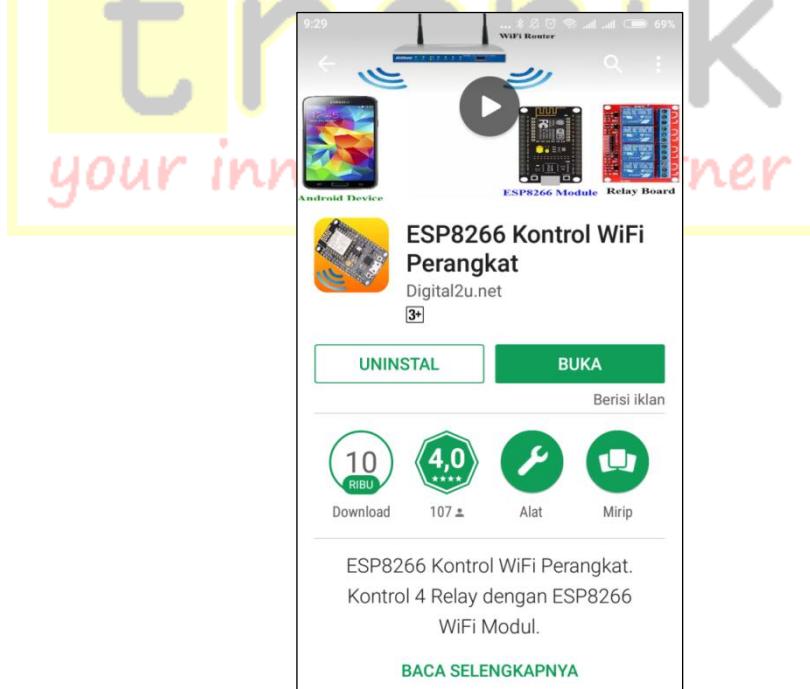


Tampak pada channel tersebut nilai suhu (temperature) dan juga kelembaban dari hasil pembacaan sensor DHT11.

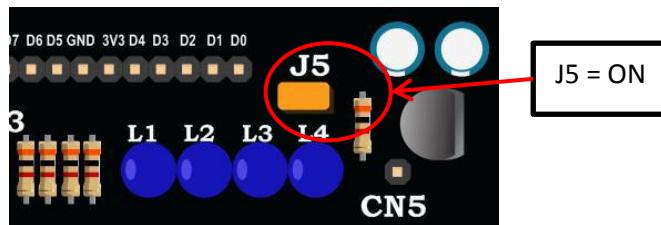
XI. Project IoT 6 : Kontrol LED dengan Android

Aplikasi IoT selanjutnya akan berhubungan dengan Android. Kita akan memakai aplikasi Android yang sudah banyak tersedia secara gratis di **Play Store**. Siapkan sebuah jaringan WiFi yang akan dipakai untuk koneksi antara IoT Starter Kit (ESP8266) dengan aplikasi di Android.

Aplikasi Android yang akan kita pakai adalah "**ESP Kontrol WiFi Perangkat**" Pilih yang "**ESP8266 Kontrol WiFi Perangkat**" kemudian **Install**.



Untuk hardware, pasang jumper di **J5** di board IoT Starter Kit agar rangkaian LED aktif.



Sekarang kita siapkan softwarenya. Buat lembar kerja baru di Arduino IDE (*File → New*).

```
*****
* Program :Project 6 Kontrol LED dg Android
* Input : -
* Output : LED1-LED4
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****/
```

```
#include <ESP8266WiFi.h>

#define ON 1
#define OFF 0

const char* ssid = "FirnasAbe"; // Ganti dengan nama jaringan wifi anda
const char* password = "podowingi"; // Ganti dengan password wifi anda

#define LED1 D5 //D5 ESP Board
#define LED2 D6 //D7 ESP Board
#define LED3 D7 //D6 ESP Board
#define LED4 D8 //D5 ESP Board

bool value1 = LOW;
bool value2 = LOW;
bool value3 = LOW;
bool value4 = LOW;
```

```
WiFiServer server(80); // Nomor PORT WiFi
//=====
void setup() {
    Serial.begin(115200);
    pinMode(LED1, OUTPUT);
    pinMode(LED2, OUTPUT);
    pinMode(LED3, OUTPUT);
    pinMode(LED4, OUTPUT);
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    digitalWrite(LED4, LOW);

    // Connect to WiFi network
    Serial.println();
    Serial.println();
    Serial.print("Connecting to ");
    Serial.println(ssid);

    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    server.begin();
    Serial.println("Server started");
    delay (2000);
    // Print the IP address
    Serial.print("Use this URL to connect: ");
}
```

```
Serial.print("http://");
Serial.print(WiFi.localIP());
Serial.println("/");
delay(2000);
}

//=====
void loop() {

    WiFiClient client = server.available();
    if (!client) {
        return;
    }

    Serial.println("new client");
    while(!client.available()){
        delay(1);
    }

    String request = client.readStringUntil('\r');
    Serial.println(request);
    delay(1000);
    client.flush();
    client.println("HTTP/1.1 200 OK");
    client.println("Content-Type: text/html");
    client.println(""); // do not forget this one
    client.println("<!DOCTYPE HTML>");
    client.println("<html>");
    client.println("<fieldset>");
    client.println("<font color = red>");
    client.print("Control 4 LED Via Wifi");
    client.println("</font>");
    client.println("</fieldset>");
    client.println("<br><br>");

}
```

```
// Kontrol LED 1
if (request.indexOf("/status1=1") != -1) {
    digitalWrite(LED1, ON);
    value1 = HIGH;
}
if (request.indexOf("/status1=0") != -1) {
    digitalWrite(LED1, OFF);
    value1 = LOW;
}
if(value1==HIGH){
    client.println("<br><br>");
    client.print("status1=1");
}
if(value1==LOW){
    client.println("<br><br>");
    client.print("status1=0");
}

// Kontrol LED 2
if (request.indexOf("/status2=1") != -1) {
    digitalWrite(LED2, ON);
    value2 = HIGH;
}
if (request.indexOf("/status2=0") != -1) {
    digitalWrite(LED2, OFF);
    value2 = LOW;
}
if(value2==HIGH){
    client.println("<br><br>");
    client.print("status2=1");
}
```

```
if(value2==LOW){  
    client.println("<br><br>");  
    client.print("status2=0");  
}  
  
// Kontrol LED 3  
if (request.indexOf("/status3=1") != -1) {  
    digitalWrite(LED3, ON);  
    value3 = HIGH;  
}  
  
if (request.indexOf("/status3=0") != -1) {  
    digitalWrite(LED3, OFF);  
    value3 = LOW;  
}  
  
if(value3==HIGH){  
    client.println("<br><br>");  
    client.print("status3=1");  
}  
  
if(value3==LOW){  
    client.println("<br><br>");  
    client.print("status3=0");  
}  
  
// Kontrol LED 4  
if (request.indexOf("/status4=1") != -1) {  
    digitalWrite(LED4, ON);  
    value4 = HIGH;  
}  
  
if (request.indexOf("/status4=0") != -1) {  
    digitalWrite(LED4, OFF);  
    value4 = LOW;  
}  
  
if(value4==HIGH){  
    client.println("<br><br>");  
}
```

```
client.print("status4=1");
}

if(value4==LOW){
    client.println("<br><br>");
    client.print("status4=0");
}

delay(1);

Serial.println("Client disconnected");
Serial.println("");
delay(1000);

}
```

Sesuaikan nama jaringan WiFi dan password-nya.

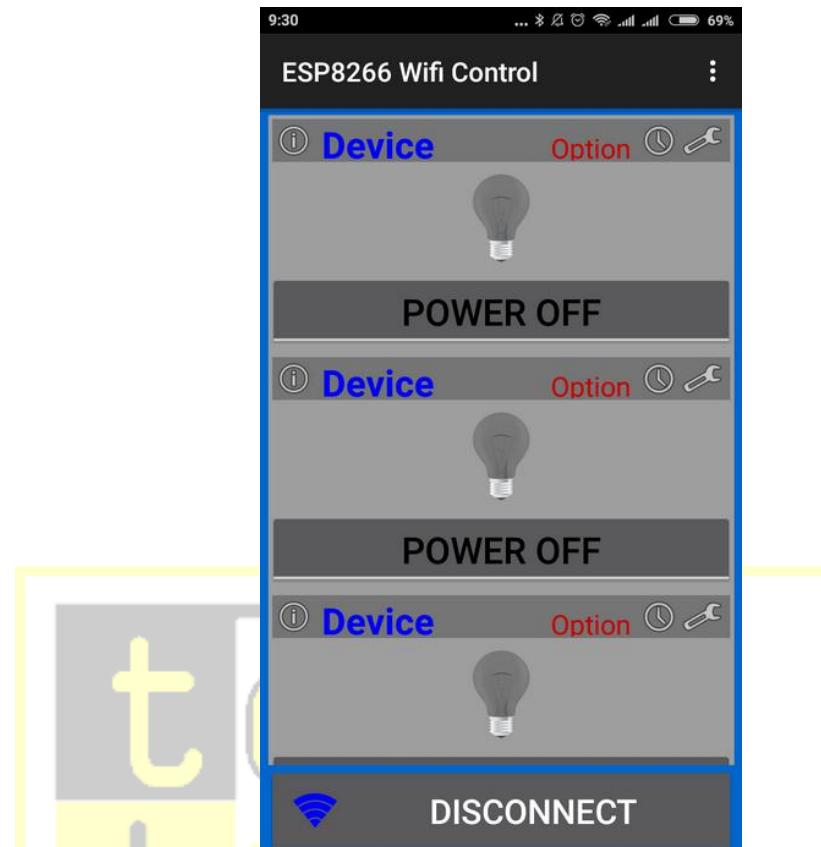
Buka Serial Monitor (*Tools→ Serial Monitor*). Kalau tidak muncul keterangan status di Serial Monitor cek lagi baud rate dan modul di reset (tekan tombol RST yang ada di NodeMCU).

Sekarang buka aplikasi Android “ ESP8266 Kontrol Perangkat ” .

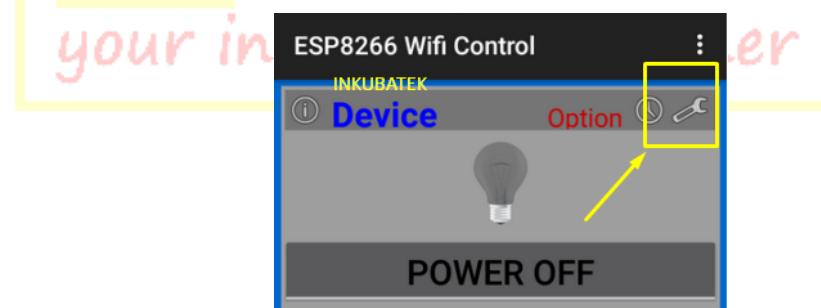


IoT Starter Kit Inkubatek

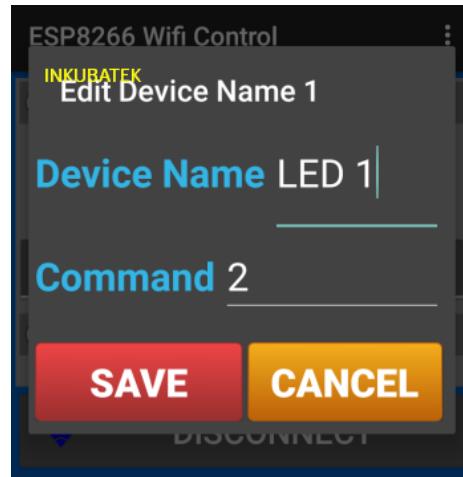
Pastikan nama WiFi (SSID) sesuai, kemudian klik “CONNECT TO”. Kalau sukses akan tampil seperti ini :



OK, selanjutnya kita edit sedikit ya, kita kasih nama untuk masing – masing Device. Pertama klik tombol *Seting* (gambar kunci pas) pada Device 1.



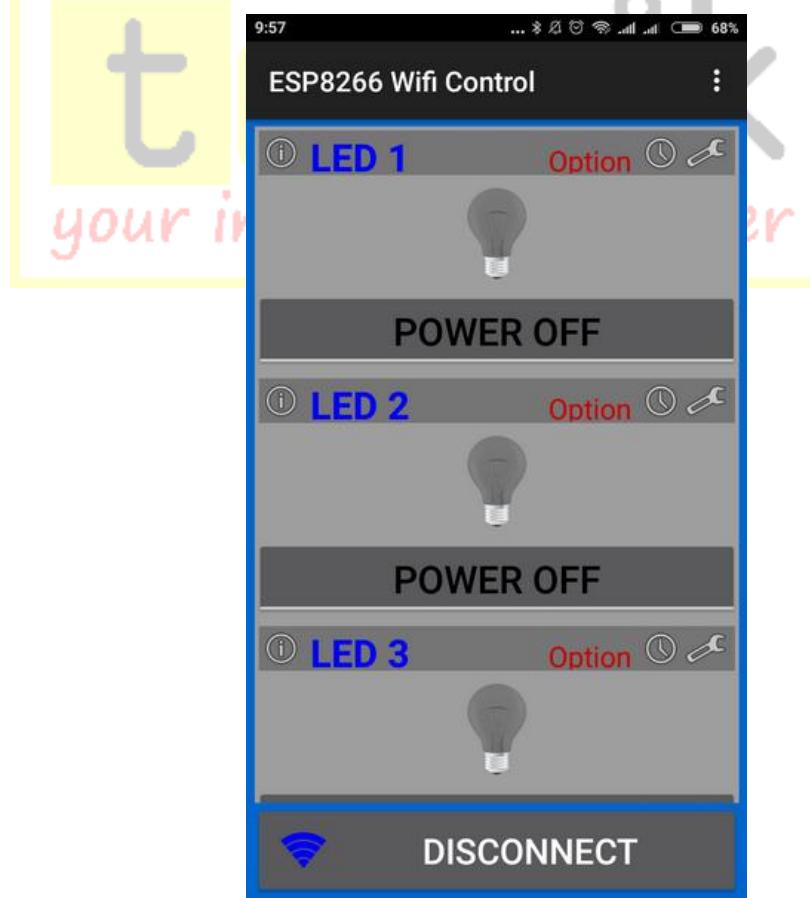
Isikan *Device Name* = LED 1 dan *Command* = 2, kemudian klik *SAVE*.



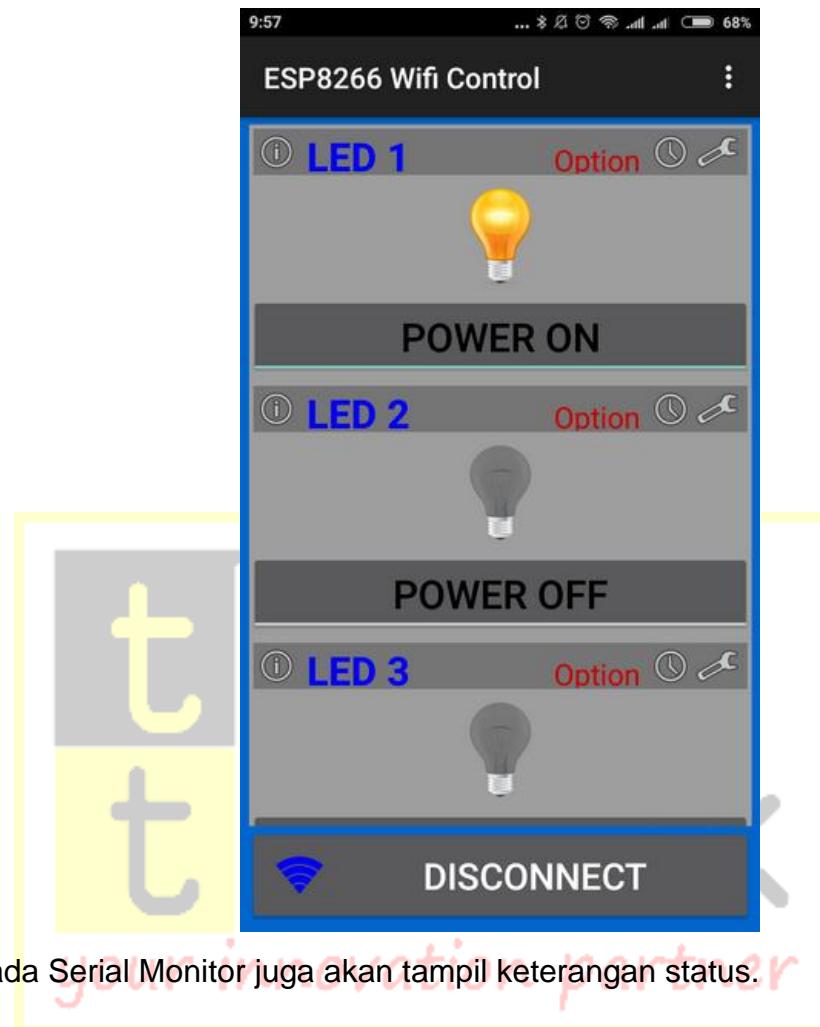
Lakukan pada Device 2 sampai Device 4.

Device	Name	Command
2	LED 2	3
3	LED 3	4
4	LED 4	5

Sehingga tampilannya menjadi seperti ini :



Tekan tombol POWER OFF pada LED 1. Perhatikan LED 1 pada board IoT Starter Kit akan nyala.



Pada Serial Monitor juga akan tampil keterangan status.

XII. Project IoT 7 : Mengirim Notifikasi ke Telegram

Sekarang aplikasi IoT dari inkubatek lebih keren lagi, koneksi dengan aplikasi *chatting* yaitu Telegram, yang hampir mirip dengan WhatsApp (WA).



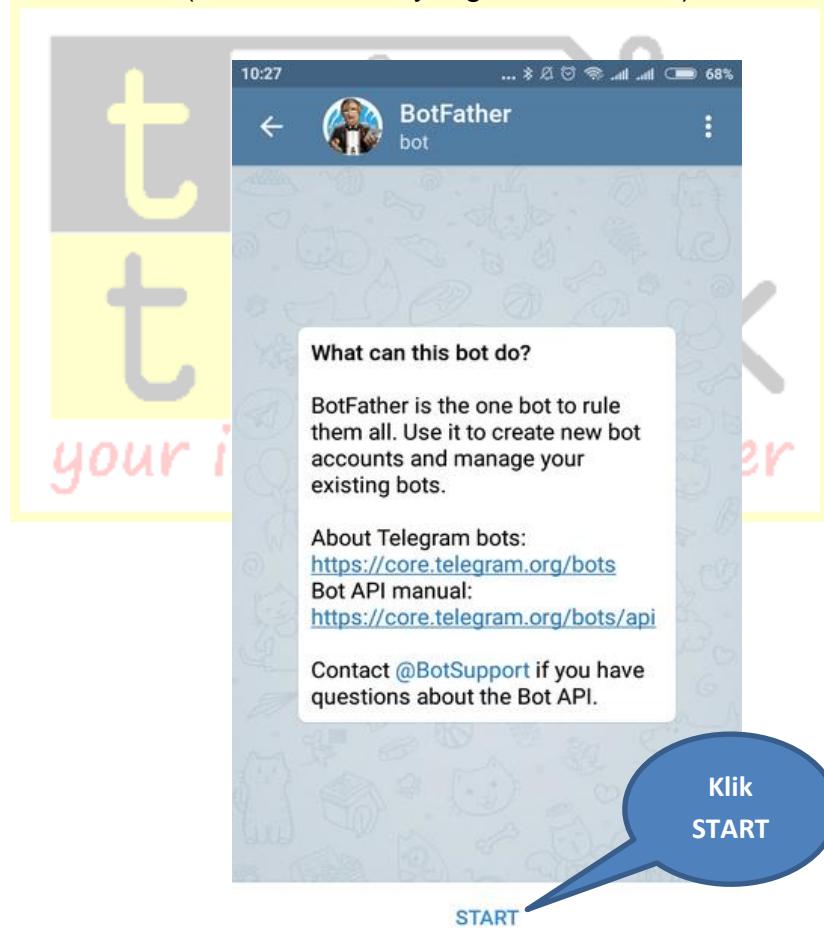
Sebelumnya kita harus siapkan **Telegram Bot** (Robot). Fungsinya untuk memudahkan kita membuat aplikasi chatting atau moderasi di dalam Telegram. Ikuti langkah – langkah pembuatan Bot Telegram berikut ini :

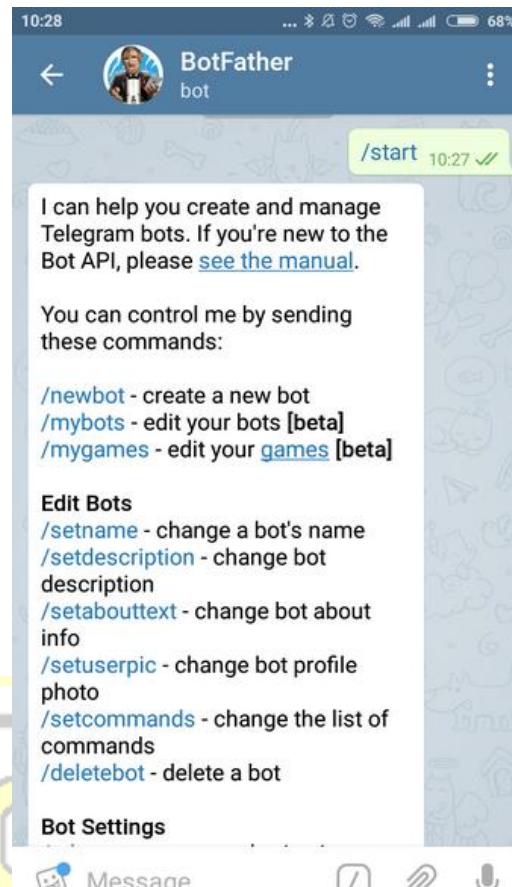
IoT Starter Kit Inkubatek

- Anda harus punya aplikasi Telegram di HP Android , jika belum silakan instal terlebih dulu.
- Pada kolom search ketik botfather. Pilih @BotFather .



- Klik *Start* (tombol START yang ada di bawah)





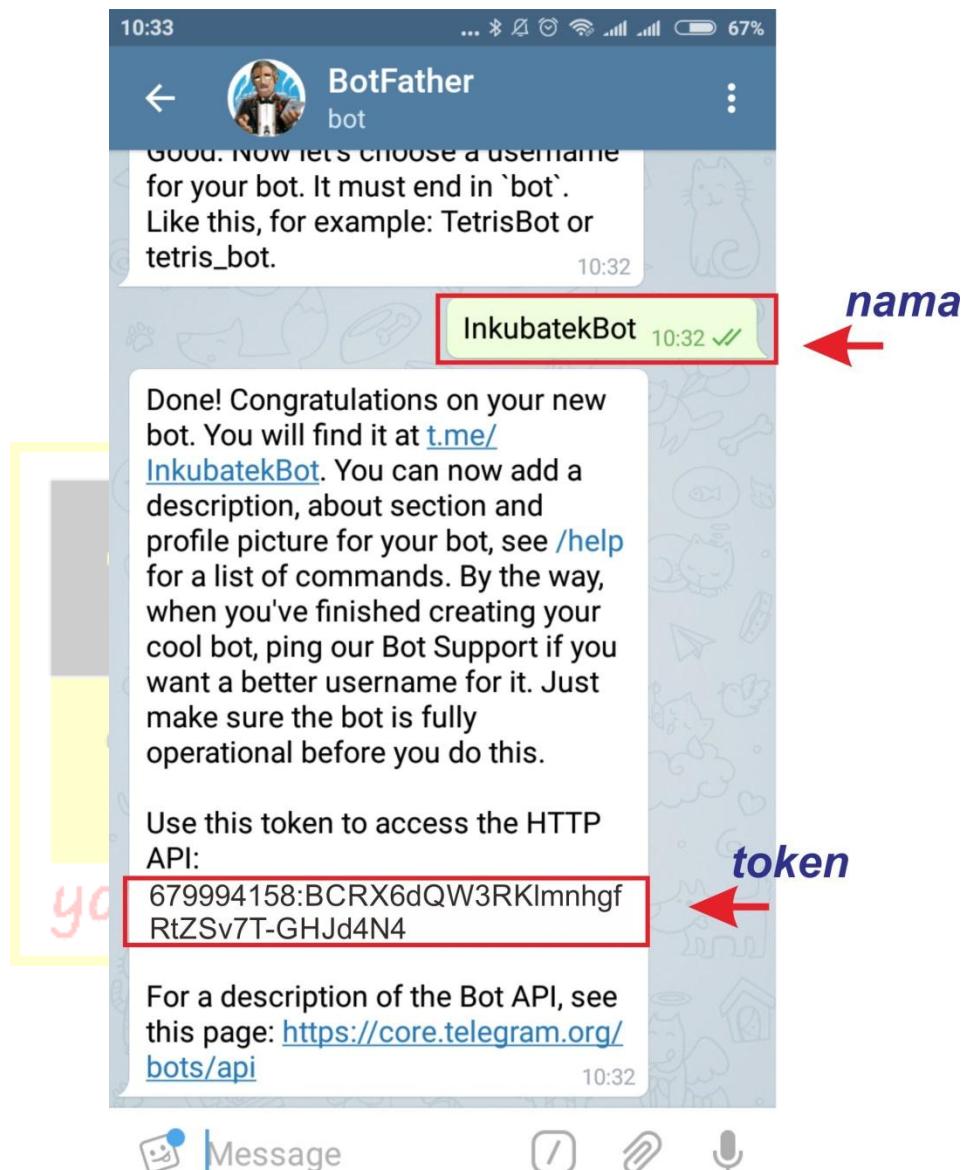
- Selanjutnya kita akan membuat bot baru. Ketik `/newbot`.



- Berikutnya buatlah sebuah nama untuk Bot anda. Setiap nama ditambah kata bot. Pada contoh ini namanya Inkubatek maka menjadi **InkubatekBot**.

IoT Starter Kit Inkubatek

- Kemudian akan muncul juga sebuah token API. Kode ini penting, nanti kita pakai untuk pemrograman NodeMCU agar dapat berkomunikasi dengan Telegram. Pada contoh ini nomor token InkubatekBot : **679994158:BCRX6dQW3RKlmnhgfRtZSv7T-GHJd4N4**.

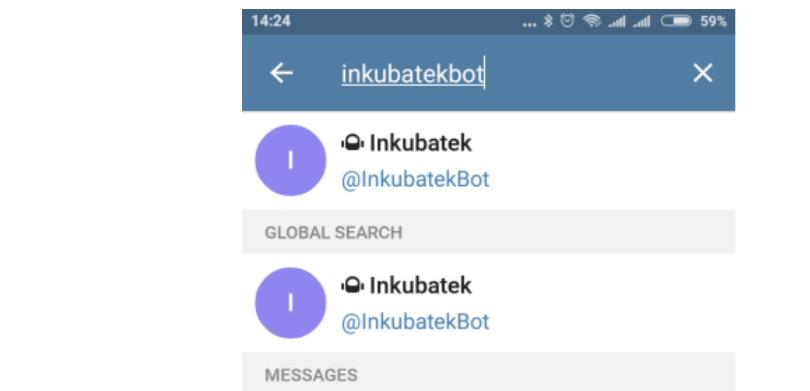


- Oke, selanjutnya kita cek apakah token sudah sesuai atau belum. Buka browser kemudian masuk ke alamat :
<https://api.telegram.org/bot679994158:BCRX6dQW3RKlmnhgfRtZSv7T-GHJd4N4/getUpdates>
- Jika sudah benar maka hasilnya seperti ini :

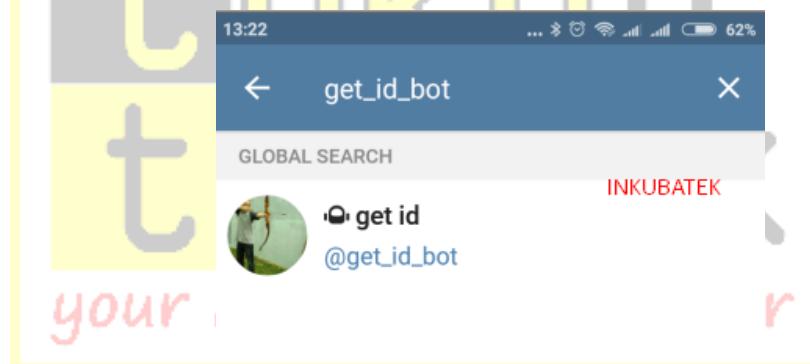
IoT Starter Kit Inkubatek

```
← → C Aman | https://api.telegram.org/bot479093158:AAEMX7bQ03RQOnjfrwWqjZN2vU5-VNJd7T4/getUpdates
{"ok":true,"result":[]}
```

- Anda juga dapat mencoba dengan mencari (*search*) di Telegram



Satu lagi kita akan cek ID telegramnya. Masih dengan aplikasi Telegram, di kolom search cari **get_id_bot**. Pilih **@get_id_bot**.



- 
- Klik “**START**” yang ada di bagian bawah, selanjutnya akan muncul ID
 - Catat ID nya kemudian simpan, nanti akan kita pakai untuk pemrograman.

IoT Starter Kit Inkubatek

Alhamdulillah, sekarang anda sudah punya 3 hal yang nantinya akan kita pakai untuk membuat aplikasi IoT dengan Telegram, yaitu :

- Telegram Bot
- Token
- ID Telegram

Selanjutnya kita buat programnya. Buat lembar kerja baru di Arduino IDE kemudian tulis program berikut :

```
*****
* Program :Project 7 Kirim Notifikasi ke Telegram
* Input : SW1 di pin D2
* Output : Telegram
* Iot Starter Kit Inkubatek
* www.tokotronik.com
*****
#include <UniversalTelegramBot.h>
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
//----- WiFi Settings -----
char ssid[] = "FirnasAbe"; // WiFi
char password[] = "podowingi"; // password WiFi
// ----- Telegram config -----
#define BOT_TOKEN "679994158:BCRX6dQW3RKlmnhgfRtZSv7T-GHJd4N4" // Bot Token Anda
#define CHAT_ID "212967234" // Chat ID Anda
#define tombol D2
WiFiClientSecure client;
UniversalTelegramBot bot(BOT_TOKEN, client);
String ipAddress = "";
volatile bool tombolFlag = false;
int button;
=====
```

```
void setup() {
    Serial.begin(115200);
    pinMode(tombol, INPUT);
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    delay(100);

    Serial.print("Connecting Wifi: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    IPAddress ip = WiFi.localIP();
    Serial.println(ip);
    ipAddress = ip.toString();
}

//=====
void sendTelegramMessage() {
    Serial.println("Kirim pesan ke Telegram");
    String message = "Tombol ditekan ....";
    message.concat("\n");
    message.concat("IoT Starter Kit Inkubatek");
    message.concat("\n");
    if(bot.sendMessage(CHAT_ID, message, "Markdown")){
        Serial.println("Pesan telah dikirim ke TELEGRAM");
    }
    else Serial.println("gagal kirim..");
    tombolFlag = false;
}
```

```
}

//=====

void loop() {
    button = digitalRead(tombol);
    if(button==HIGH){
        Serial.println("Tombol ditekan");
        while(button == digitalRead(tombol)); //nunggu low
        sendTelegramMessage();
        delay(500);
    }
}
```

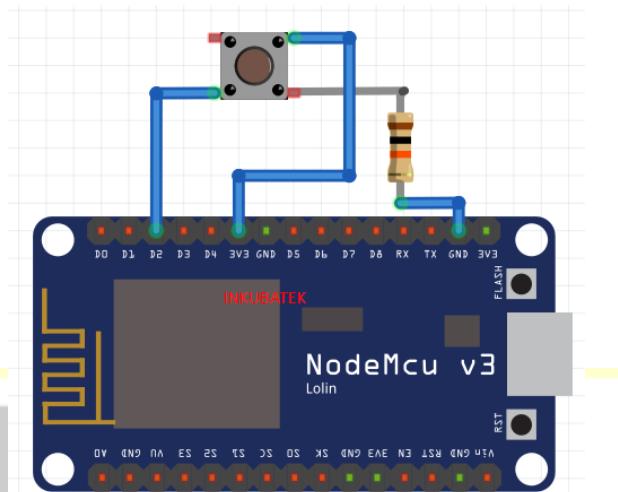
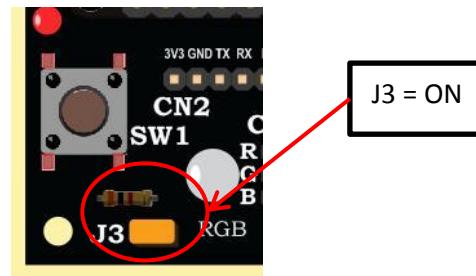
Jangan lupa ganti/sesuaikan :

- SSID
- Password WiFi
- Bot Token
- Chat ID

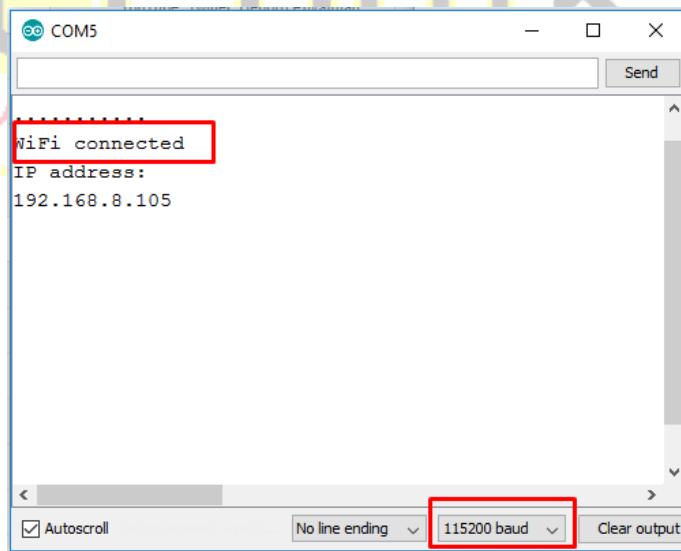


Hardware yang diperlukan satu buah push button (SW1). Pasang jumper J3 sehingga SW1 terhubung dengan pin D2.

IoT Starter Kit Inkubatek



Upload program ke board NodeMCU V3 yang ada di **IoT Starter Kit Inkubatek**. Buka Serial Monitor (*Tools → Serial Monitor*), baud rate 115200, akan tampil status koneksi dengan WiFi.



Jika belum muncul di Serial Monitor, silakan tekan tombol reset (RST) di board NodeMCU nya.

Sekarang kita coba jalannya alat. Tekan tombol push button (SW1), maka pesan akan muncul di Telegram.



Untuk mengganti isi pesan di Telegram, cukup edit tulisan yang ada di variable message :

```
Project_7_Kirim_Notify_via_Telegram | Arduino 1.8.3
File Edit Sketch Tools Help
Project_7_Kirim_Notify_via_Telegram
}

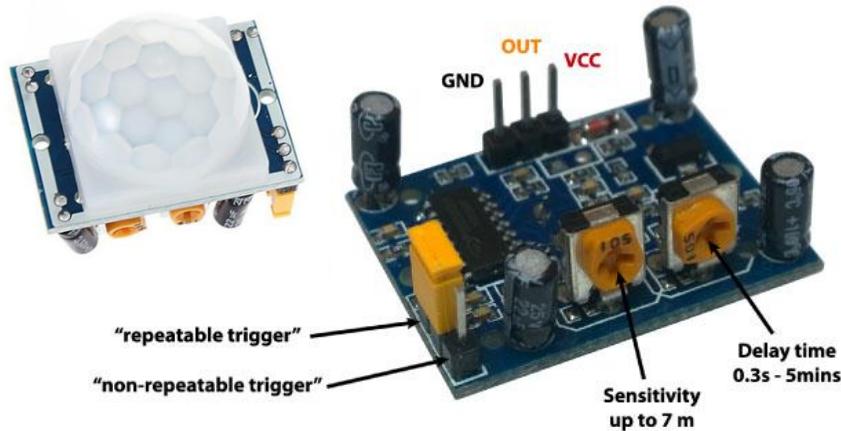
//=====
void sendTelegramMessage() {
    Serial.println("Kirim pesan ke Telegram");
    String message = "Tombol ditekan ....";
    message.concat("\n");
    message.concat("IoT Starter Kit Inkubatek");
    message.concat("\n");
    if(bot.sendMessage(CHAT_ID, message, "Markdown"))
        Serial.println("Pesan telah dikirim ke TELEGRAM");
}
```

Karakter "\n" adalah karakter "Enter" atau pindah baris baru.

XIII. Project IoT 8 : Sistem Keamanan Rumah dg Sensor PIR & Telegram

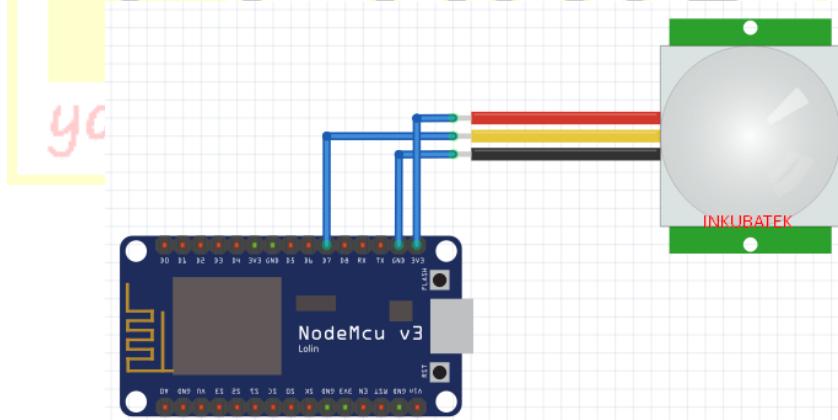
IoT Starter Kit Inkubatek

Mari kita kembangkan aplikasi IoT kita. Sistem keamanan rumah dengan sensor PIR yang akan mendeteksi keberadaan manusia kemudian mengirimkan ke Telegram.



Langkah – langkah membuat Telegram Bot dll seperti di proyek 7. Kita pakai saja data – datanya.

Koneksikan pin sensor PIR dengan modul IoT Starter Kit Inkubatek seperti rangkaian.



Sensor PIR	NodeMCU
VCC	3V
OUT	D5
GND	GND

IoT Starter Kit Inkubatek



Selanjutnya kita buat programnya. Jangan lupa untuk mengganti :

- SSID
- Password
- Bot Token
- Chat ID

```
*****  
* Program :Project 8 Sistem Keamanan Rumah sensor PIR ke Telegram  
* Input : Sensor PIR di pin D5  
* Output : Telegram  
* Iot Starter Kit Inkubatek  
* www.tokotronik.com  
* *****  
  
#include <UniversalTelegramBot.h>  
#include <ESP8266WiFi.h>  
#include <WiFiClientSecure.h>  
  
//----- WiFi Settings -----  
char ssid[] = "FirnasAbe"; // WiFi  
char password[] = "podowingi"; // password WiFi  
// ----- Telegram config -----  
#define BOT_TOKEN "679994158:BCRX6dQW3RKlmnhgfRtZSv7T-GHJd4N4" //  
Bot Token Anda  
#define CHAT_ID "212967234" // Chat ID Anda  
#define PIR_Sensor D5  
WiFiClientSecure client;
```

```
UniversalTelegramBot bot(BOT_TOKEN, client);

String ipAddress = "";
volatile bool PIRFlag = false;
int sensor;

//=====

void setup() {
    Serial.begin(115200);
    pinMode(PIR_Sensor, INPUT);
    WiFi.mode(WIFI_STA);
    WiFi.disconnect();
    delay(1000);
    Serial.print("Connecting Wifi: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    IPAddress ip = WiFi.localIP();
    Serial.println(ip);
    ipAddress = ip.toString();
}

//=====

void sendTelegramMessage() {
    Serial.println("Kirim pesan ke Telegram");
    delay(200);
    String message = "Sensor mendeteksi objek ....";
    message.concat("\n");
    message.concat("WASPADA !!!");
}
```

```
message.concat("\n");
message.concat("IoT Starter Kit Inkubatek");
message.concat("\n");
if(bot.sendMessage(CHAT_ID, message, "Markdown")){
    Serial.println("Pesan telah dikirim ke TELEGRAM");
}
else Serial.println("gagal kirim..");
}

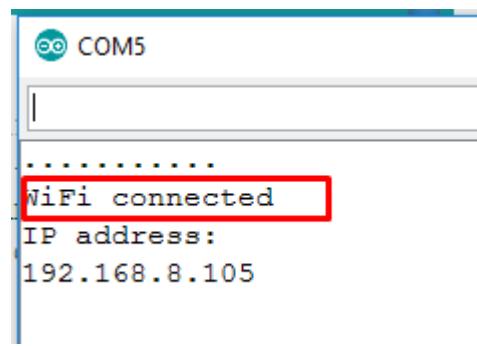
//=====
void loop() {
    sensor = digitalRead(PIR_Sensor);
    if(sensor==HIGH){
        delay(500);
        Serial.println("Sensor PIR Aktif");
        while(sensor = digitalRead(PIR_Sensor)); //nunggu low
        sendTelegramMessage();
        delay(2000);
    }
}
```

Jangan Lupa Ganti :

- **Bot Token**
- **Chat ID**
- **SSID (nama WiFi)**
- **Password WiFi**

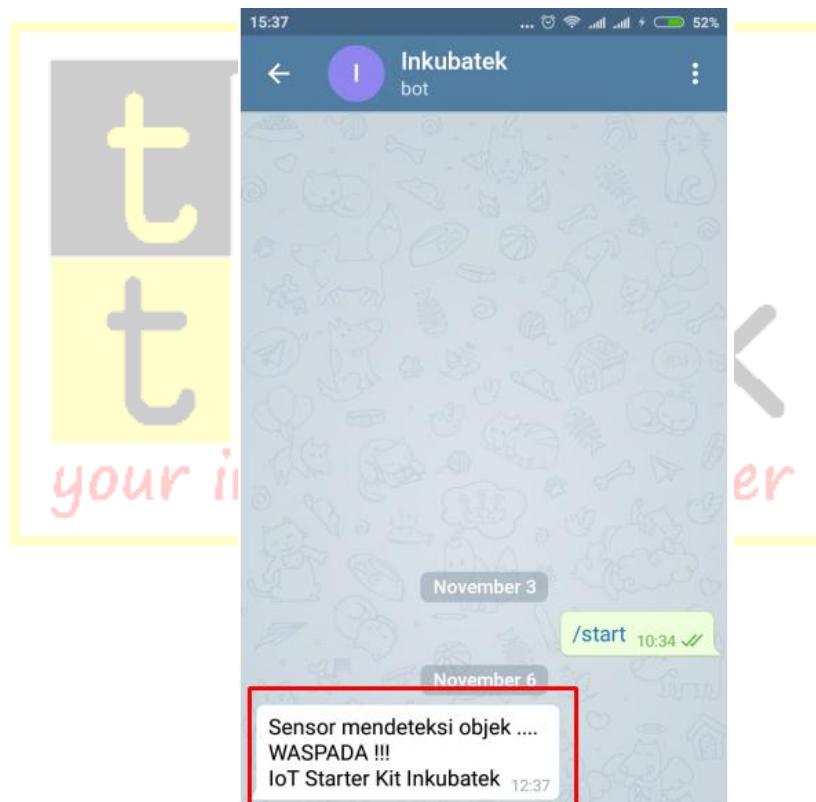
Sensor PIR ini sangat sensitive, jadi sebaiknya di daerah depan sensor PIR jangan ada benda2 yang bergerak.

Ok, simpan program (Save) kemudian Upload ke board NodeMCU. Selanjutnya buka Serial Monitor (*Tools → Serial Monitor*) maka akan muncul informasi koneksi :



Jika belum muncul data di Serial Monitor (blank) silakan reset modul (tombol RST) , cek juga baud ratanya (115200).

Kita coba ya, gerakkan benda (tangan anda) di depan sensor PIR, maka sensor akan mendeteksi dan alat akan mengirim pesan Telegram HP anda.



Untuk mengganti pesan ke Telegram cukup anda ganti isi variable “**message**” saja.



```
Project_8_Sensor_PIR_Telegram | Arduino 1.8.3
File Edit Sketch Tools Help
Project_8_Sensor_PIR_Telegram
//=====
void sendTelegramMessage() {
    Serial.println("Kirim pesan ke Telegram");
    delay(200);
    String message = "Sensor mendeteksi objek ... ";
    message.concat("\n");
    message.concat("WASPADA !!!");
    message.concat("\n");
    message.concat("IoT Starter Kit Inkubatek");
    message.concat("\n");
    if(bot.sendMessage(CHAT_ID, message, "Markdown"))
        Serial.println("Peser telah dikirim ke TELEGRAM"
}
```

Karakter "\n" adalah karakter "Enter" atau ganti baris baru.

XIV. Project IoT 9 : Kontrol Lampu via Telegram

Masih memakai aplikasi Telegram. Pada proyek/aplikasi ke-9 ini kita akan mengontrol lampu AC melalui Telegram.

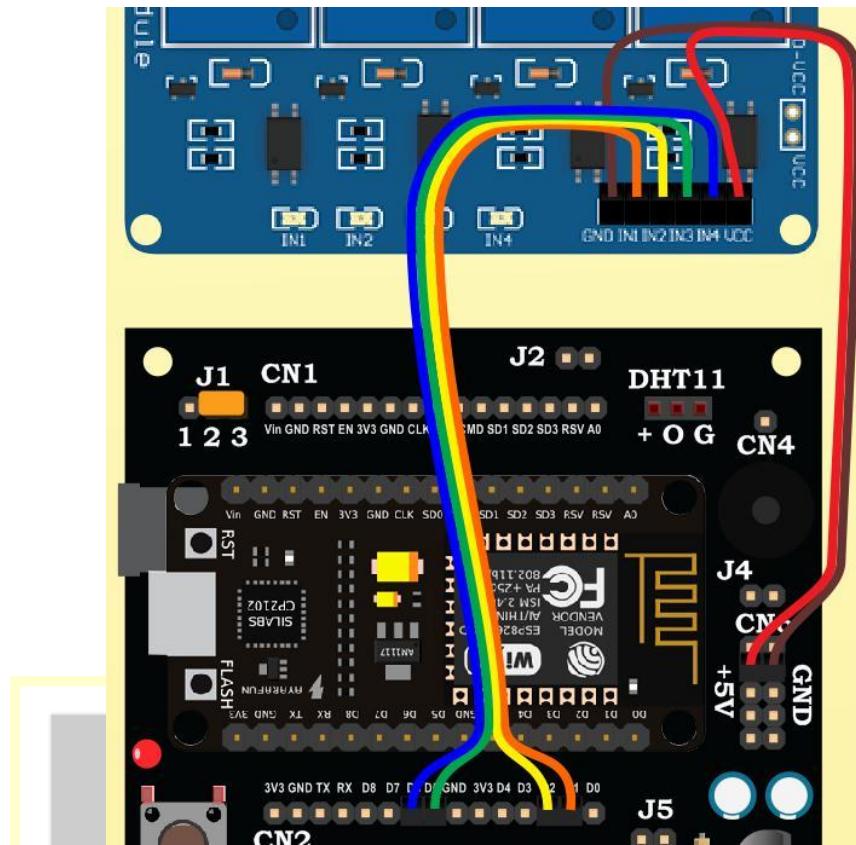
Langkah – langkah membuat Telegram Bot dll seperti di proyek 7. Kita pakai saja data – datanya.

Hardware yang diperlukan sebuah modul relay 4 channel dengan NodeMCU, semua sudah ada di **IoT Starter Kit Inkubatek**.

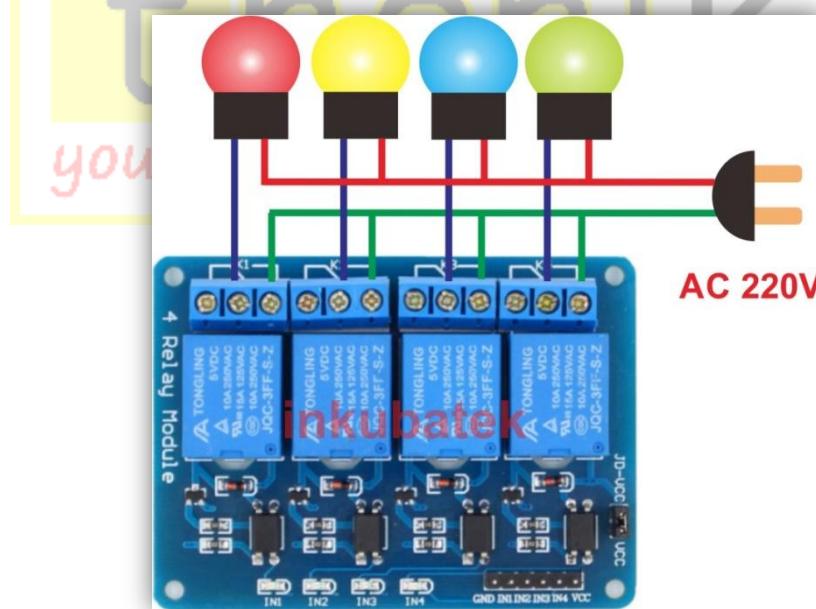
Koneksi modul 4 relay dengan modul **IoT Starter Kit Inkubatek** :

Modul Relay	IoT Starter Kit
VCC	VCC
GND	GND
IN1	D1
IN2	D2
IN3	D5
IN4	D6

IoT Starter Kit Inkubatek



Koneksi modul relay dengan lampu AC :



Selanjutnya kita siapkan programnya di Arduino IDE. Jangan Lupa Ganti :

- Password WiF
 - Bot Token

➤ **SSID (nama WiFi)**

➤ **Chat ID**

```
*****
* Program :Project 9 Kontrol Lampu via Telegram
* Input :
* Output : Relay
* Iot Starter Kit Inkubatek
* www.tokotronik.com
*****
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <TelegramBot.h>

#define RL1 D1
#define RL2 D2
#define RL3 D5
#define RL4 D6

// Sesuaikan dengan jaringan anda
const char* ssid    = "FirnasAbe"; // ganti dengan WiFi anda
const char* password = "podowingi";//ganti dengan password anda

// ganti dengan Token anda
const char BotToken[] = "889oyu158:AAEMX7bQo3RQOnjfrwWqjZN2vU5-
VNJd7T4";

WiFiClientSecure net_ssl;
TelegramBot bot(BotToken, net_ssl);

void setup()
{
  pinMode(RL1, OUTPUT);
  pinMode(RL2, OUTPUT);
```

```
pinMode(RL3, OUTPUT);
pinMode(RL4, OUTPUT);
digitalWrite(RL1,HIGH);
digitalWrite(RL2,HIGH);
digitalWrite(RL3,HIGH);
digitalWrite(RL4,HIGH);
delay(500);
Serial.begin(115200);

delay(3000);

WiFi.mode(WIFI_STA);
WiFi.disconnect();
delay(100);
Serial.print("Connecting Wifi: ");
Serial.println(ssid);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
}
Serial.println("");
Serial.println("WiFi connected");
delay(1000);
bot.begin();

}

void loop()
{
message m = bot.getUpdates(); // Read new messages
=====
if (m.text.equals("1 on")){
    digitalWrite(RL1,LOW);
    bot.sendMessage(m.chat_id, "Lampu 1 ON");
}
```

```
delay(1000);
}

else if (m.text.equals("1 off")){
    digitalWrite(RL1,HIGH);
    bot.sendMessage(m.chat_id, "Lampu 1 OFF");
    delay(1000);
}

//=====
else if (m.text.equals("2 on")){
    digitalWrite(RL2,LOW);
    bot.sendMessage(m.chat_id, "Lampu 2 ON");
    delay(1000);
}

else if (m.text.equals("2 off")){
    digitalWrite(RL2,HIGH);
    bot.sendMessage(m.chat_id, "Lampu 2 OFF");
    delay(1000);
}

//=====
else if (m.text.equals("3 on")){
    digitalWrite(RL3,LOW);
    bot.sendMessage(m.chat_id, "Lampu 3 ON");
    delay(1000);
}

else if (m.text.equals("3 off")){
    digitalWrite(RL3,HIGH);
    bot.sendMessage(m.chat_id, "Lampu 3 OFF");
    delay(1000);
}

//=====

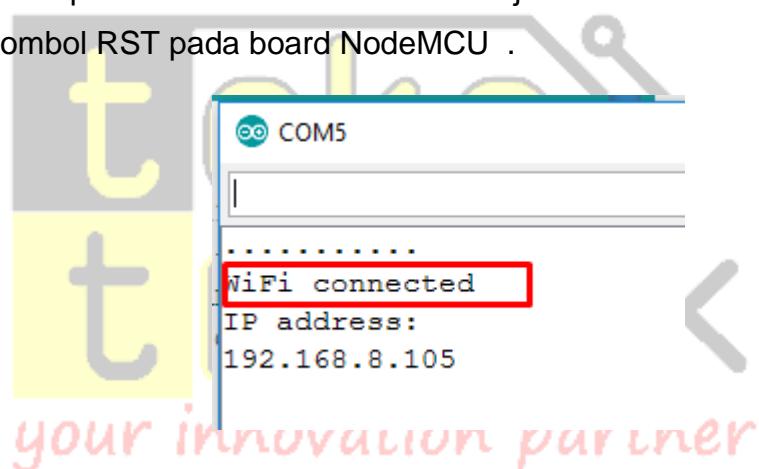
else if (m.text.equals("4 on")){
    digitalWrite(RL4,LOW);
    bot.sendMessage(m.chat_id, "Lampu 4 ON");
    delay(1000);
}
```

```
digitalWrite(RL4,LOW);
bot.sendMessage(m.chat_id, "Lampu 4 ON");
delay(1000);
}

else if (m.text.equals("4 off")){
digitalWrite(RL4,HIGH);
bot.sendMessage(m.chat_id, "Lampu 4 OFF");
delay(1000);
}

}
```

Save kemudian Upload. Buka Serial Monitor (*Tools → Serial Monitor*), tunggu sampai muncul “*WiFi connected*” jika belum muncul coba RESET, tekan tombol RST pada board NodeMCU .

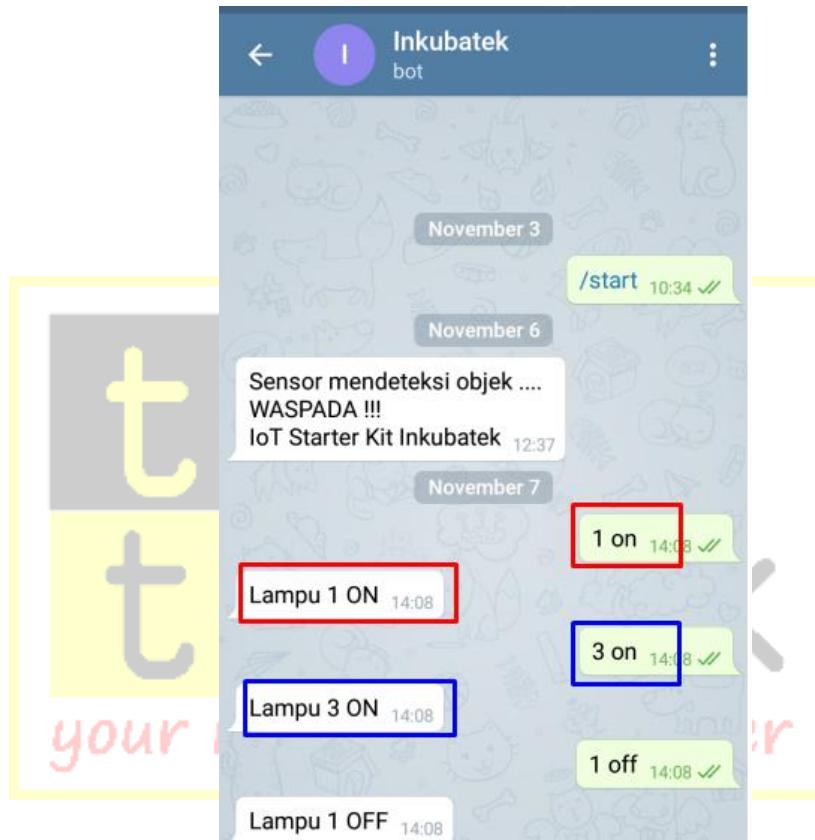


Perhatikan perintah untuk kontrol lampunya, **besar – kecilnya huruf akan berpengaruh.**

Command/perintah	Kontrol
1 on	Lampu 1 ON
1 off	Lampu 1 OFF
2 on	Lampu 2 ON
2 off	Lampu 2 OFF
3 on	Lampu 3 ON
3 off	Lampu 3 OFF

4 on	Lampu 4 ON
4 off	Lampu 4 OFF

Setelah kita ketik perintah misalnya menyalakan lampu 1 : “1 on” (ada spasi) maka relay 1 akan aktif dan menyalakan lampu 1, kemudian akan mengirim Telegram ke Bot Telegram respon balik “Lampu 1 ON”.



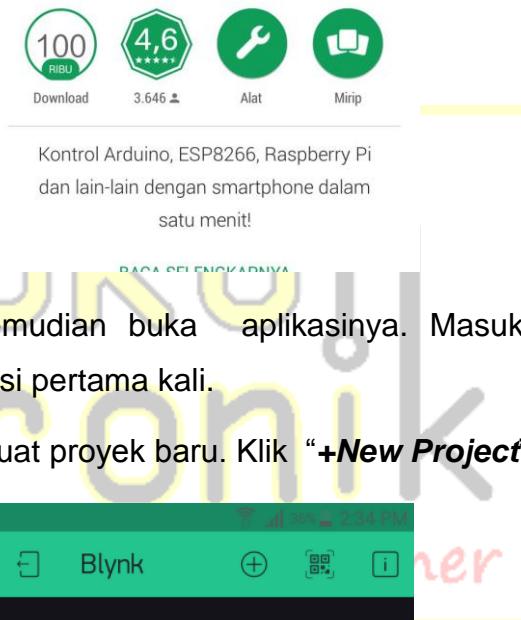
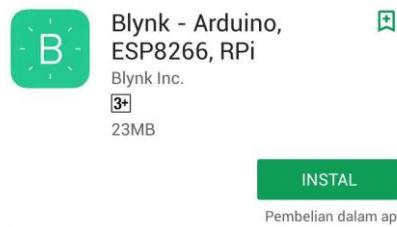
XV. Project IoT 10 : Membuat aplikasi Android Kontrol LED dengan Blynk

([BACA JUGA : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF YANG ADA DI CD](#))

Sekarang kita akan membuat sendiri aplikasi di Android. Keren bukan ? Jadi kita dapat ‘customize’ nantinya sesuai kebutuhan aplikasi kita. Di Play Store sudah ada aplikasi “**BLYNK**” namanya. Ok langsung saja kita siapkan ..

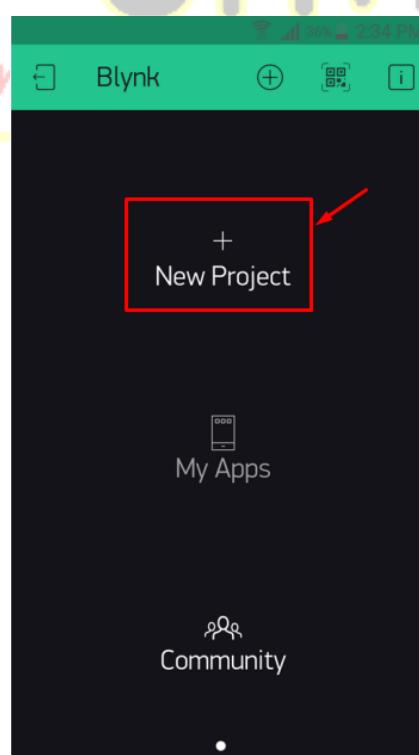
IoT Starter Kit Inkubatek

Buka *Play Store* di Android anda . Ketik “*Blynk*” pilih kemudian *Instal*



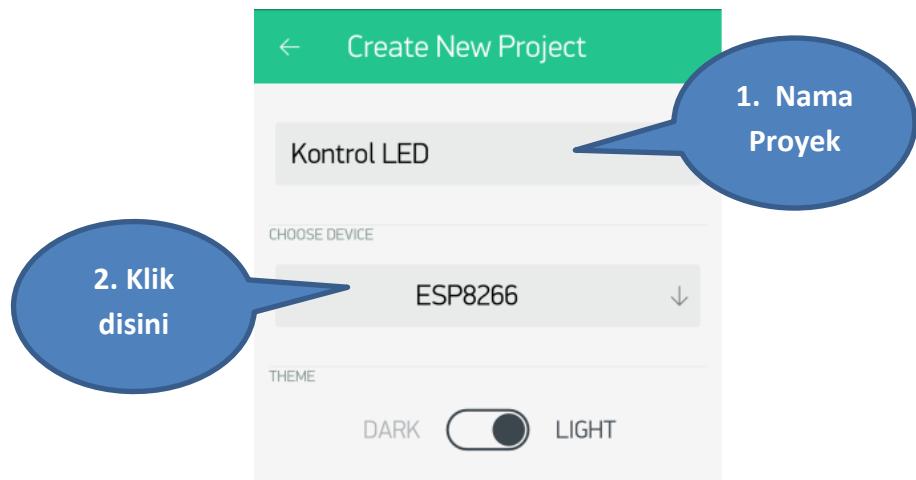
Jika sudah selesai kemudian buka aplikasinya. Masukkan email dan password untuk registrasi pertama kali.

Langsung saja ya kita buat proyek baru. Klik “**+New Project**”

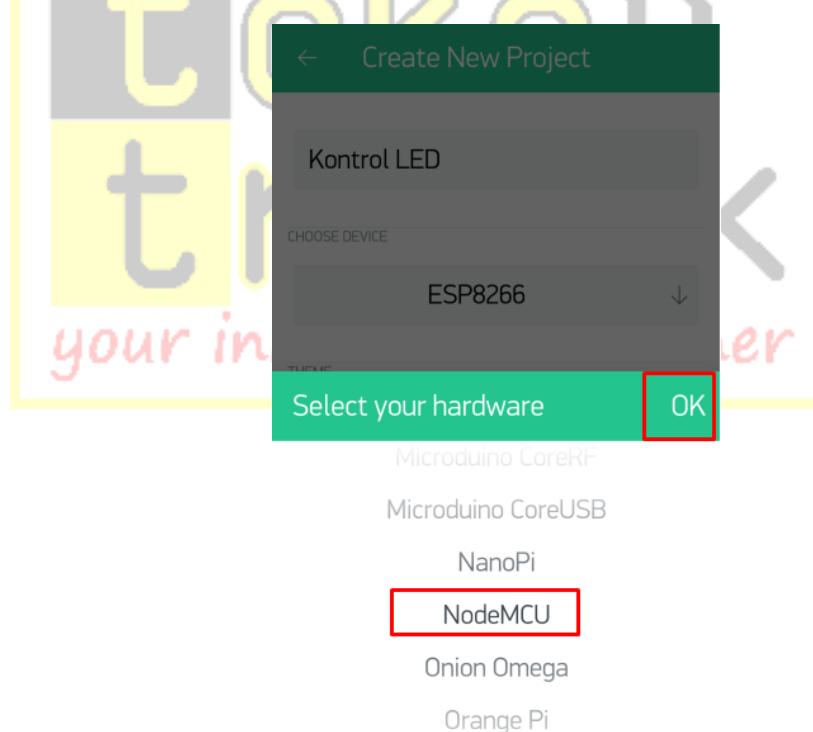


IoT Starter Kit Inkubatek

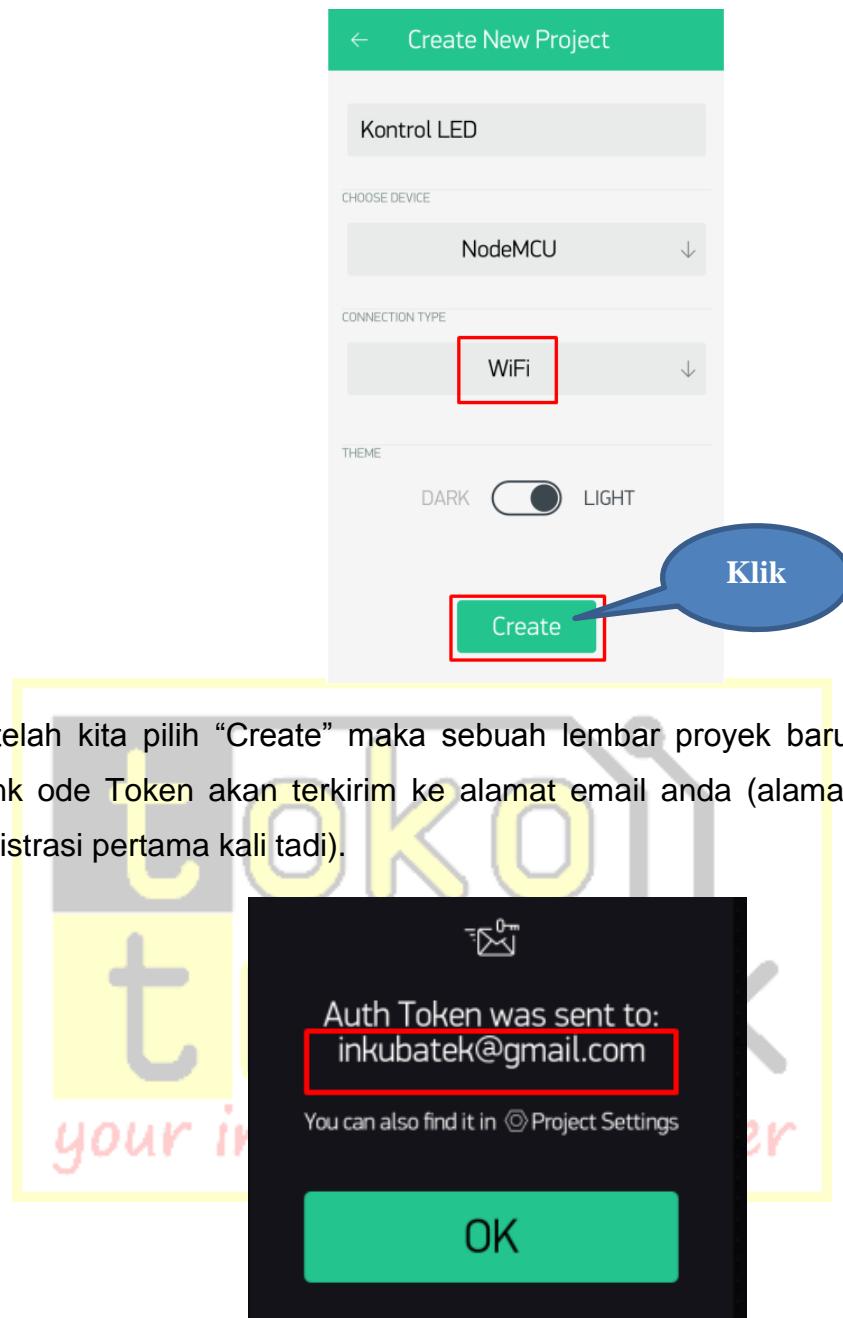
Pada bagian “*Project Name*” beri nama project misalnya **Kontrol LED**



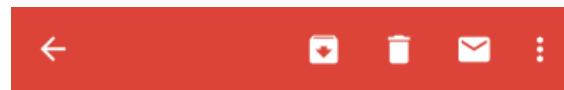
Selanjutnya kita pilih device nya, klik di bagian *CHOOSE DEVICE*. Pilih NodeMCU (gulir kebawah), kemudian OK.



Bagian “*CONNECTION TYPE*” pilih *WiFi*, kemudian klik tombol *Create*



Token adalah sebuah kode (susunan angka dan huruf) yang unik, dipakai nanti ketika pemrograman. Hampir sama dengan kode API nya Thingspeak dan Bot Token Telegram (Aplikasi IoT sebelumnya). Setiap proyek mempunyai token yang berbeda dengan proyek yang lain. Catat dan simpan nomor token-nya.



Auth Token for Kontrol LED project and device Kontrol LED



Inbox



Blynk

to me

2:36 PM [View details](#)



Auth Token : b4c803570c284b618e18ac29a5860b06

Happy Blynking!

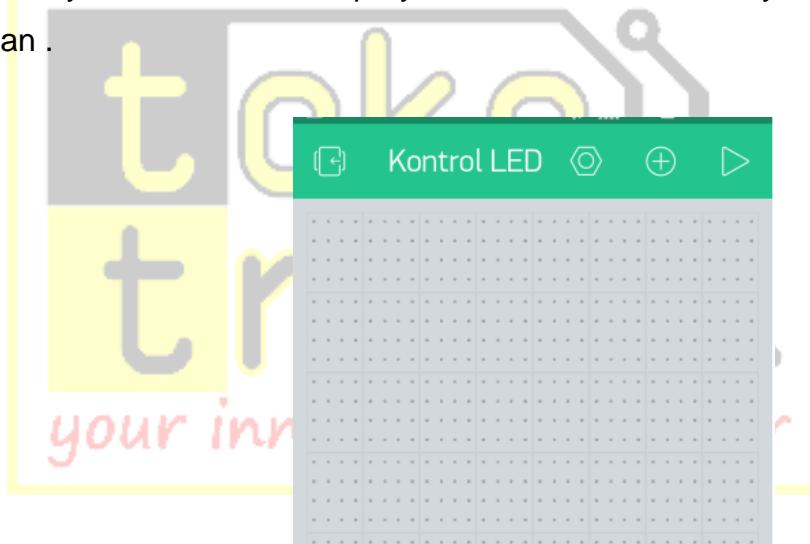
- Getting Started Guide ->

<https://www.blynk.cc/getting-started>

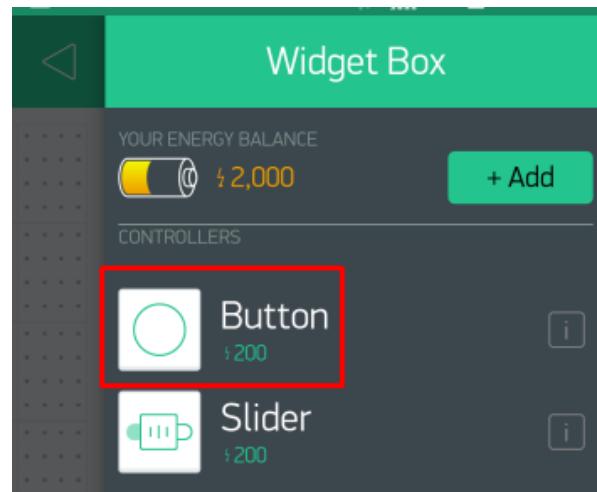
Documentation -> <http://docs.blynk.cc/>

Sketch generator -> <https://examples.blynk.cc/>

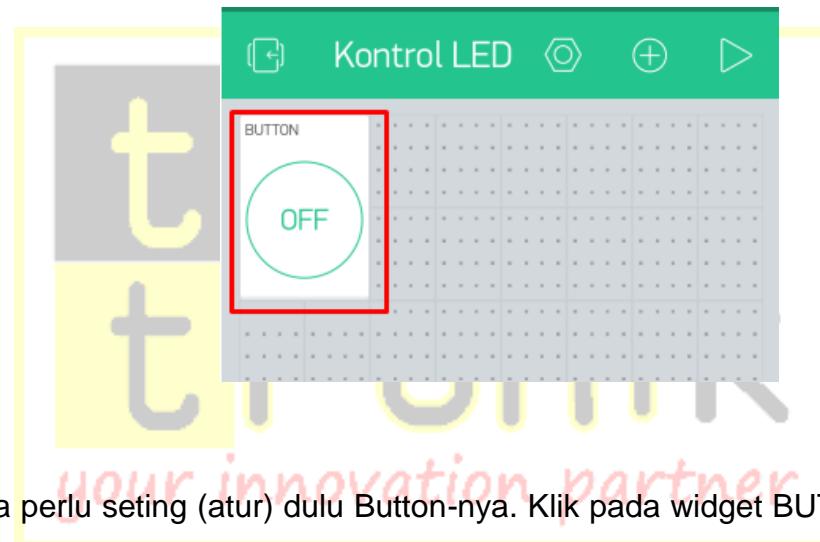
Berikutnya muncul halaman proyek baru. Klik tanda *+* yang ada di atas - kanan .



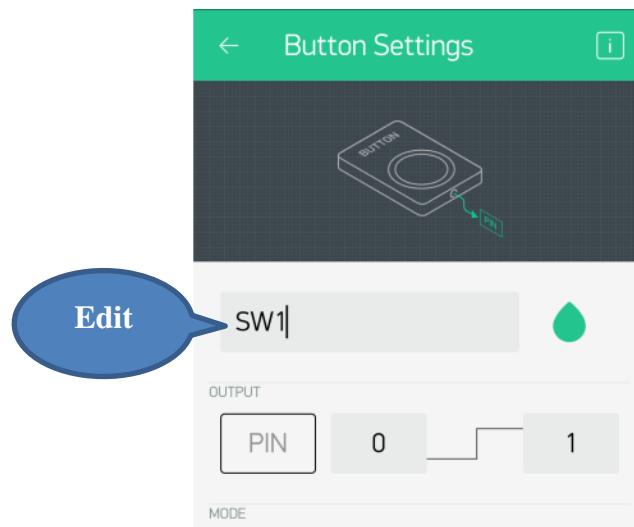
Muncul pilihan device (*widget box*) yang akan kita pakai pada aplikasi yang akan dibuat.



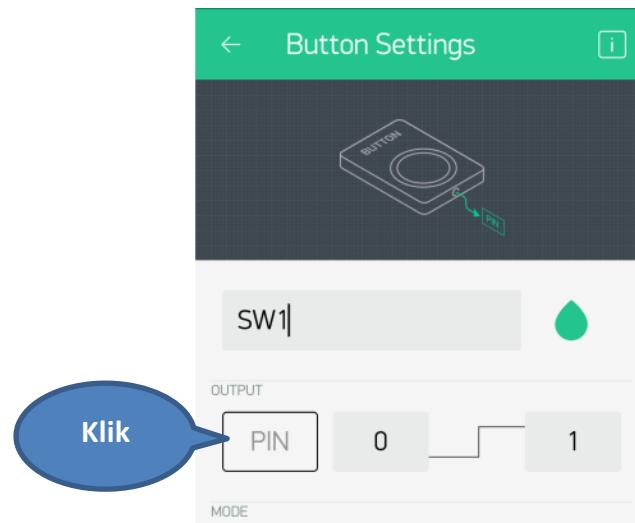
Ok, klik pada widget Button, selanjutnya sebuah widget Button (tombol) telah ditambahkan ke proyek.



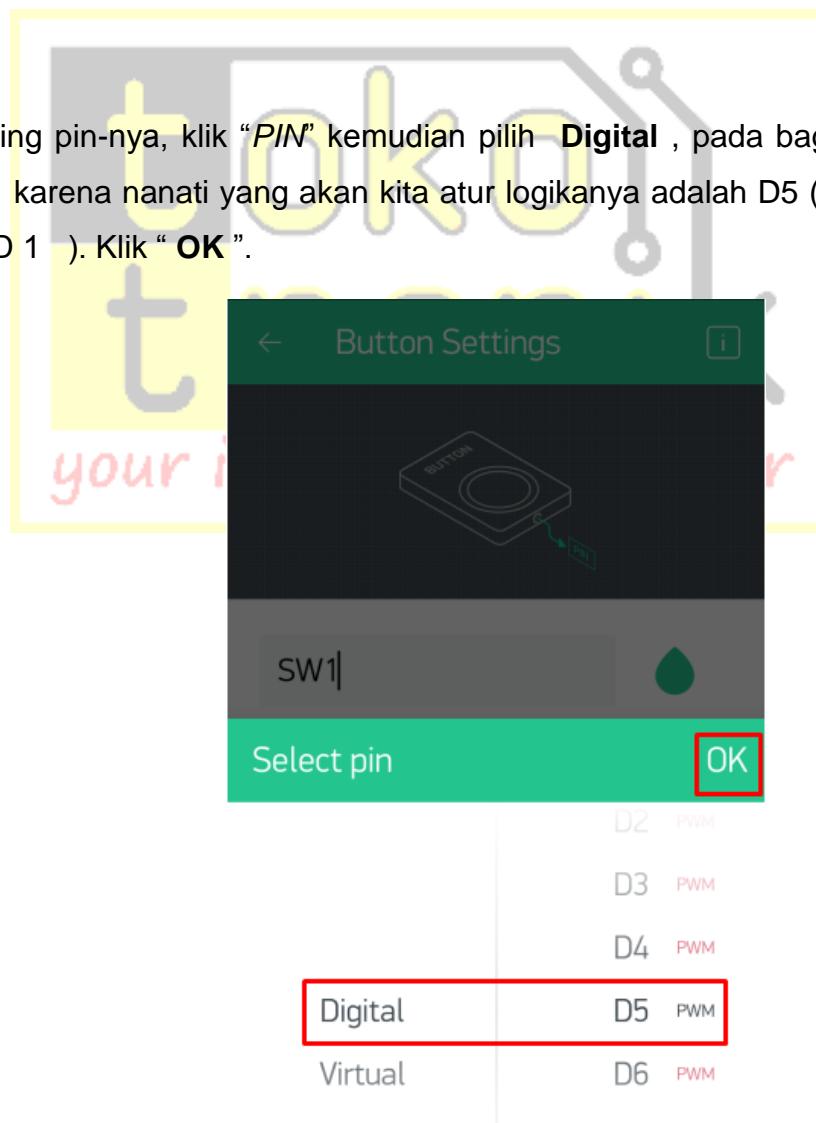
Kita perlu seting (atur) dulu Button-nya. Klik pada widget BUTTON sehingga muncul menu Button Settings



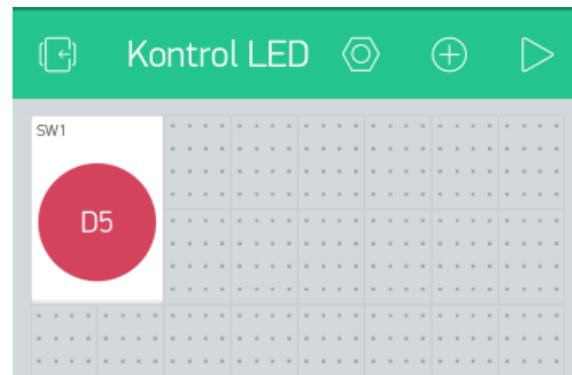
Pertama kita beri nama “**SW1**”



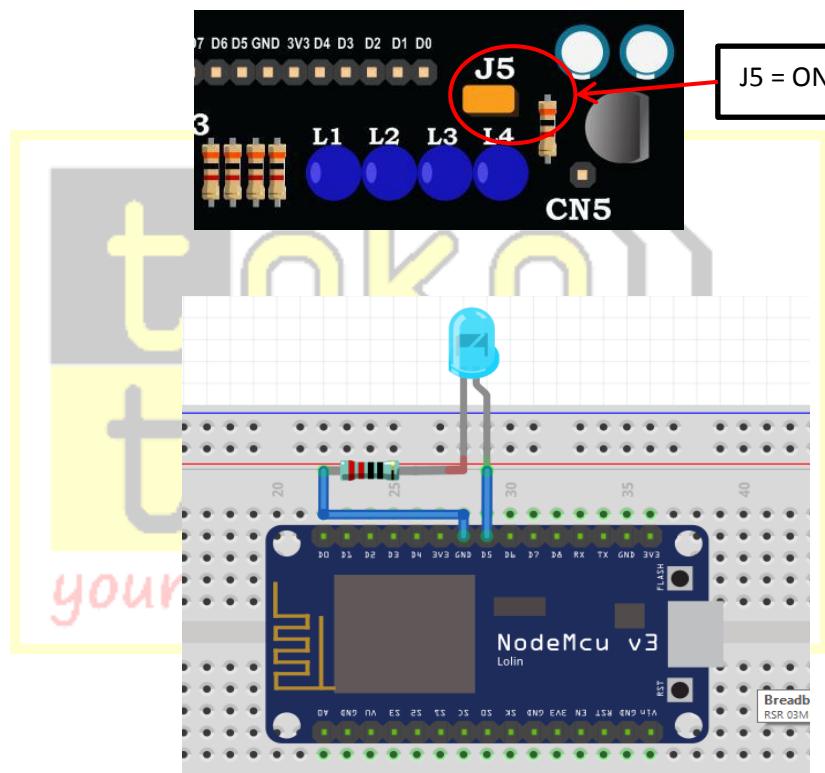
Setting pin-nya, klik “**PIN**” kemudian pilih **Digital**, pada bagian kanan pilih **D5** karena nanati yang akan kita atur logikanya adalah D5 (terhubung ke LED 1). Klik “**OK** ”.



Kembali ke tampilan utama. Nah aplikasi Kontrol LED sudah siap digunakan.



Hardware yang dibutuhkan berupa LED1, [pasang jumper J5](#) sehingga rangkaian seperti aplikasi 1. LED1 terhubung dengan pin D5 (aktif HIGH).



Cek email anda untuk melihat kode Token nya. Catat ya..

Sekarang kita siapkan programnya. Buat lembar kerja baru di Arduino IDE (File → New) kemudian tulis program berikut. [Sesuaikan Token, SSD dan Password WiFi nya .](#)

```
*****
* Program :Project 10 Kontrol LED via Blynk
* Input :
* Output : LED di D5
```

```
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

// Ganti dengan tokedn anda yang dikirim via email.
char auth[] = "ee2ba9a5boaa438cb697c4f9846e4f5f";

// Ganti dengan WiFi dan Password anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi1";

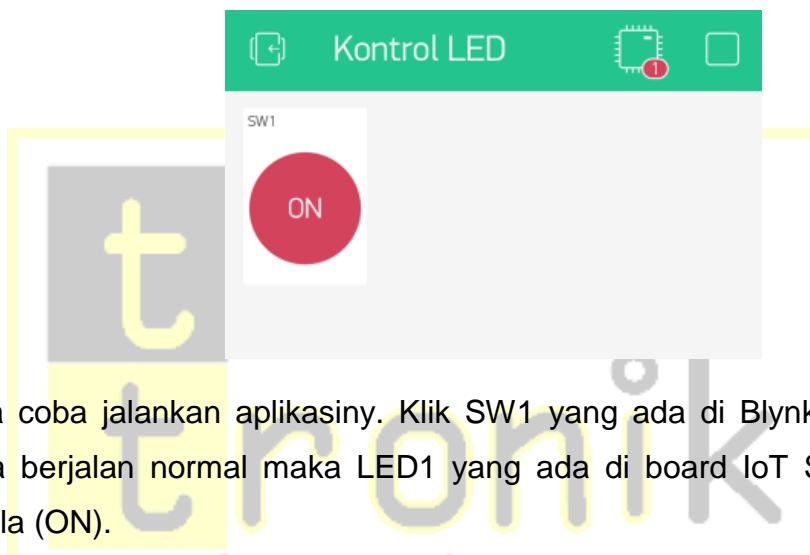
void setup()
{
    Serial.begin(9600);
    Blynk.begin(auth, ssid, pass);
}

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

Save programnya kemudian *Upload*. Tunggu sampai selesai. Jika sudah terhubung dengan jaringan WiFi, anda buka aplikasi Kontrol LED di BLYNK yang tadi sudah kita buat. Tekan tombol “Run” segitiga di pojok kanan atas.



Jika sudah tersambung akan tampil seperti ini :

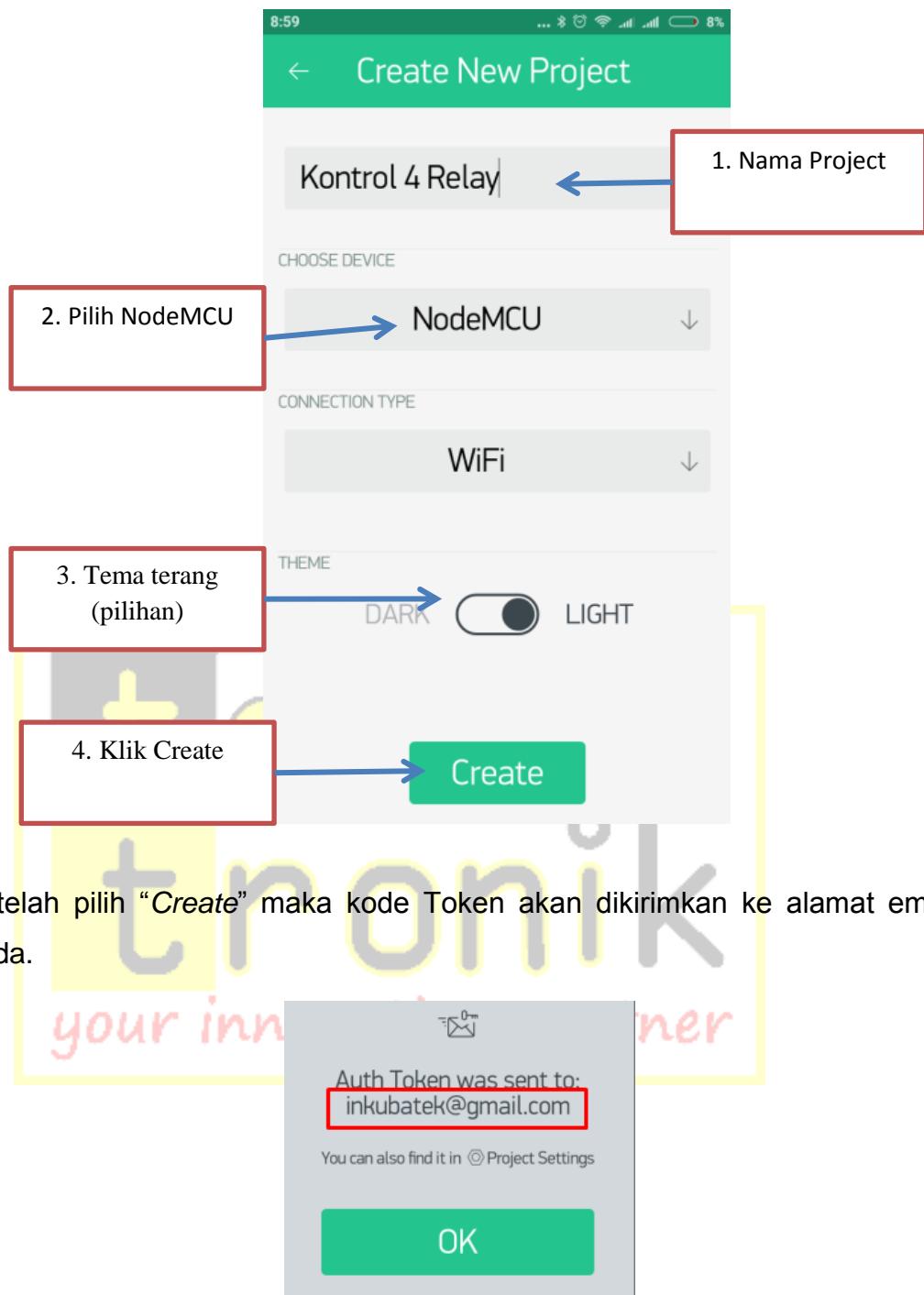


Kita coba jalankan aplikasinya. Klik SW1 yang ada di Blynk Android anda. Jika berjalan normal maka LED1 yang ada di board IoT Starter Kit akan nyala (ON).

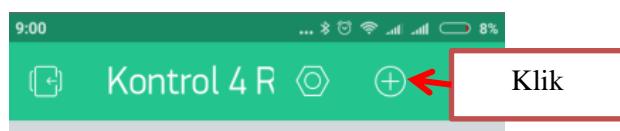
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XVI. Project IoT 11 : Kontrol 4 Relay dengan Aplikasi Android

Buat proyek baru di *Blynk* (caranya sama dengan proyek 10) beri nama : **Kontrol 4 Relay**.

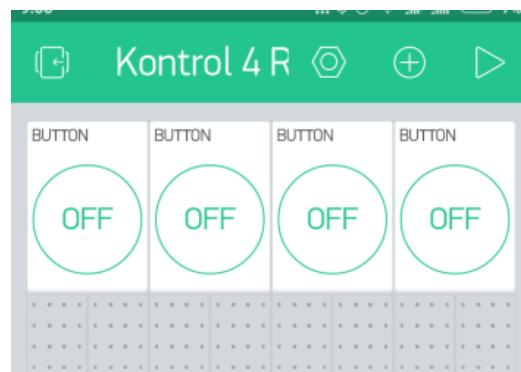


Catat dan simpan. Selanjutnya tambahkan 4 buah tombol (widget Button) dengan cara klik tanda '+'.

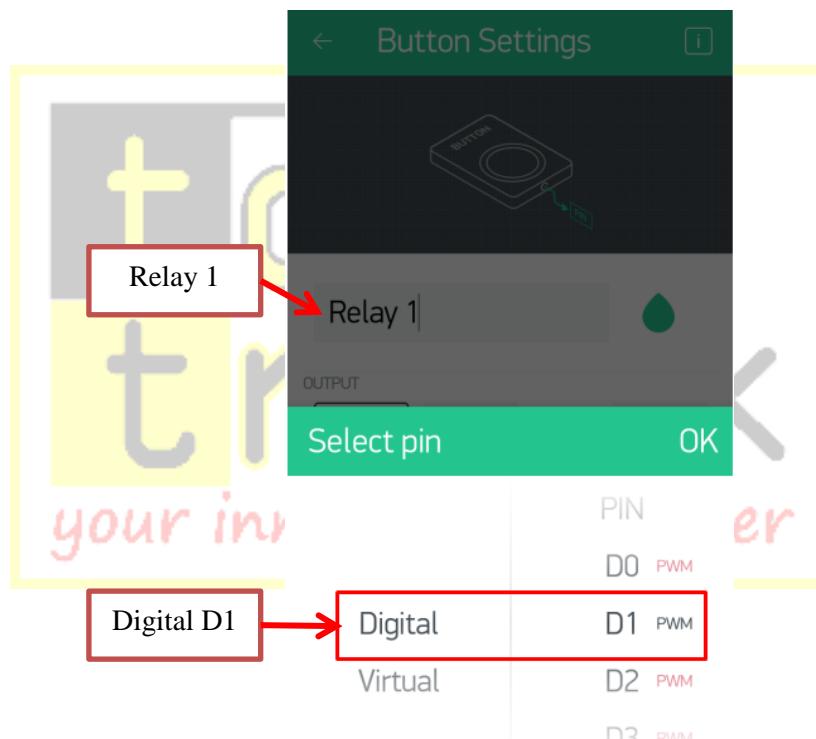


IoT Starter Kit Inkubatek

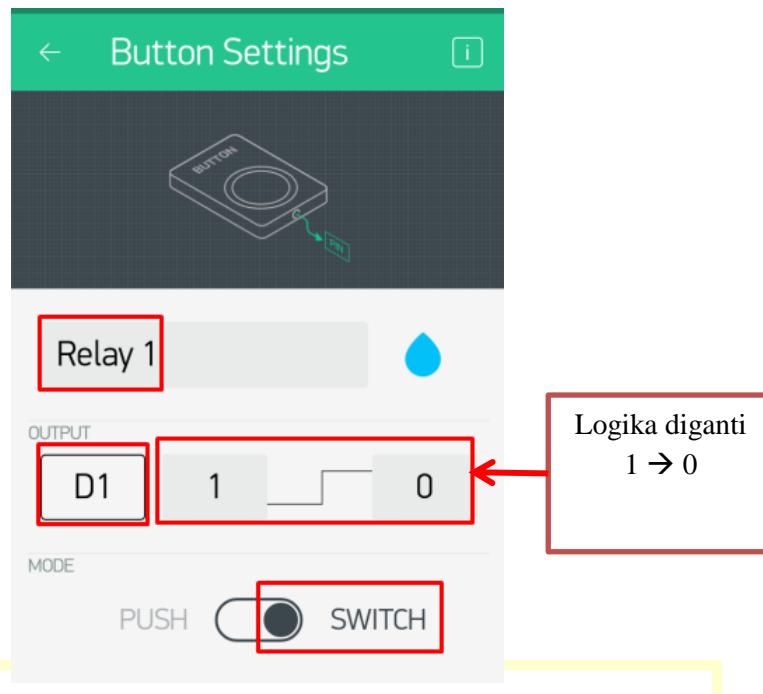
Buat 4 tombol (Button) dan atur sehingga menjadi seperti ini.



Edit/Seting masing – masing Button (klik Button-nya). Kita mulai dari Button 1, klik pada widget BUTTON 1 (paling kiri)

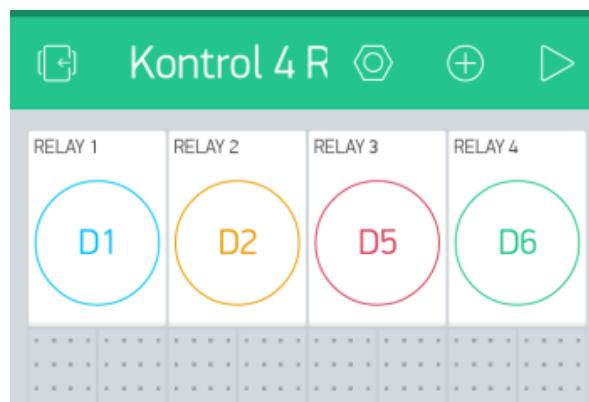


Beri nama “**Relay 1**”, kemudian klik pada bagian OUTPUT. Pilih **Digital D1** (Relay 1 terhubung ke pin D1 NodeMCU) kemudian klik **OK**.



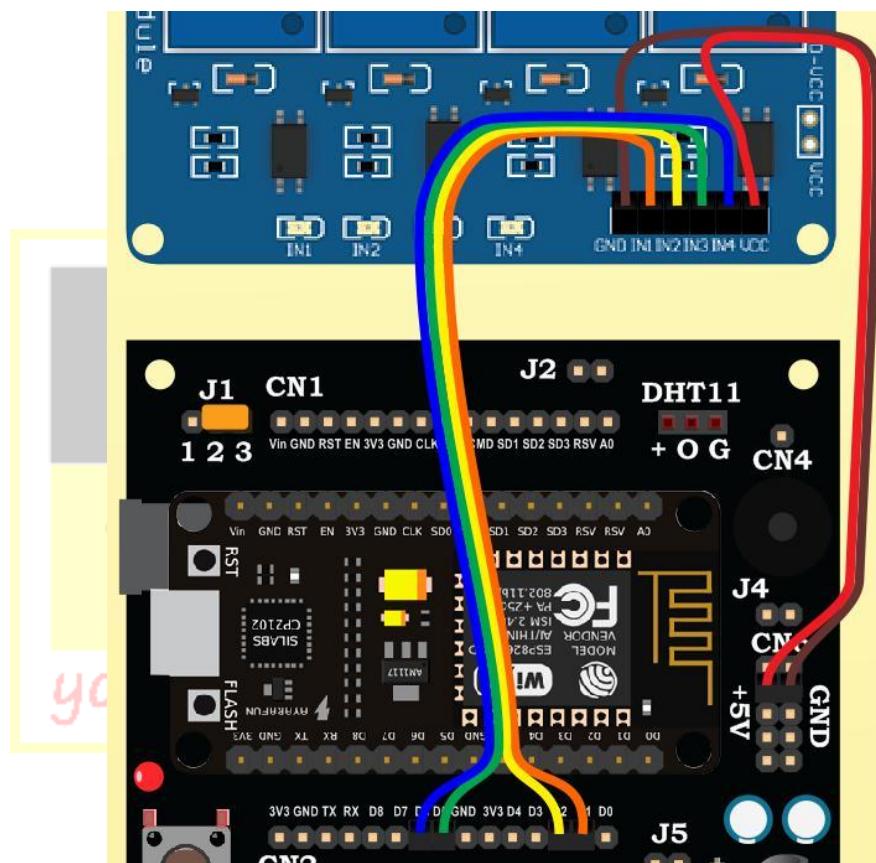
Logika juga diganti karena relay-nya aktif LOW (jika input diberi logika “LOW” relay akan “ON” dan sebaliknya). MODE diganti ke SWITCH. Lakukan untuk 3 BUTTON yang lain dengan parameter yang sama (logika diganti $1 \rightarrow 0$ dan MODE SWITCH).

BUTTON	Label	OUTPUT
2	Relay 2	D2
3	Relay 3	D5
4	Relay 4	D6



Koneksi modul 4 relay dengan modul IoT Starter Kit Inkubatek :

Modul Relay	IoT Starter Kit
VCC	VCC
GND	GND
IN1	D1
IN2	D2
IN3	D5
IN4	D6



Sekarang kita siapkan programnya di Arduino IDE. Ganti variabel ssid, password dan token.

```
*****
* Program :Project 11 Kontrol 4 Relay via Blynk
* Input :
* Output : Relay
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****
```

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

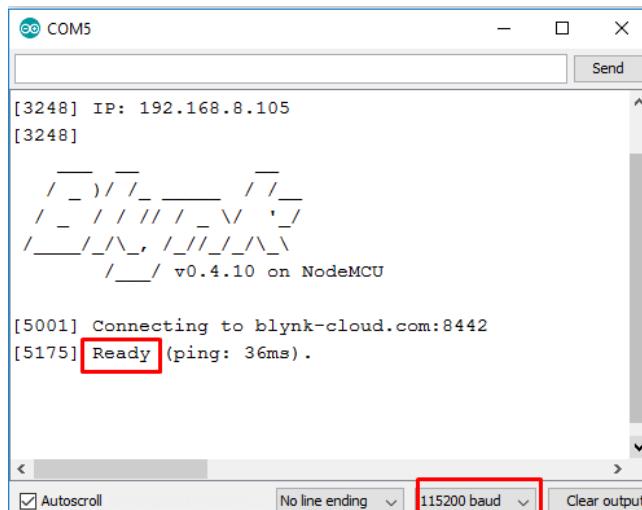
// Ganti dengan token anda yang dikirim via email.
char auth[] = "9654efc3a4f64d31b46cc17996905d61";

// Ganti dengan WiFi dan Password anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";

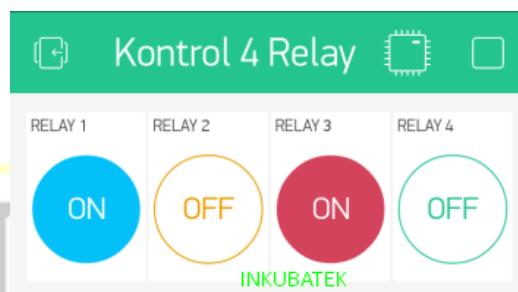
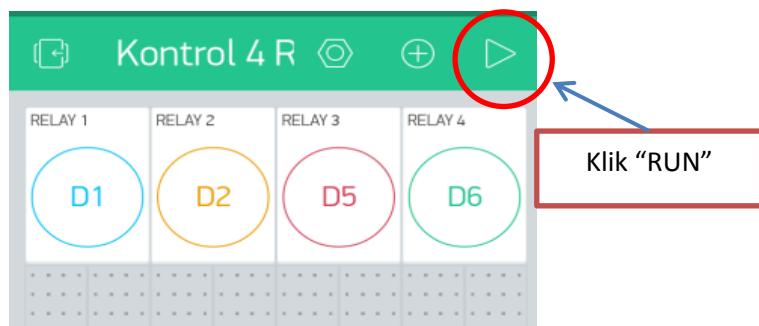
void setup()
{
    Serial.begin(115200);
    Blynk.begin(auth, ssid, pass);
}

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

Save programnya kemudian *Upload*. Tunggu sampai selesai. Buka Serial Monitor, jika sudah terhubung dengan WiFi tampilannya :



Kembali ke Project Kontrol 4 Relay di *Blynk* Android anda. Klik tombol “Run” untuk menjalankannya.

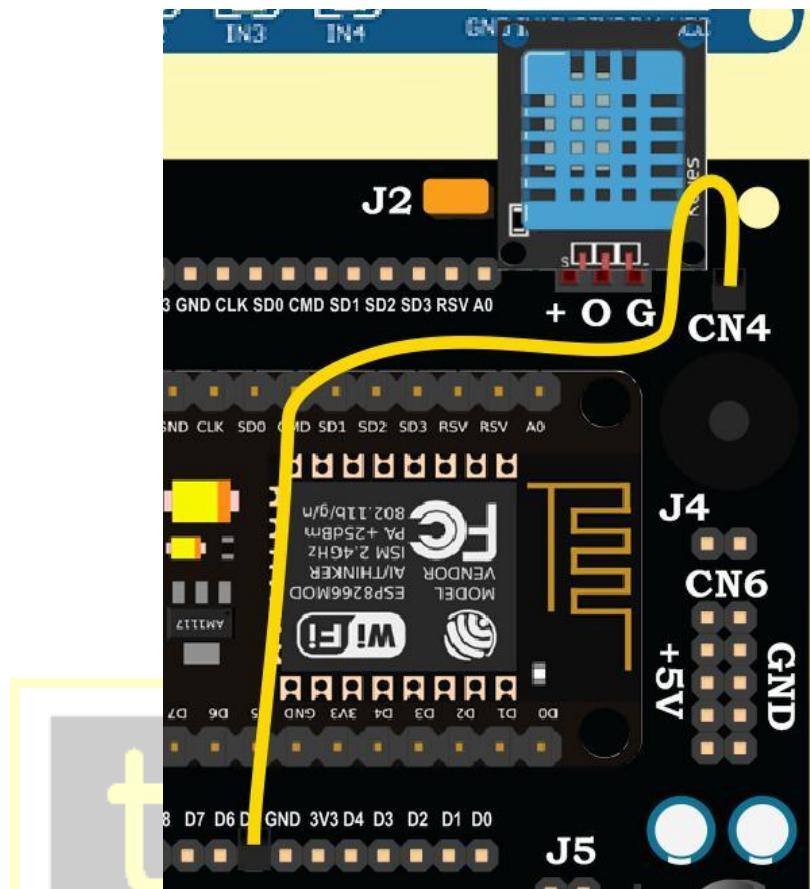


Tekan tombol Relay 1 maka relay 1 akan ON, coba juga untuk mengontrol relay yang lain. Jika belum jalan cek lagi koneksi kabel dan juga programnya.

[Jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD \]](#)

XVII. Project IoT 12 : Monitoring DHT11 dg Tampilan Grafis

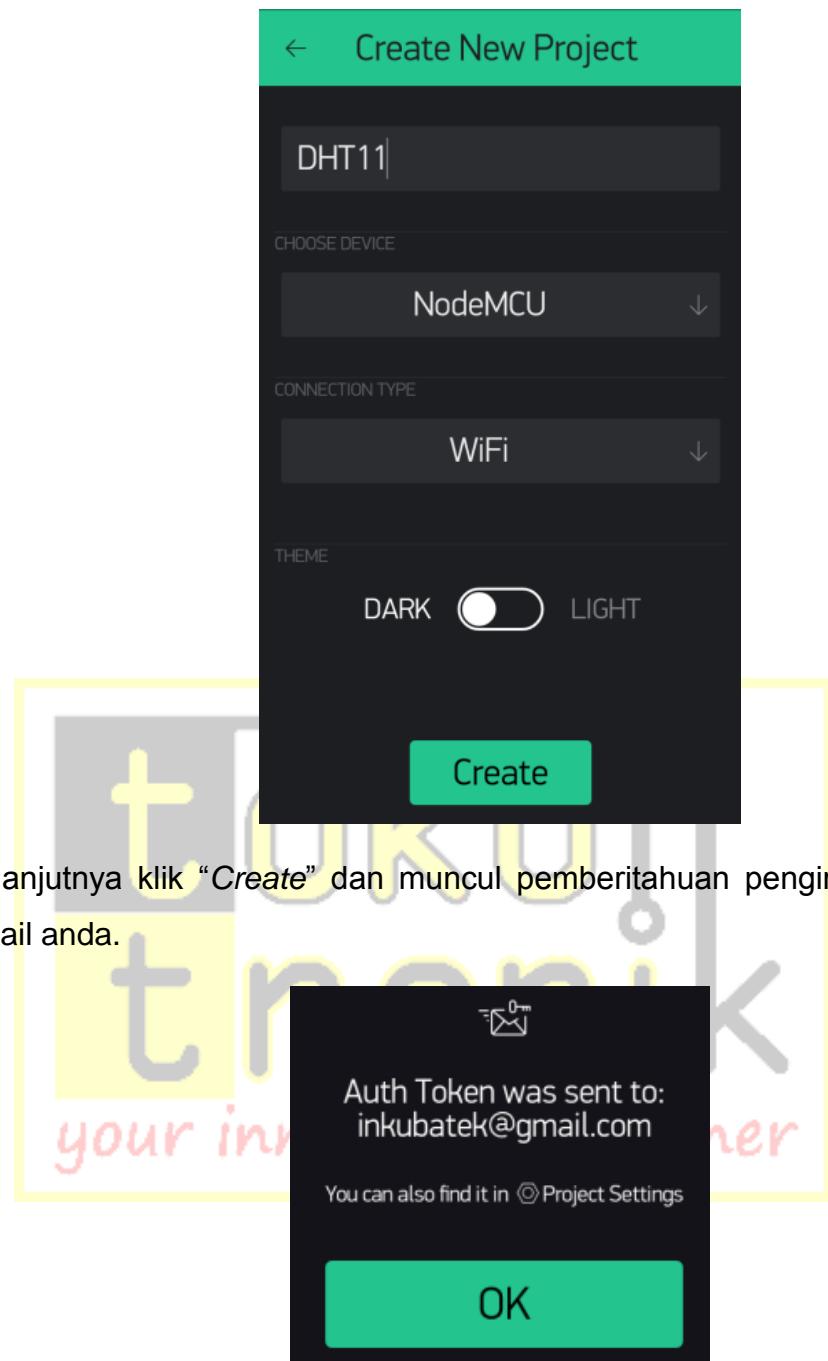
Masih setia dengan *Blynk* ya ... hehe... Kita lanjutkan membuat aplikasi monitoring suhu dan kelembaban dengan sensor DHT11. Rangkaian sama dengan Project IoT 5.



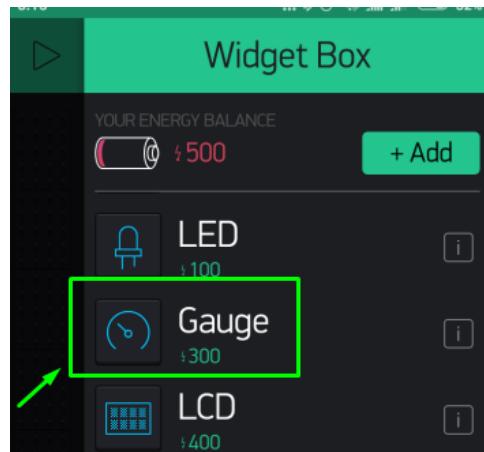
Koneksikan CN4 (output sensor DHT11) dengan pin D5. Jumper J5 dilepas/OFF.

Sensor DHT11	IoT Starter Kit
VCC	3V
OUT	D5
GND	GND

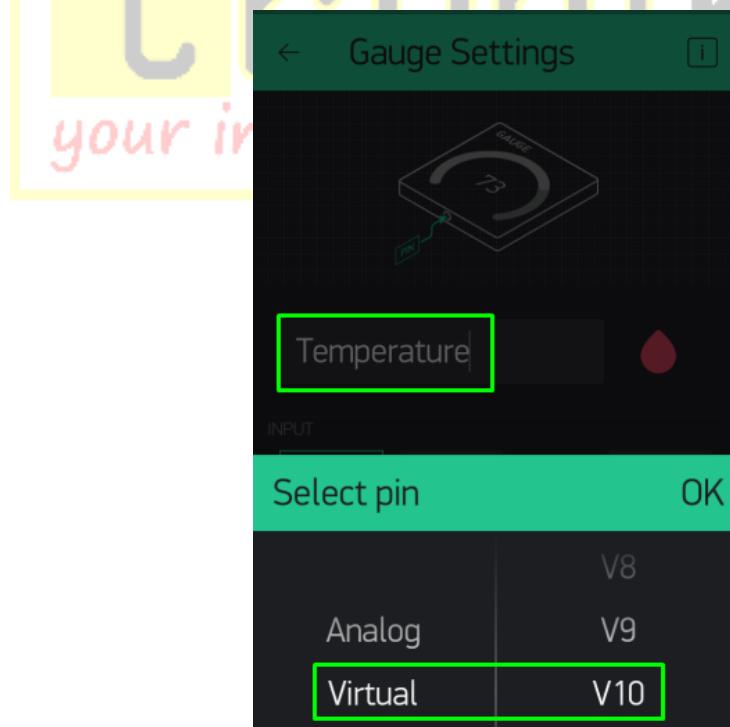
Sekarang kita buka *Blynk* kemudian buat proyek baru. Beri nama **DHT11**. CHOOSE DEVICE : *NodeMCU*.



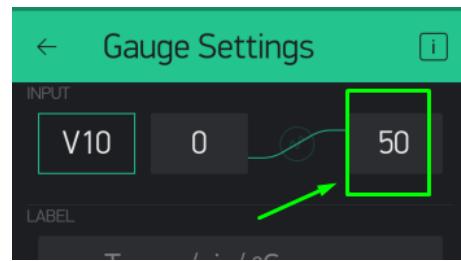
Selanjutnya tambahkan 2 buah "Gauge" di Widget Box (caranya sama dengan proyek ke-10)



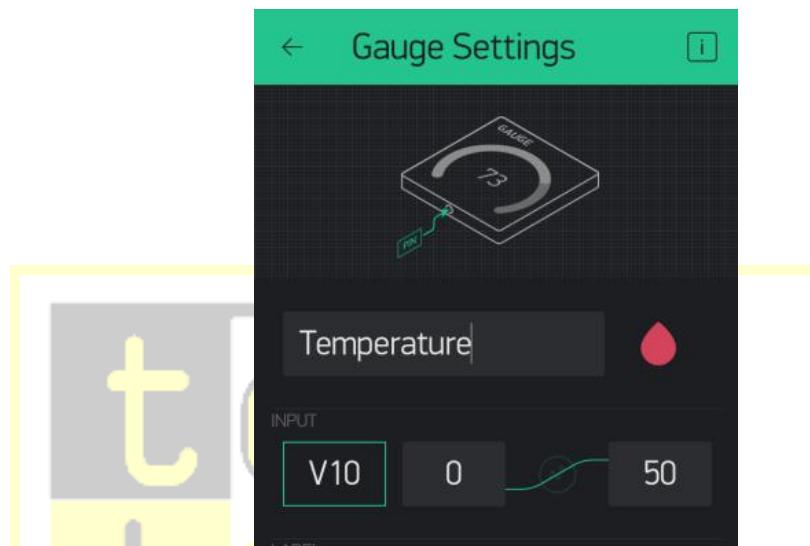
Setting Widget Gauge 1 untuk temperature (suhu). Klik pada widget "GAUGE"



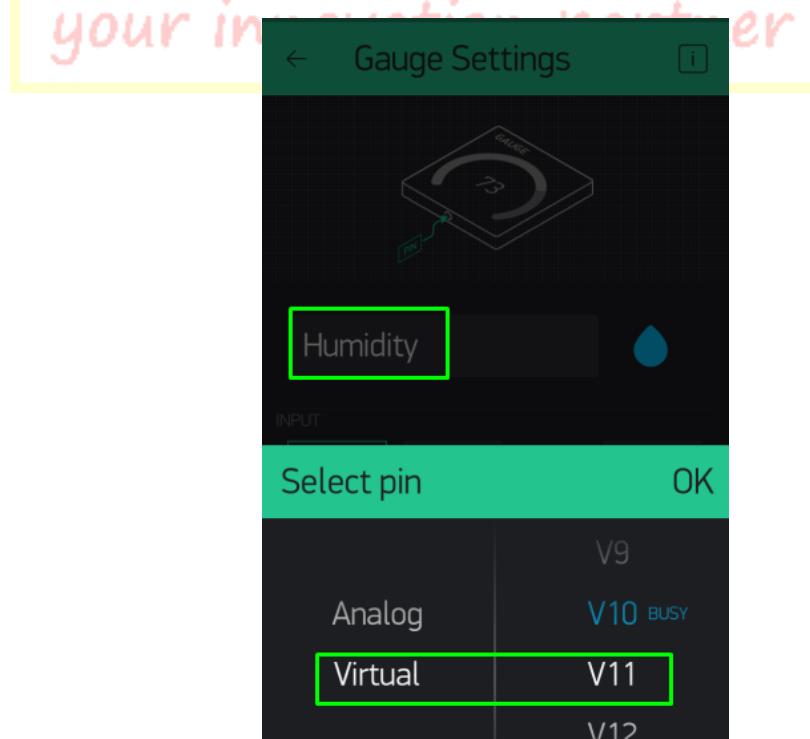
Beri label/nama “Temperature” trus klik INPUT,pilih **Virtual V10**.



Klik di kotak sebelah kanan INPUT, isikan nilai maksimal 50, hal ini karena pembacaan suhu maksimal DHT11 adalah 50°C .

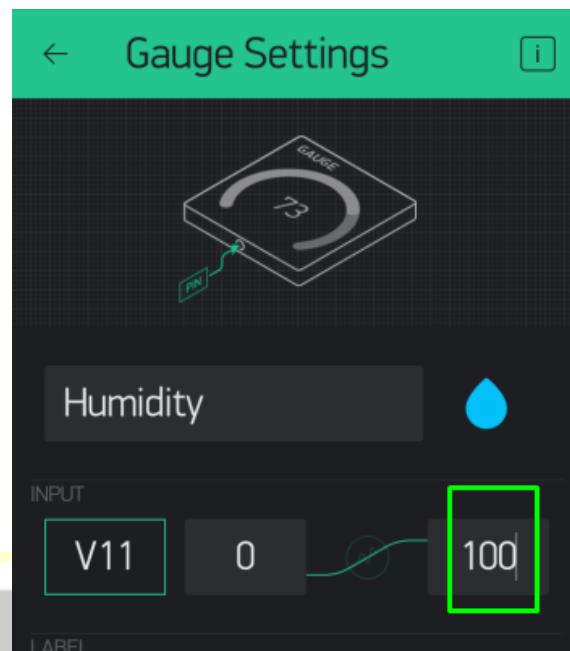


Kembali ke tampilan utama, sekarang kita setting Gauge2, klik pada widget Gauge 2.



IoT Starter Kit Inkubatek

Beri label/nama “**Humidity**” nantinya untuk tampilan nilai kelembaban.
Setting inputnya : Virtual V11 kemudian klik OK



Atur nilai maksimalnya menjadi 100.

Selanjutnya kita buat program di Arduino IDE. **Ganti Token, ssid dan password WiFi-nya.**

```
*****
* Program :Project 12 Monitoring DHT11 Blynk
* Input : DHT11 di pin D5
* Output : Relay
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
#include <SimpleTimer.h>
#define DHTPIN D5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
```

```
SimpleTimer timer;

float humi, temp;
// Ganti dengan kode token anda
char auth[] = "123a369f2894c55b219aa4def71346e";
// Ganti dengan WiFi dan Password anda
char ssid[] = "FirnasAbe";
char pass[] = "hamasfahreza";

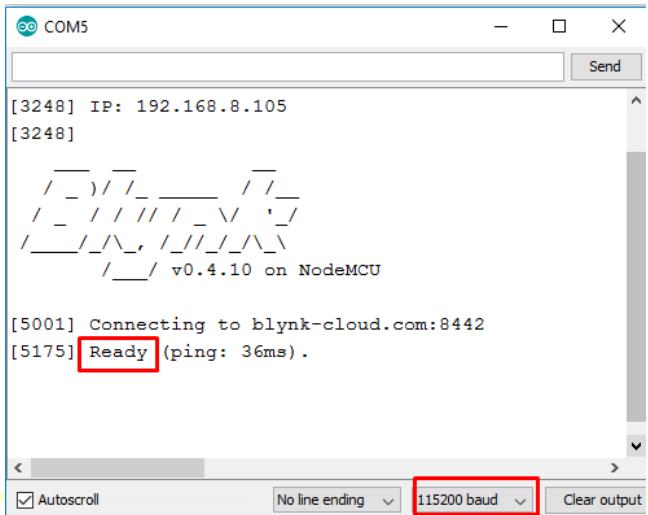
void setup()
{
    Serial.begin(115200);
    delay(10);
    Blynk.begin(auth, ssid, pass);
    timer.setInterval(1000, sendData);
}

void sendData()
{
    humi = dht.readHumidity();
    temp = dht.readTemperature();
    Blynk.virtualWrite(10, temp); //virtual input V10 di Blynk (suhu)
    Blynk.virtualWrite(11, humi); //virtual input V11 di Blynk (kelembaban)
}

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

IoT Starter Kit Inkubatek

Save programnya kemudian *Upload*. Tunggu sampai selesai. Jangan lupa untuk mengganti Token, ssid dan password WiFi. Buka Serial Monitor, jika sudah terhubung dengan WiFi tampilannya :

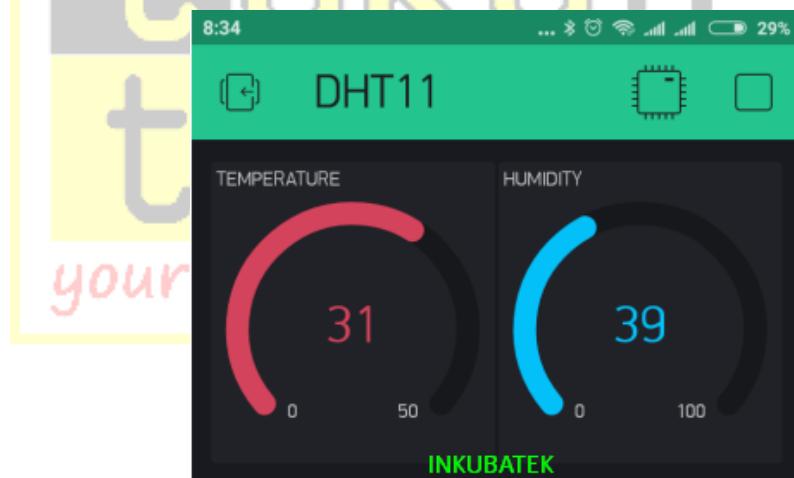


```
[3248] IP: 192.168.8.105
[3248]
/ \ ) / \
/ / / / \ ' /
/ / \ , / / / \
/ \ v0.4.10 on NodeMCU

[5001] Connecting to blynk-cloud.com:8442
[5175] Ready (ping: 36ms).
```

Autoscroll No line ending 115200 baud Clear output

Kembali ke Project DHT11 di Blynk Android anda. Klik tombol “Run” untuk menjalankannya.



Hasilnya tampak nilai temperatur dan kelembaban ditambah dengan tampilan grafis-nya.

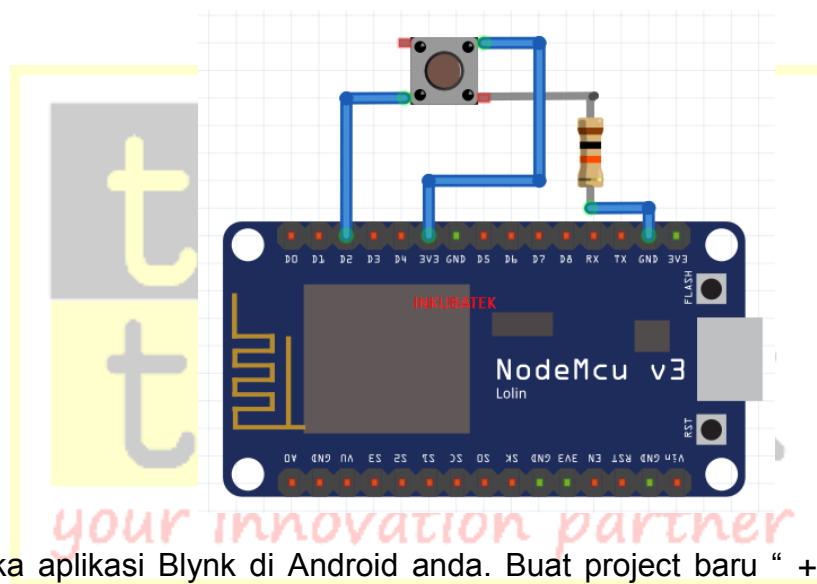
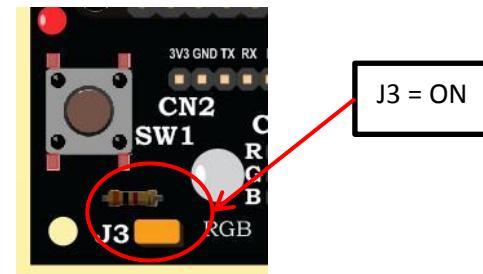
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XVIII. Project IoT 13 : Mengirim Notifikasi ke Email

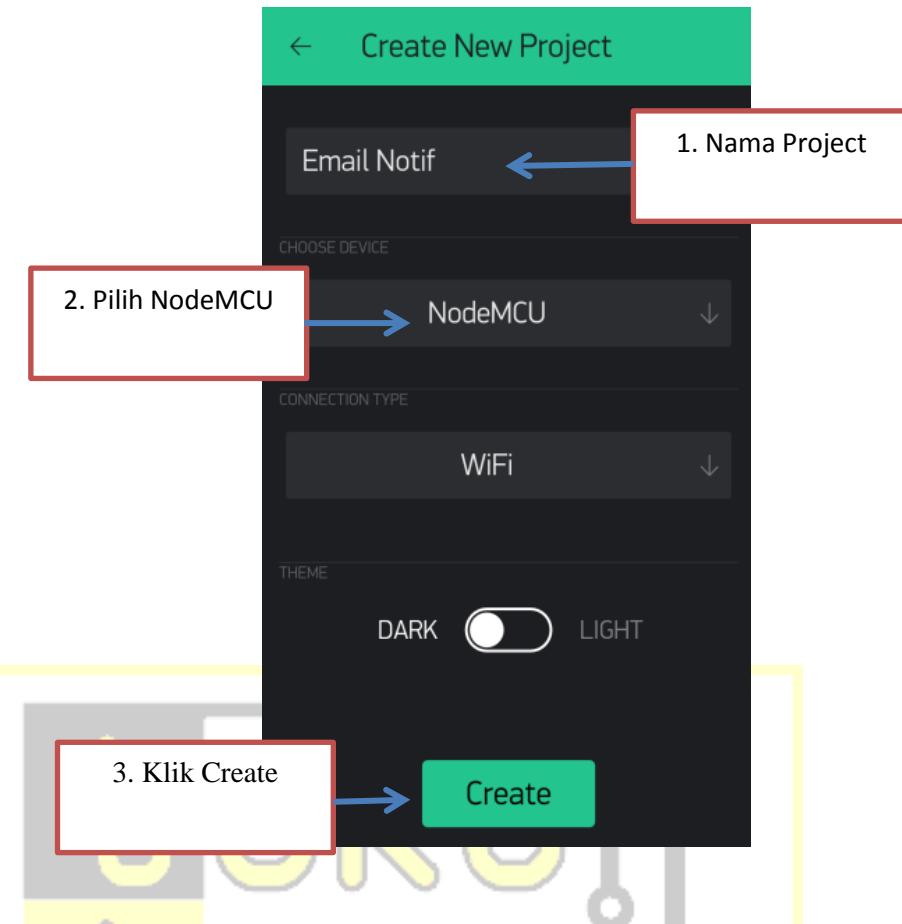
IoT Starter Kit Inkubatek

Kalau pada project sebelumnya kita mengirim notifikasi ke Telegram, kali ini kita akan mengirim notifikasi ke alamat email. Siapkan sebuah alamat email, misalnya disini saya pakai inkubatek@yahoo.com

Sebagai trigger kita pakai tombol SW1 yang ada di board IoT Starter Kit Inkubatek. **Pasang jumper J3 sehingga SW1 terhubung dengan pin D2.**

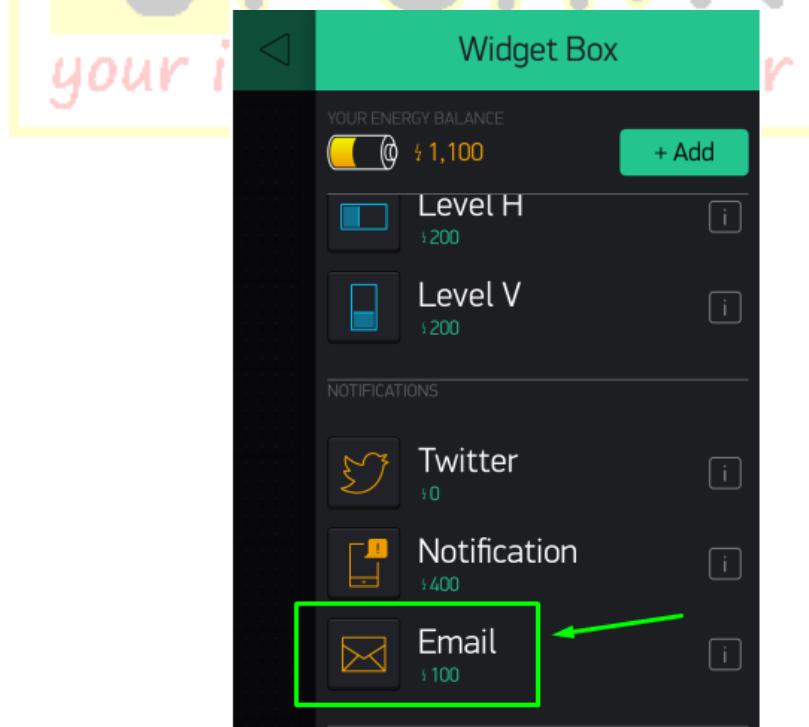


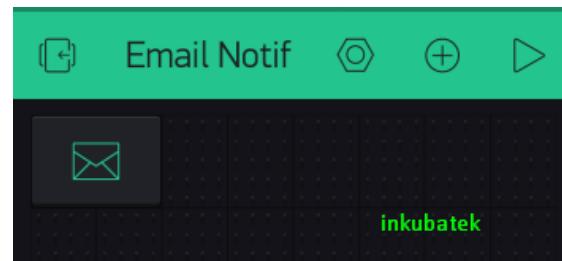
Buka aplikasi Blynk di Android anda. Buat project baru “ + New Project ” kemudian beri label/nama “Email Notif”



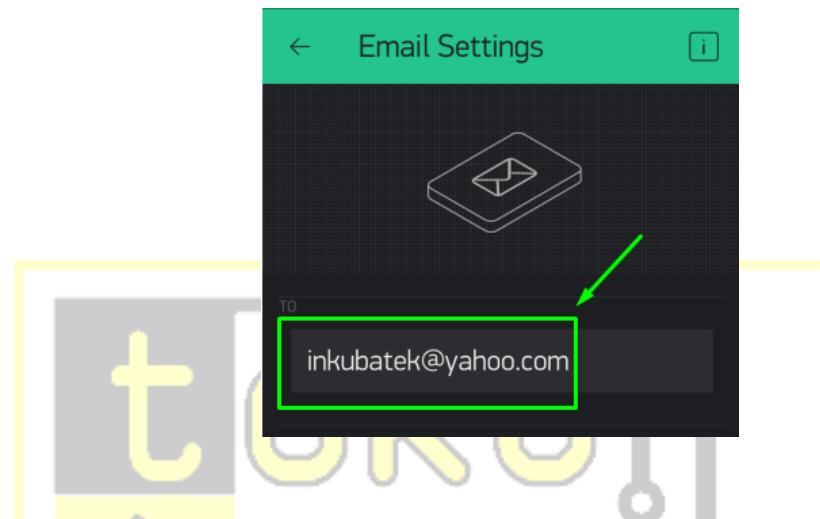
Setelah klik "Create" token akan dikirim ke email anda, cek dan catat.

Selanjutnya pada Widget Box tambahkan 1 buah email (*notification*).





Klik pada widget *Email* kemudian isikan sebuah alamat email yang nanti akan dikirim notifikasi, misalnya email saya tadi `inkubatek@yahoo.com`



Selanjutnya kita buat programnya di Arduino IDE. Buat lembar kerja baru (File → New) kemudian tulis program berikut. Jangan lupa mengganti **ssid, token, password dan email**.

```
*****
* Program :Project 13 Kirim Notifikasi Email
* Input : SW1 di pin D2
* Output : Email notification
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/
#define BLYNK_MAX_SENDBYTES 1200
#include <ESP8266WiFi.h>

#define BLYNK_PRINT Serial
#include <BlynkSimpleEsp8266.h>
//ganti dengan kode token yang ada terima di email
```

```
char auth[] = "777ef75e90344862a8705f9ded3f508d";
//ganti dengan jaringan dan password WiFi anda
char ssid[] = "FirnasAbe1";
char pass[] = "podowingi";
#define tombol D2
int tbValue;

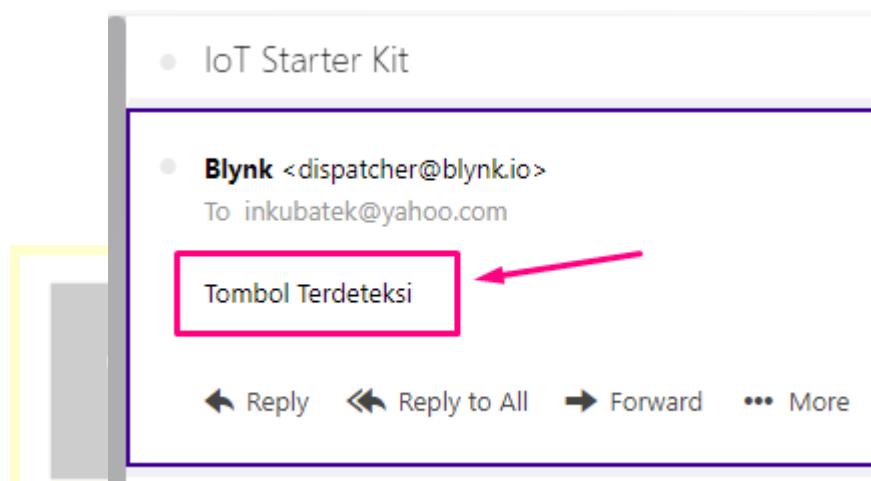
void setup()
{
    Serial.begin(115200);
    delay(10);
    Blynk.begin(auth, ssid, pass);
    pinMode(tombol, INPUT);
}

void loop()
{
    cek_tombol();
    Blynk.run();
}

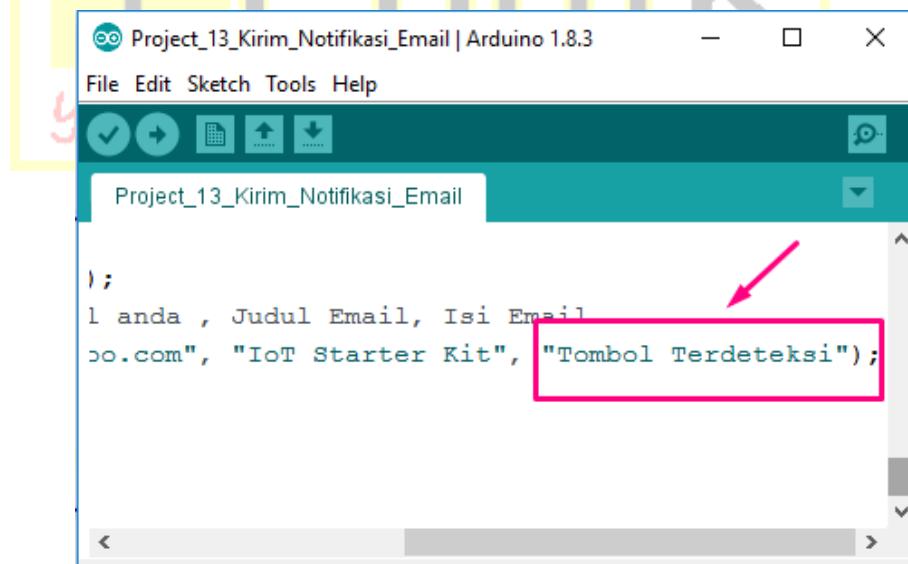
// ****
void cek_tombol(void)
{
    tbValue = digitalRead(tombol);
    if (tbValue)
    {
        Serial.println("Tombol ON");
        //ganti dengan alamat email anda , Judul Email, Isi Email
        Blynk.email("inkubatek@yahoo.com",      "IoT      Starter      Kit",      "Tombol
Terdeteksi");
        delay(1000);
    }
}
```

Jangan lupa untuk mengganti token (auth), ssid, password dan alamat emailnya.

Ok, Save kemudian *Upload*. Jika sudah sukses proses Uploadnya, jalankan aplikasi “Email Notif” yang di Blynk Android anda, klik “Run”. Untuk mencobanya tombol SW 1 di board **IoT Starter Kit** anda tekan, maka email akan terkirim ke alamat yang tadi dibuat.



Untuk mengganti isi pesan email cukup edit/ganti string :



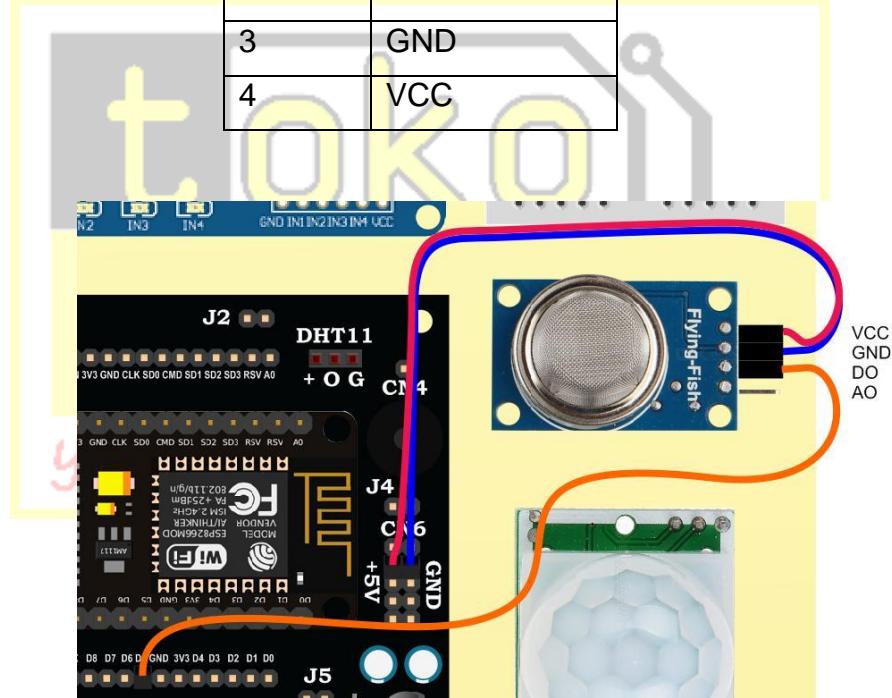
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XIX. Project IoT 14 : Early Warning System Kebocoran Gas

Pada project IoT Starter Kit Inkubatek ke 14 ini kita akan membuat alat pendeteksi kebocoran gas (LPG) dengan sensor MQ2.



Pin	Keterangan
1	A0
2	D0
3	GND
4	VCC



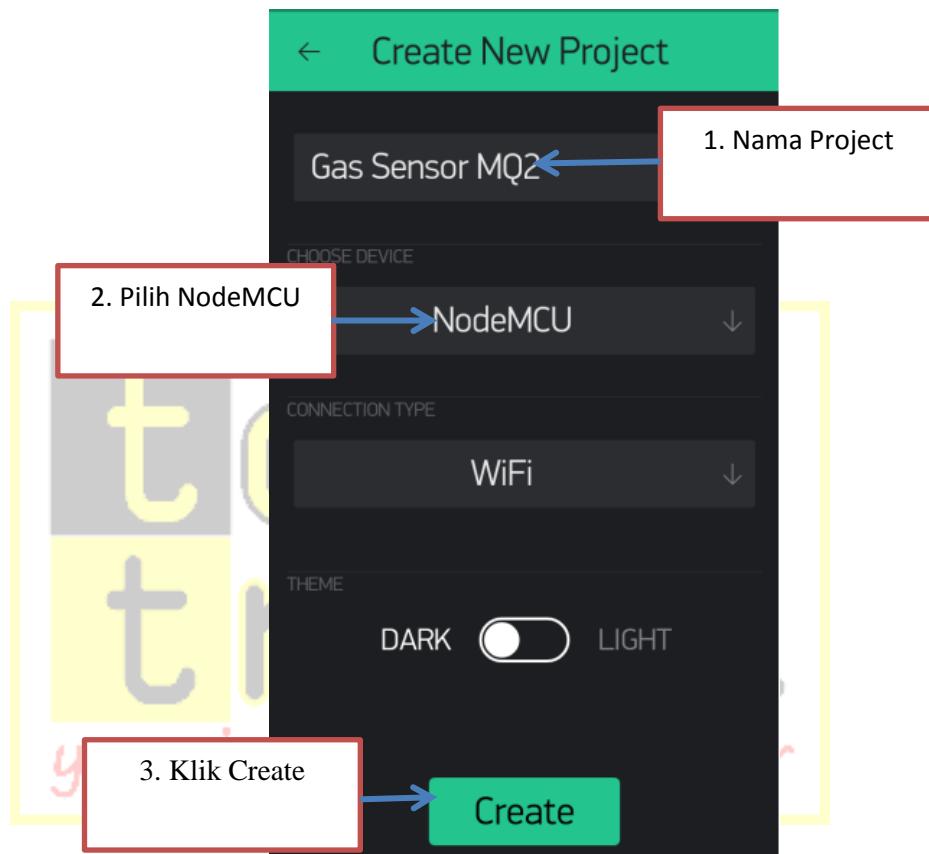
Pin D0 merupakan pin output digital dari sensor MQ2. Apabila tidak mendeteksi gas maka pin D0 berlogika "HIGH" dan jika mendeteksi adanya gas maka berlogika "LOW".

Koneksikan sensor MQ2 dengan IoT Starter Kit memakai kabel jumper (kabel pelangi) :

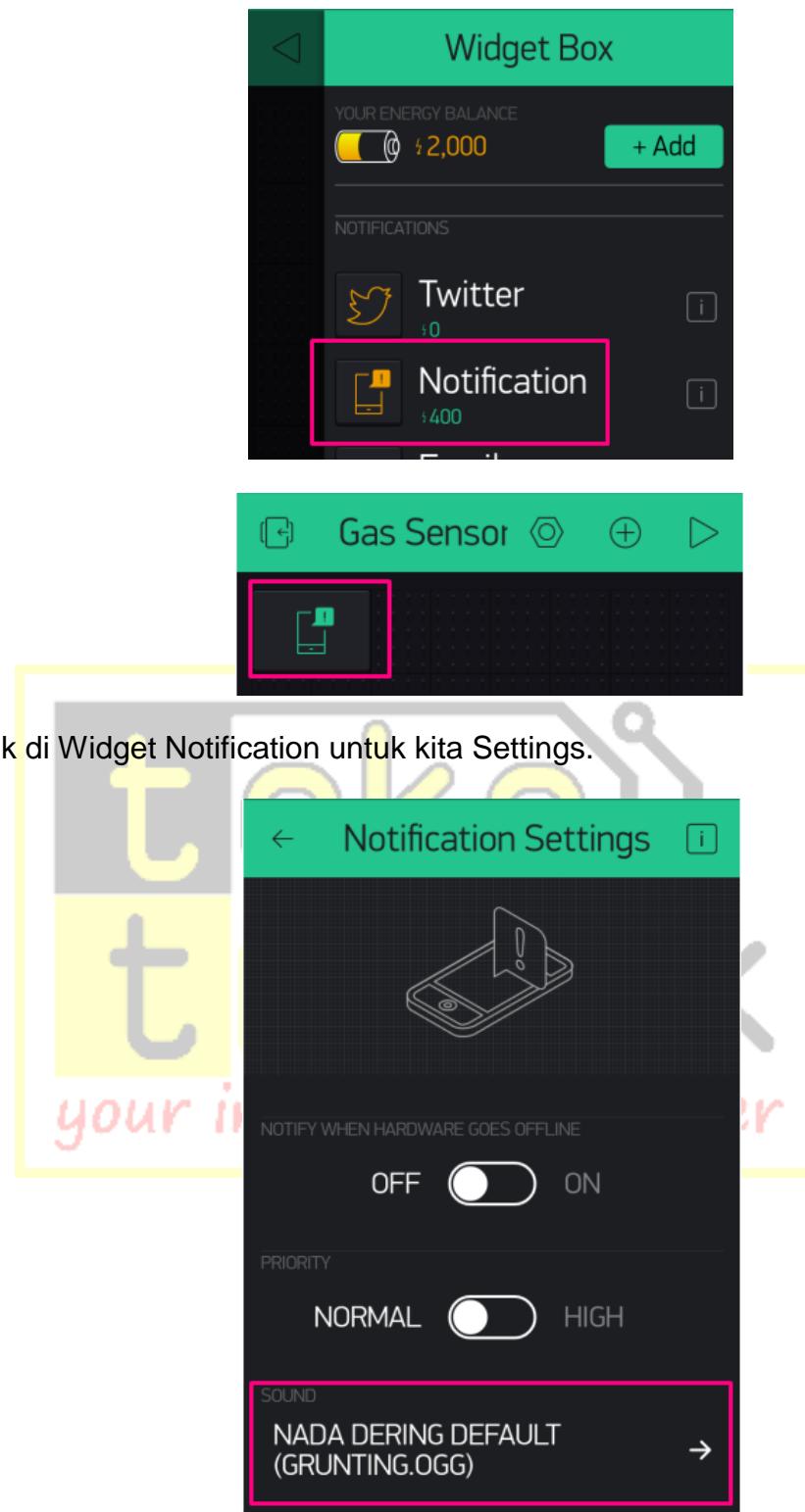
Sensor MQ2	IoT Starter Kit
A0	-

D0	D5
GND	GND
VCC	+5V

Kita buat aplikasinya di Blynk Android. Seperti biasa buta proyek baru “+New Project”. Beri nama/label **“Gas Sensor MQ2”**



Klik tombol “Create” kemudian cek di email anda kode Token-nya. Catat untuk pembuatan program nanti. Tambahkan sebuah Notification di Widget Box.



Untuk memilih nada dering (suara alarm) ketika notifikasi aktif, anda dapat mengganti suara/nada dering dengan memilih ‘sound’, klik di menu ‘ SOUND ’ kemudian pilih nada yang dikehendaki.

Sekarang kita siapkan program di Arduino IDE nya. **Ganti auth token, ssid dan passwordnya.**

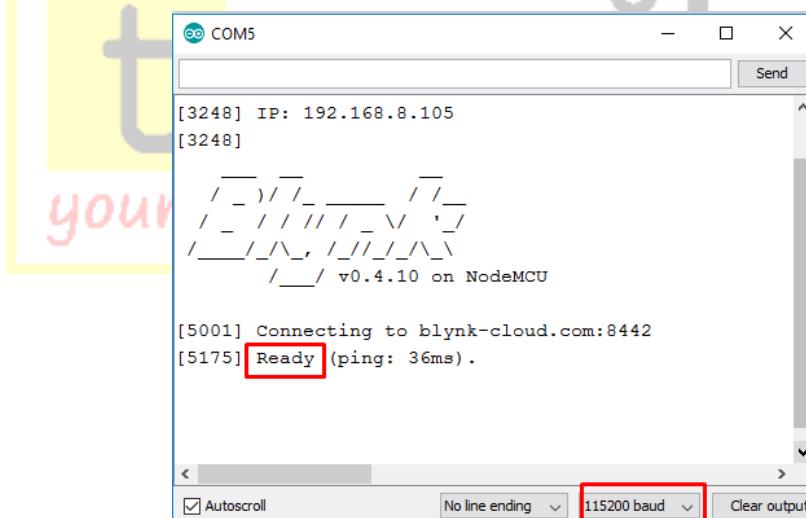
```
*****  
* Program :Project 14 Deteksi Kebocoran Gas  
* Input : MQ2 di pin D5  
* Output : Email notification  
* IoT Starter Kit Inkubatek  
* www.tokotronik.com  
* *****  
#include <ESP8266WiFi.h>  
#define BLYNK_PRINT Serial // Comment this out to disable prints and save space  
#include <BlynkSimpleEsp8266.h>  
char auth[] = "1928e00202647eb833bc78cb26b9d84";  
  
/* Ganti dengan ssid dan password WiFi anda */  
char ssid[] = "FirnasAbe";  
char pass[] = "podowingi";  
  
#define MQ2Pin D5  
int MQ2Value;  
  
void setup()  
{  
    Serial.begin(115200);  
    delay(10);  
    Blynk.begin(auth, ssid, pass);  
    pinMode(MQ2Pin, INPUT);  
}  
  
void loop()  
{  
    getMQ2Value();  
    Blynk.run();  
}
```

```
// ****
void getMQ2Value(void)
{
    MQ2Value = digitalRead(MQ2Pin);
    if (!MQ2Value)
    {
        Serial.println("==>Gas terdeteksi");
        Blynk.notify("Kebocoran Gas terdeteksi.....!");
        delay(1000);
    }
}
```

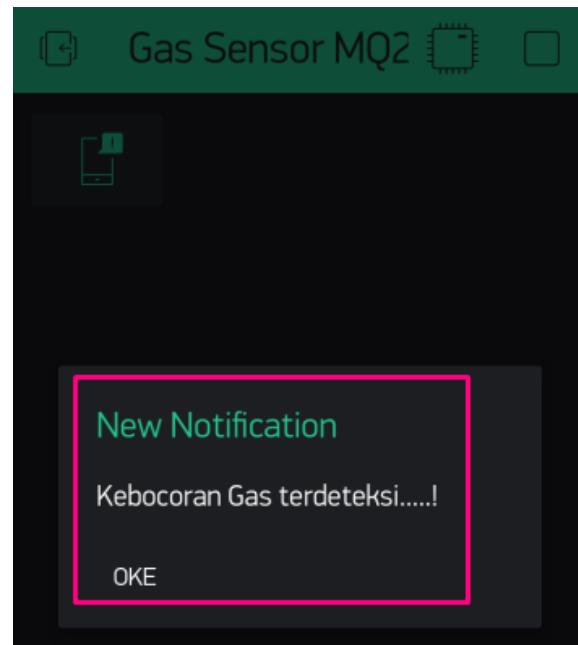
Sesuaikan Token, SSID dan Password WiFi nya.

Pada hardware modul IoT Starter Kit Inkubatek anda lepas J5/OFF.

Save programnya kemudian *Upload*. Tunggu sampai selesai. Buka Serial Monitor, jika sudah terhubung dengan WiFi tampilannya :



Kembali ke Sensor Gas MQ2 di Blynk Android anda. Klik tombol “Run ” untuk menjalankannya coba beri gas LPG di sensor MQ2, atau gas dari korek api gas. Jika berhasil maka di aplikasi Blynk Android tadi akan muncul notifikasi + suaranya.



Jika ada ingin mengganti isi pesan maka cukup edit/ganti :

A screenshot of the Arduino IDE. The menu bar shows "File Edit Sketch Tools Help". The title bar says "Project_14_Early_Warning_Gas". The code editor contains the following C-like pseudocode:

```
// *****
void getMQ2Value(void)
{
    MQ2Value = digitalRead(MQ2Pin);
    if (!MQ2Value)
    {
        Serial.println("==>Gas terdeteksi");
        Blynk.notify "Kebocoran Gas terdeteksi.....!";
        delay(1000);
    }
}
```

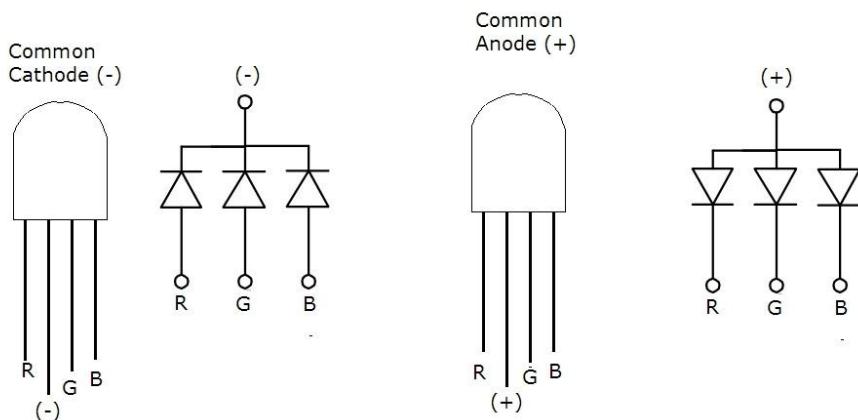
An arrow points from the text "Blynk.notify" in the code to a yellow box containing the text "Kebocoran Gas terdeteksi.....!". Both the code and the yellow box are highlighted with a pink rectangular border.

[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XX. Project IoT 15 : Kontrol LED RGB via Android

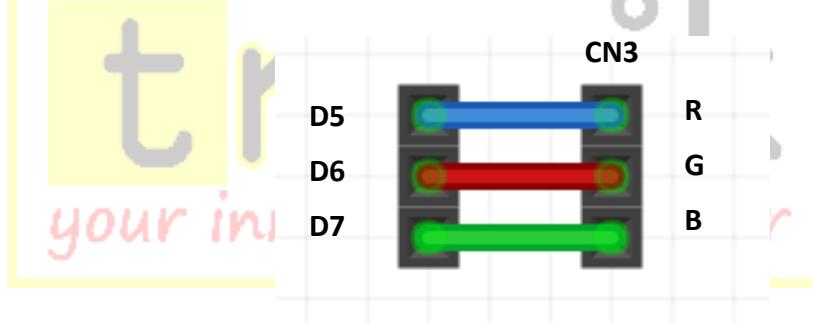
IoT Starter Kit Inkubatek

Led RGB merupakan LED yang dapat menampilkan 3 warna dalam 1 LED : Merah (R), Hijau (G) dan Biru (B). Kombinasi dari 3 warna tersebut akan menghasilkan warna LED yang berwarna – warni.



Nah nilai tegangan (PWM nantinya) di setiap warna tersebut dapat kita kontrol sehingga menampilkan perpaduan warna diantara ke-3 warna dasar tersebut.

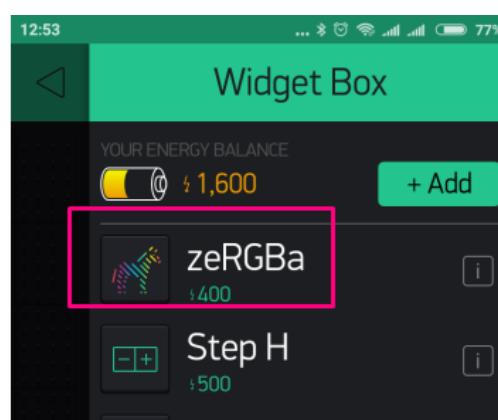
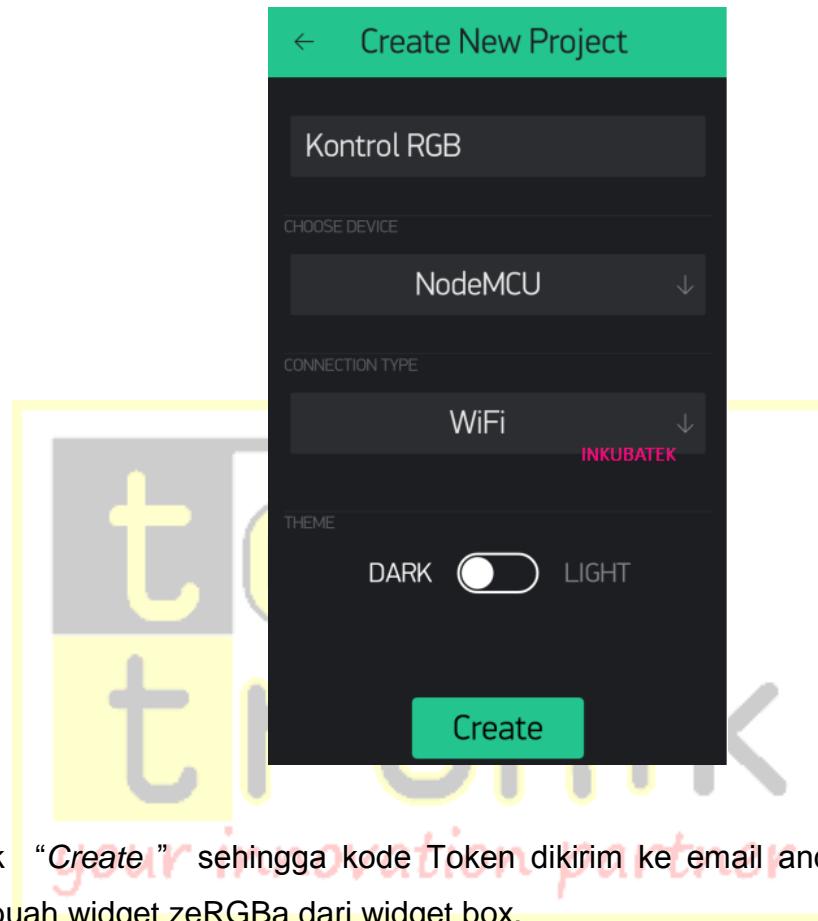
Pada board IoT Starter Kit Inkubatek sudah dilengkapi dengan LED RGB. Pasang konektor CN3 ke pin D5 – D7 seperti rangkaian.

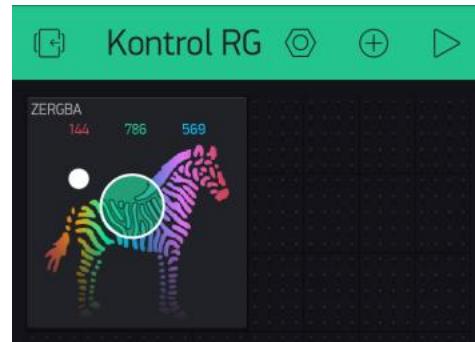


LED RGB	IoT Starter Kit
R	D5
G	D6

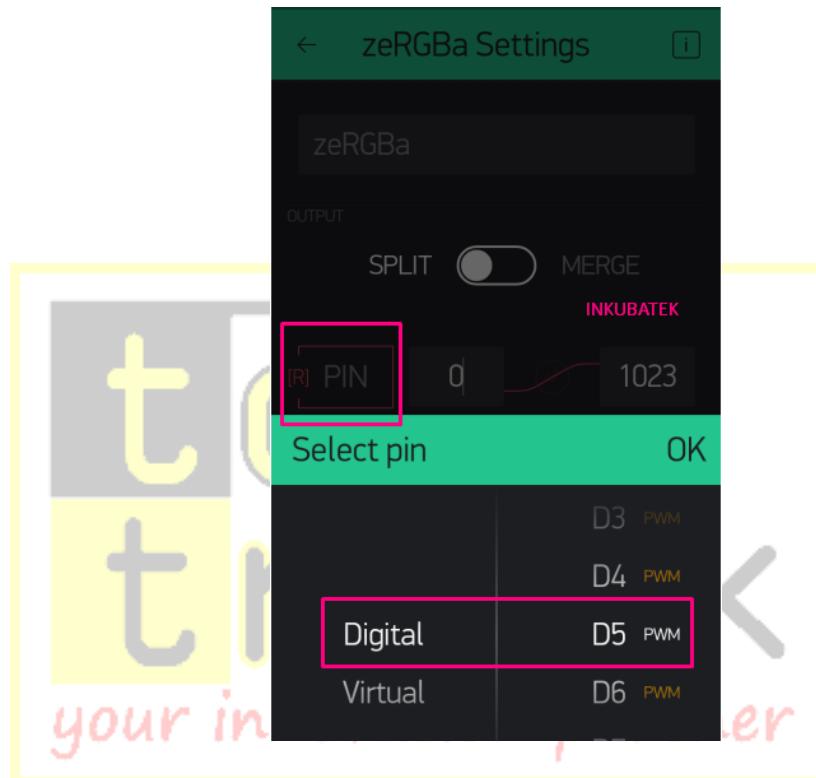
B	D7
---	----

Kita masih memakai Blynk. Bikin proyek baru, *New Project* kemudian beri nama **Kontrol RGB** , DEVICE-nya NodeMCU dan konesinya WiFi. ([cara membuat aplikasi Blynk sama dengan proyek 10](#)).





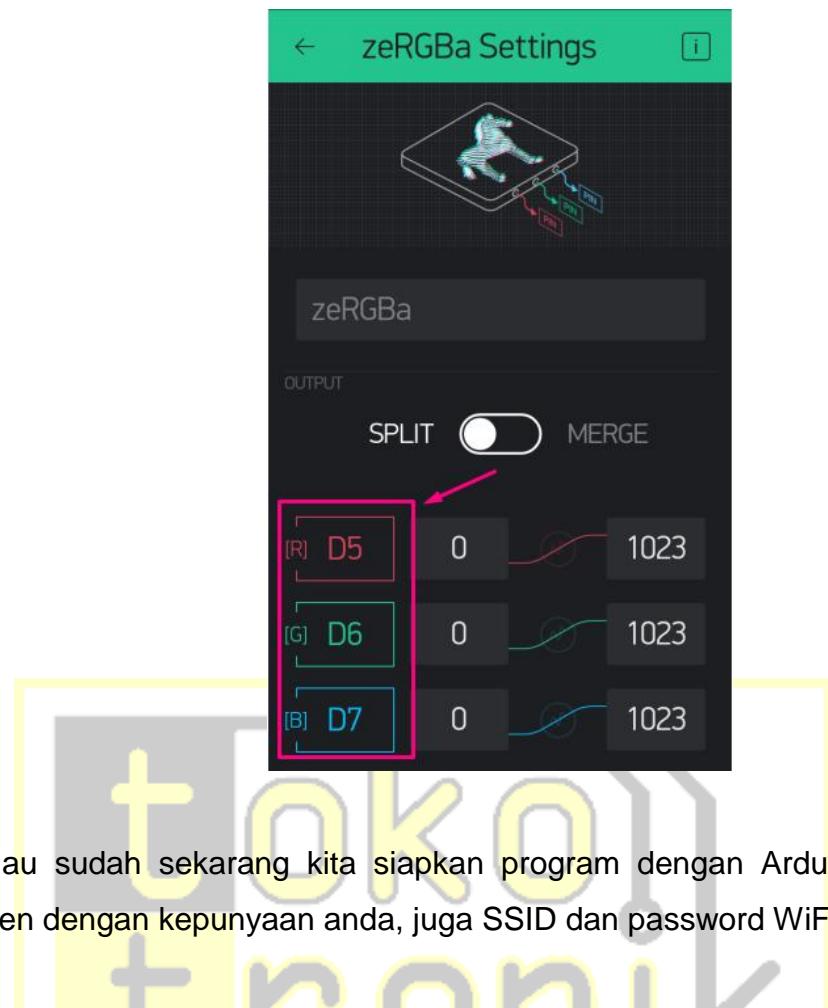
Setting widget zeRGBa. Klik di widget zeRGBa.



Klik “PIN [R]” kemudian pada “Select pin” pilih “Digital” dan “D5” karena pin Red (R) nantinya kita hubungkan dengan D5.

Klik “PIN [G]” kemudian pada “Select pin” pilih “Digital” dan “D6” karena pin Red (G) nantinya kita hubungkan dengan D6

Klik “PIN [B]” kemudian pada “Select pin” pilih “Digital” dan “D7” karena pin Red (B) nantinya kita hubungkan dengan D7



Kalau sudah sekarang kita siapkan program dengan Arduino IDE. Ganti token dengan kepunyaan anda, juga SSID dan password WiFi-nya.

```
*****
* Program :Project 15 Kontrol LED RGB
* Input :
* Output : RGB LED
* IoT Starter Kit Inkubatek
* www.tokotronik.com
*****
```

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
//Ganti dengan Token anda
char auth[] = "ae604d9ca16244aea483564013a119eo";

// Ganti dengan WiFi anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";

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

```
}
```

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

Sesuaikan Token, SSID dan Password WiFi nya.

Save programnya kemudian *Upload*. Tunggu sampai selesai. Jika sudah tersambung dengan WiFi, buka aplikasi Kontrol RGB tadi kemudian jalankan (Run) tombol play di atas kanan.



Geser dengan jari lingkaran putih untuk menentukan warna LED RGB, perhatikan LED RGB di board IoT Starter Kit juga berubah warna menyesuaikan warna yang kita pilih.

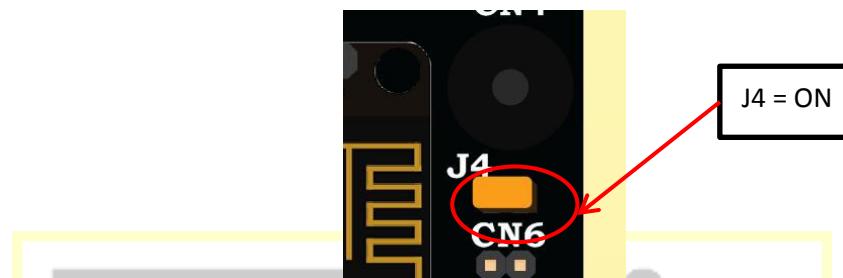
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XXI. Project IoT 16 : Kontrol Buzzer via Android

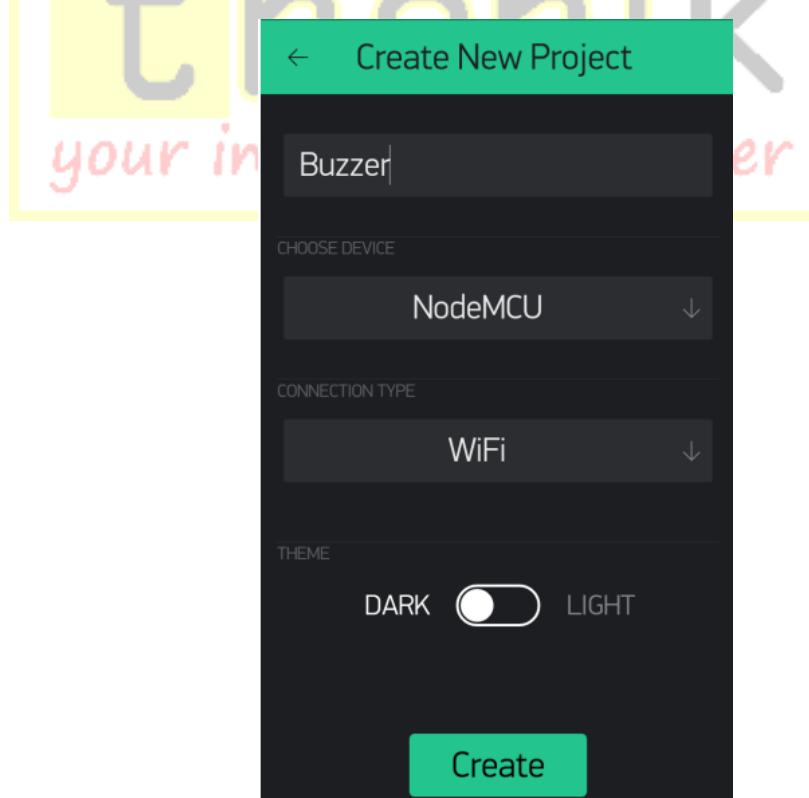
Buzzer biasa dipakai untuk ‘alarm’ dengan bunyi ‘khas’nya. Kita akan kontrol buzzer ini dengan aplikasi Android yang kita develop melalui Blynk..



Rangkaian Buzzer ada di board IoT Starter Kit Inkubatek. Pasang jumper J4 maka Buzzer akan terhubung dengan pin D1 (tanpa kabel). Jika pin D1 diberi logika “HIGH” maka Buzzer akan berbunyi.

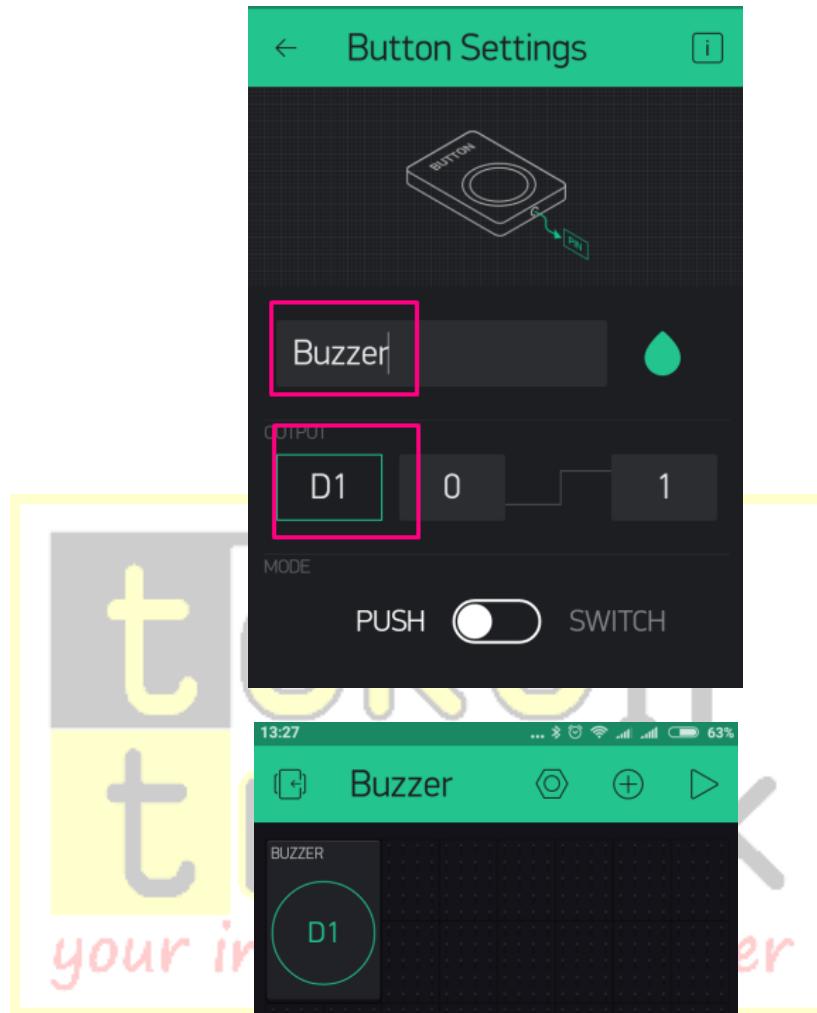


Buat aplikasi di Blynk, Create Project kemudian beri nama “**Buzzer**” dengan “NodeMCU” dan koneksi “WiFi”, klik tombol “Create” untuk mengirim kode Token ke email anda. Catat token tersebut, nanti kita pakai untuk pemrograman.



IoT Starter Kit Inkubatek

Klik pada widget Button untuk seting nilainya. Beri label Buzzer kemudian pada OUTPUT pilih pin **Digital D1** karena Buzzer terhubung dengan pin D1



Sekarang kita buat programnya di Arduino IDE. Ganti auth/token, ssid dan password WiFi.

```
*****
* Program :Project 16 Kontrol Buzzer
* Input : -
* Output : Buzzer di D1
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/

```

```
#define BLYNK_PRINT Serial
```

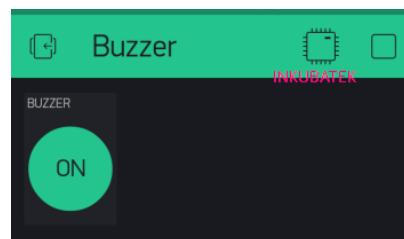
```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
//Ganti dengan Token anda
char auth[] = "6d2c7f338eob498a87c8ec4fbaf07406";

// Ganti dengan WiFi anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";

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

void loop()
{
```

Save programnya kemudian *Upload*. Tunggu sampai selesai. Jika sudah tersambung dengan WiFi, buka aplikasi Buzzer tadi kemudian jalankan (Run) tombol play di atas kanan.



Tekan tombol (tekan tahan) maka Buzzer akan berbunyi.

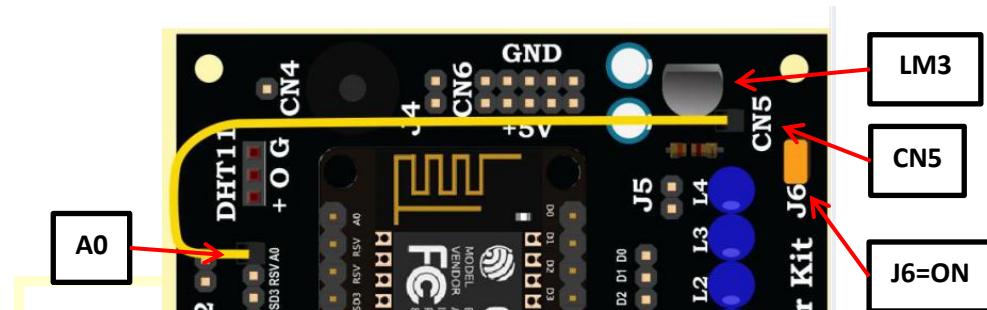
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XXII. Project IoT 17 : Monitoring Suhu LM35 Tampilan Grafik di Android

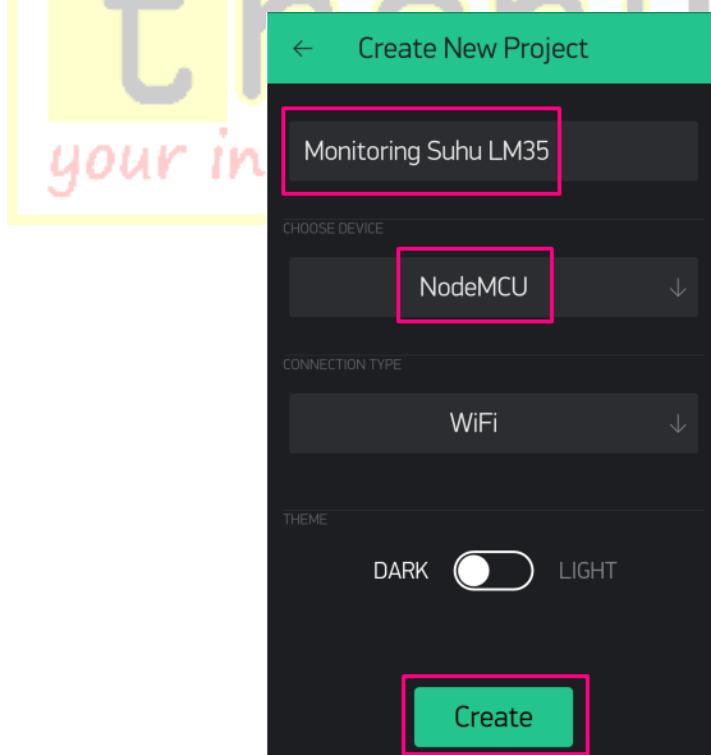
IoT Starter Kit Inkubatek

Tampilan grafik atau chart akan terlihat lebih menarik karena kita dapat melihat pola hasil pembacaan sensornya. Kali ini kita membaca suhu dengan sensor LM35 dan menampilkan hasilnya ke grafik dengan aplikasi Android Blynk. Rangkaian LM35 seperti pada Project IoT 4.

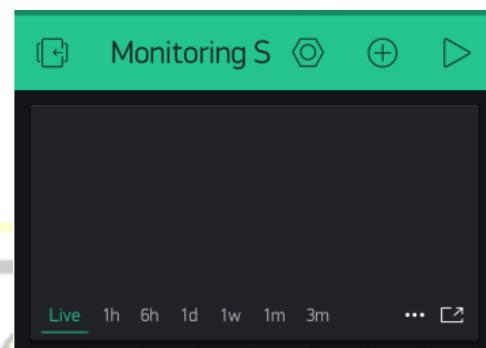
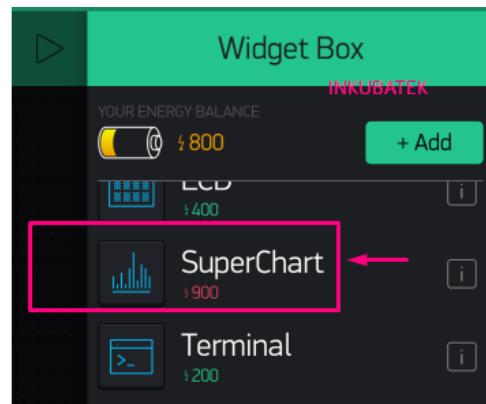
Pasang jumper “J6” agar tegangan LM35 terhubung ke 5V dan hubungkan CN5 ke A0 dari NodeMCU V3.



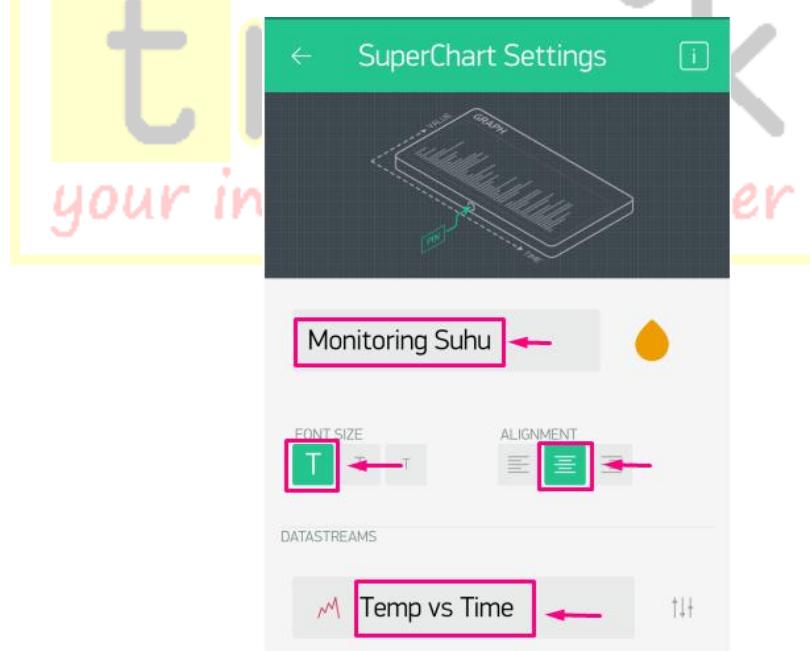
Jalankan Blynk, kemudian buat proyek baru “New Project” beri nama “**Monitoring Suhu LM35**”, DEVICE “NodeMCU” kemudian klik “Create” sehingga token terkirim ke email anda. Catat dan simpan.



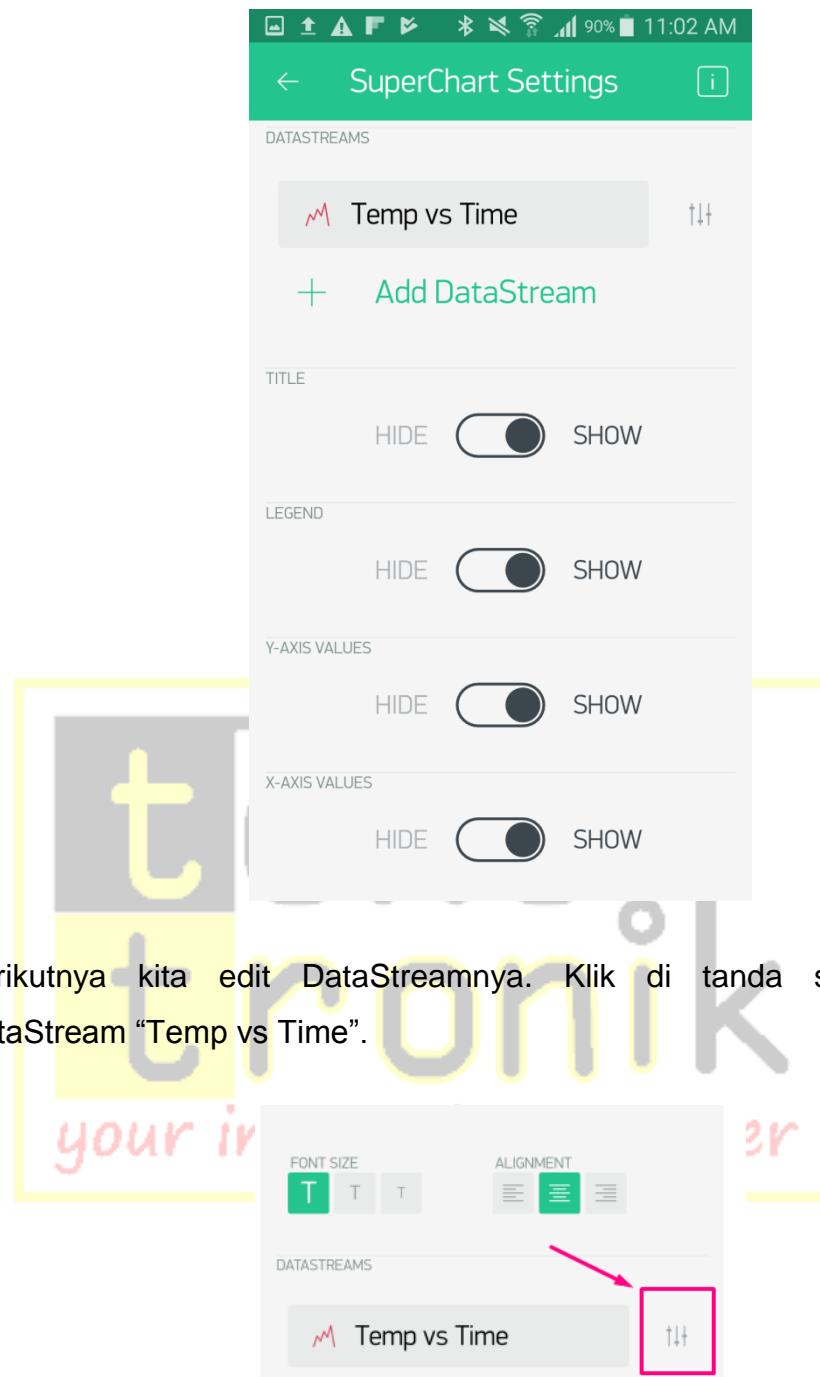
Tambahkan sebuah widget “*SuperChart*” yang ada di Widget Box.



Klik widget *SuperChart* untuk melakukan konfigurasi (setting).



Ganti judulnya misalnya “**Monitoring Suhu**” kemudian set Font Size nya juga Alignment (rata tengah). Datastreams nya ganti “**Temp vs Time**” . Y-AXIS VALUES diset ke “**SHOW**”.



DESIGN pilih yang "Line" , nilai MIN diisi "0" dan MAX diisi "100". Klik bagian INPUT.



Selanjutnya kita buat programnya di Arduino IDE. Jangan lupa untuk mengganti TOKEN (auth), ssid dan password.

```
*****
```

* Program :Project 17 Monitoring Suhu LM35 Tampilan Grafik di Android

* Input : Sensor LM35

```
* Output : Grafik Blynk
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SimpleTimer.h>

SimpleTimer timer;
int adco;
int outputpin= Ao;
float celcius,millivolts;

// Ganti dengan kode token anda
char auth[] = "8053d6d2fe5546cob21822c8b714bef4";
// Ganti dengan WiFi dan Password anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";

void setup()
{
    Serial.begin(115200);
    delay(10);
    Blynk.begin(auth, ssid, pass);
    timer.setInterval(1000, sendData);
}

void sendData()
{
    adco = analogRead(outputpin);
    millivolts = (adco/1024.0) * 3300;
    celcius = millivolts/10;
    Serial.println(celcius);
```

```
Blynk.virtualWrite(o, celcius);//virtual input V10 di Blynk (suhu)
```

```
}
```

```
void loop()
```

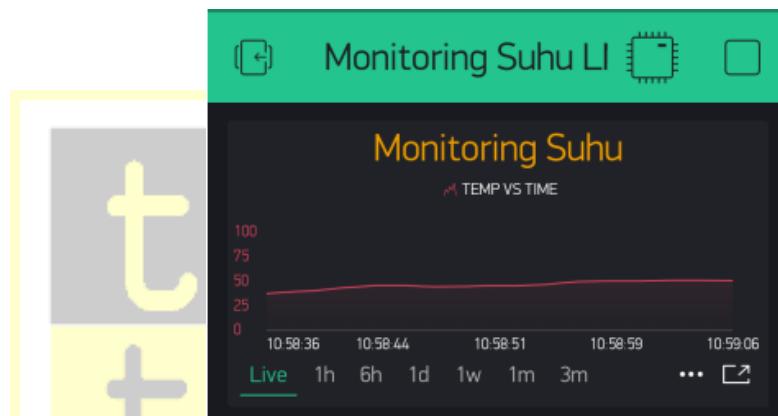
```
{
```

```
    Blynk.run();
```

```
    timer.run();
```

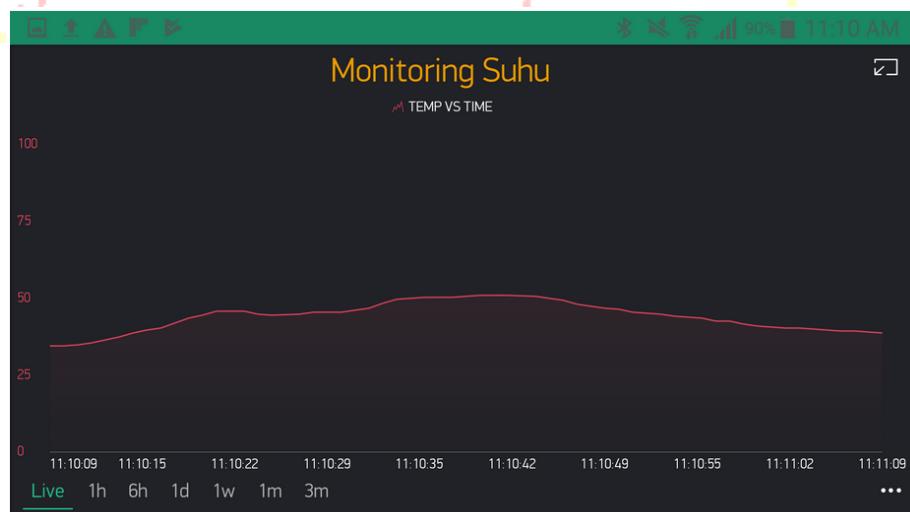
```
}
```

Save kemudian *Upload*, tunggu sampai selesai. Klik tombol “Run” di aplikasi monitoring LM35 yang tadi dibuat di Blynk.



Untuk tampilan landscape, klik tanda panah di pojok kanan bawah :

your innovation partner

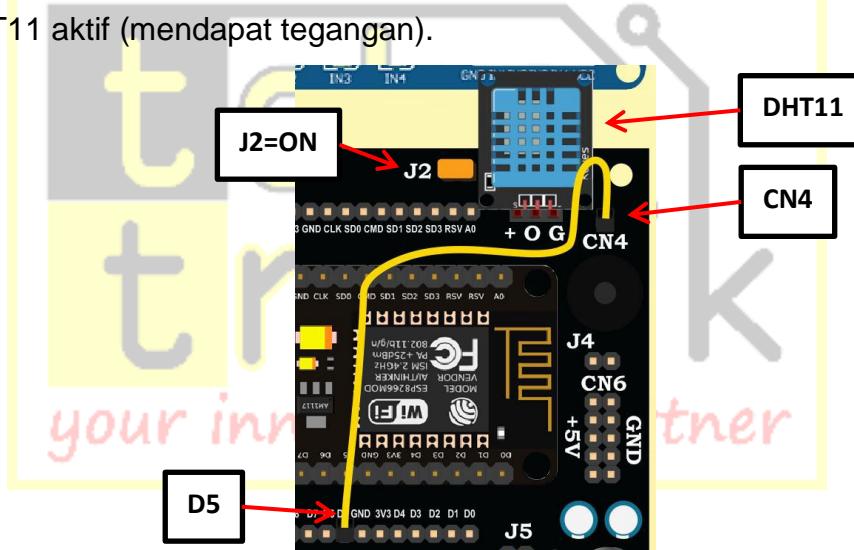


[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

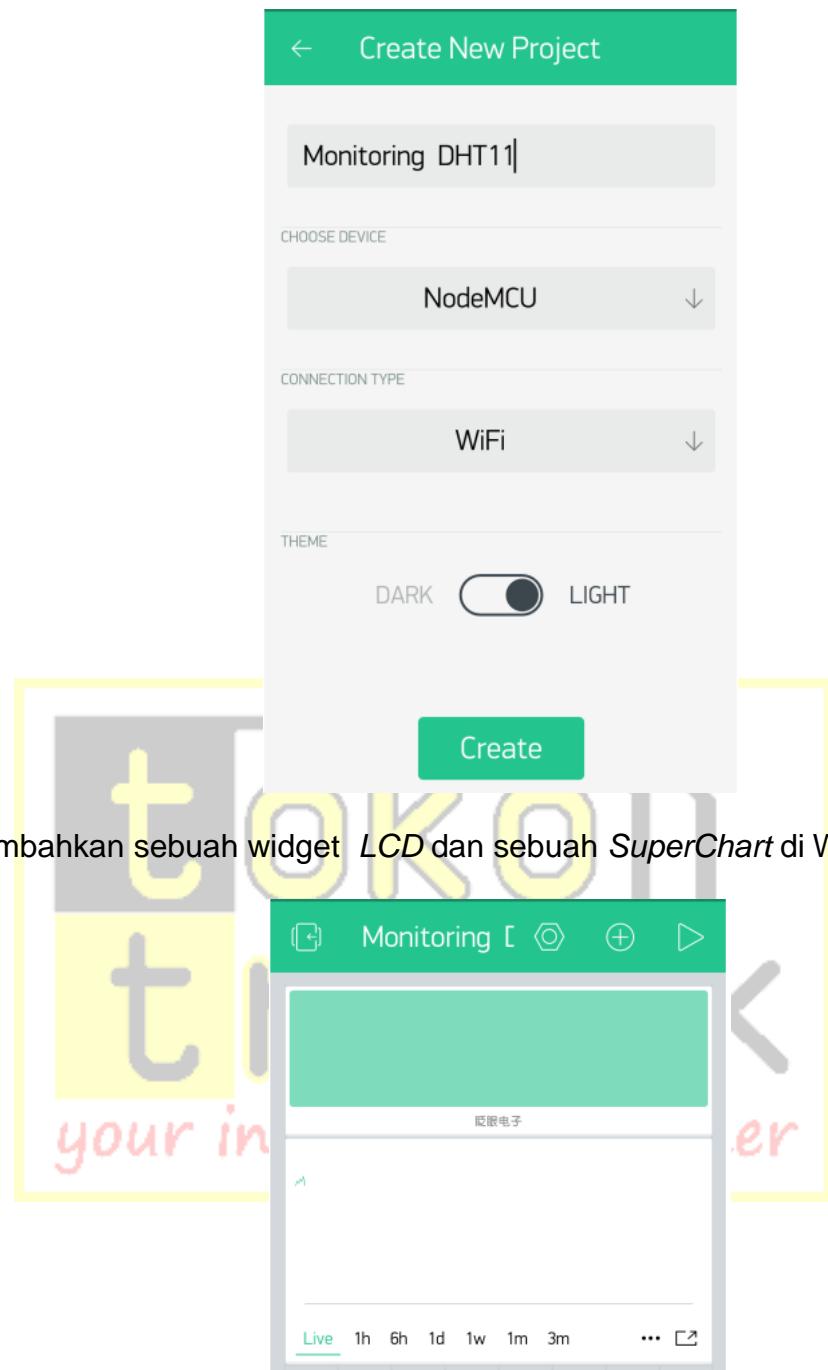
XXIII. Project IoT 18 : Monitoring Suhu kelembaban Tampilan Chart Android

Kita kembangkan proyek 17 dengan mengganti sensornya memakai DHT11. Hasil pembacaan sensor DHT11 berupa suhu dan kelembaban kita tampilkan dalam *SuperChart*.

Pasang sensor DHT11 ke konektor female 3 pin yang ada di board IoT Starter Kit Inkubatek, **jangan sampai terbalik ya**. Pasang jumper J2 agar DHT11 aktif (mendapat tegangan).



Koneksikan CN4 (output sensor DHT11) dengan pin D5. Sekarang kita buat Jalankan aplikasi *Blynk* kemudian “New Project”. Beri nama proyek “Monitoring DHT11” pilih NodeMCU kemudian klik “Create” sehingga Token akan terkirim ke email anda. Buka emailnya, catat dan simpan.

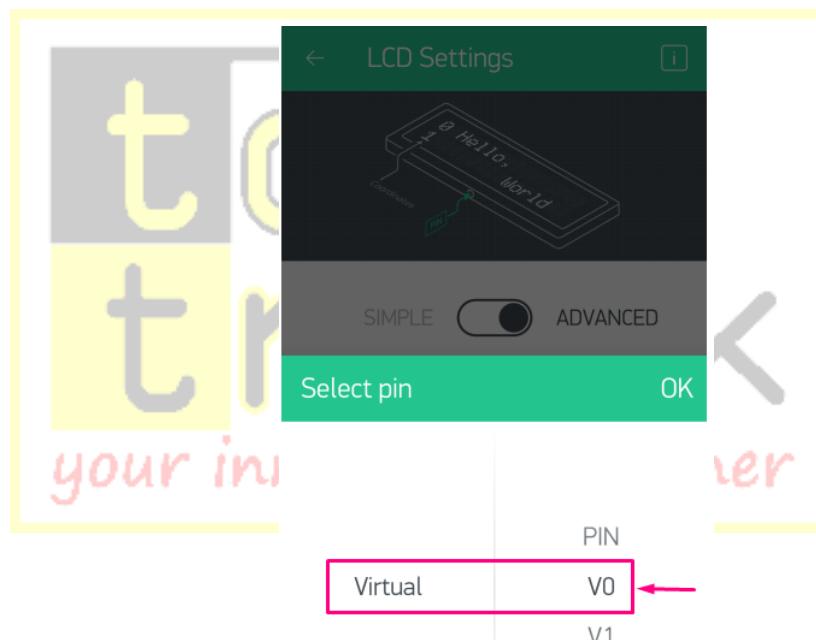


Tambahkan sebuah widget LCD dan sebuah SuperChart di Widget Box.

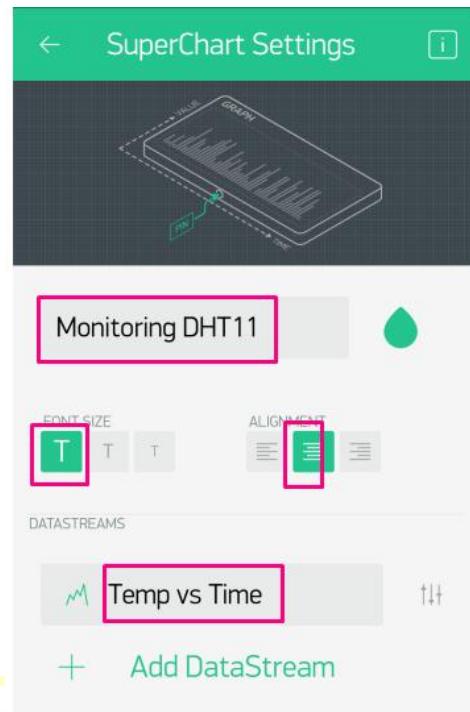
Setting widget LCD, klik di bagian LCD.



Set ke “ADVANCED” kemudian klik INPUT “PIN”.

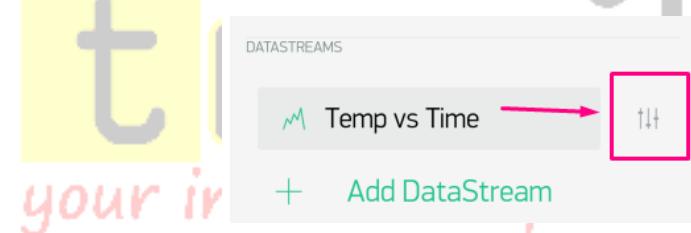


Pilih **Virtual V0** kemudian klik “OK”. Klik *back* (anak panah kiri pojok atas kiri) untuk kembali ke tampilan utama. Sekarang kita seting SuperChart, klik widget SuperChart.



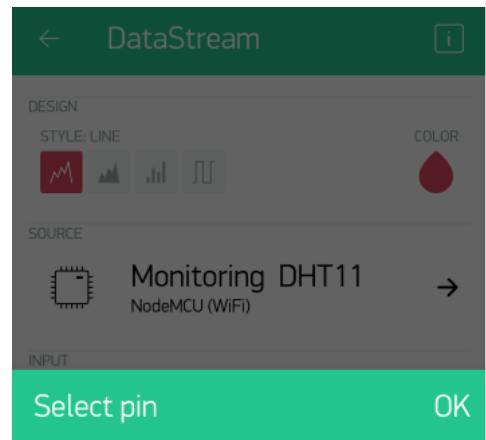
Beri label/nama “Monitoring DHT11”, FONT SIZE pilih yang besar kemudian ALIGNMENT pilih rata tengah. DataStream diganti tulisan “Temp vs Time”.

Klik toolbar disebelah kanan “Temp vs Time”



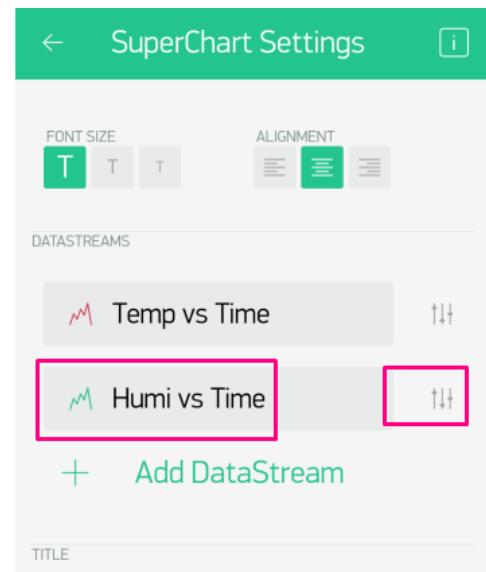
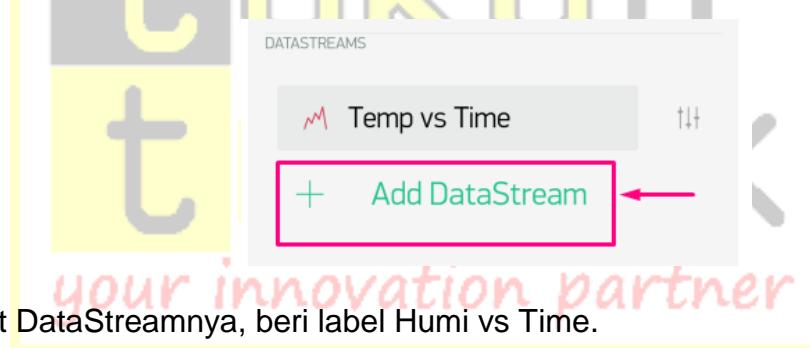
Pada INPUT pilih “Virtual V1” kemudian klik “OK ”. Klik tombol kembali.

IoT Starter Kit Inkubatek



Tambahkan sebuah “DataStream” untuk grafik kelembabannya. Klik tombol “Add DataStream” .

Edit DataStreamnya, beri label Humi vs Time.



Klik toolbar disebelah kanan DataStream tadi untuk seting INPUT



Klik “INPUT” kemudian pilih “Virtual V2” trus klik “OK”. Klik ‘back’ hingga ke tampilan utama.

Sekarang kita buat programnya di Arduino IDE. Jangan lupa untuk mengganti kode Token, SSID dan Password WiFi.

```
*****
* Program :Project 18 Monitoring Suhu LM35 Tampilan Grafik di Android
* Input : Sensor DHT11
* Output : Grafik Blynk
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>
#include <SimpleTimer.h>
#define DHTPIN D5
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

SimpleTimer timer;
float humi, temp;
// Ganti dengan kode token anda
char auth[] = "429bb07491c24282a4d50d2b33f64ff0";
```

```
// Ganti dengan WiFi dan Password anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";
char outstr[15];

WidgetLCD lcd(V0);

void setup()
{
    Serial.begin(115200);
    delay(10);
    Blynk.begin(auth, ssid, pass);
    lcd.clear();
    lcd.print(0, 0, "Monitoring DHT11");
    lcd.print(0, 1, "Inkubatek Jogja");
    delay(2000);
    lcd.clear();
    lcd.print(0, 0, "Temp=   C");
    lcd.print(0, 1, "Humi=   %RH");
    timer.setInterval(1000, sendData);
}

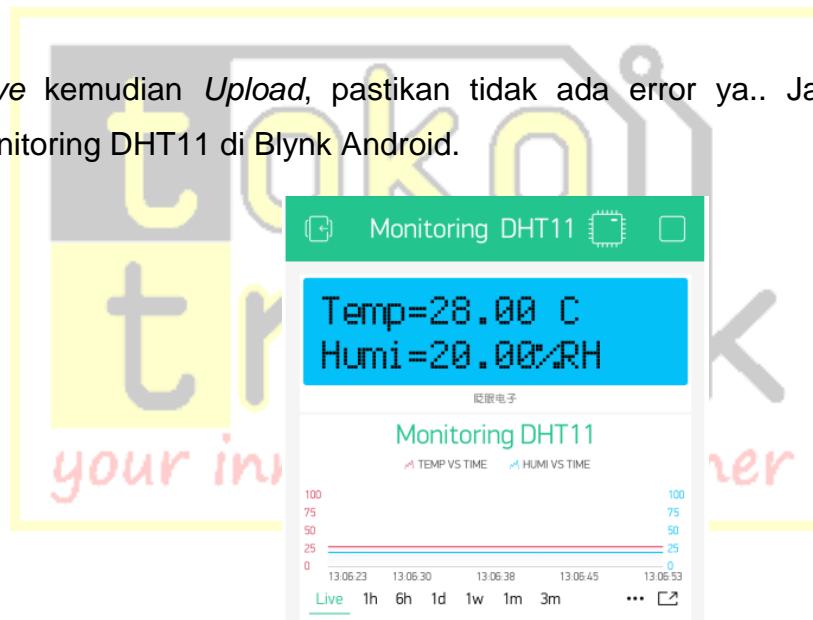
void sendData()
{
    humi = dht.readHumidity();
    temp = dht.readTemperature();
    if (isnan(humi) || isnan(temp)) {
        Serial.println("DHT11 tidak terbaca... !");
        return;
    }
    else{
        Serial.println(temp);
        Serial.println(humi);
        Blynk.virtualWrite(1, temp); //virtual input V10 di Blynk (suhu)
        Blynk.virtualWrite(2, humi); //virtual input V11 di Blynk (kelembaban)
```

```
dtostrf(temp,4, 2, outstr);
LCD.print(5, 0, outstr);
dtostrf(humi,4, 2, outstr);
LCD.print(5, 1, outstr);
}

}

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

Save kemudian *Upload*, pastikan tidak ada error ya.. Jalankan aplikasi Monitoring DHT11 di Blynk Android.



Hasil pembacaan sensor akan ditampilkan dalam LCD dan Grafik

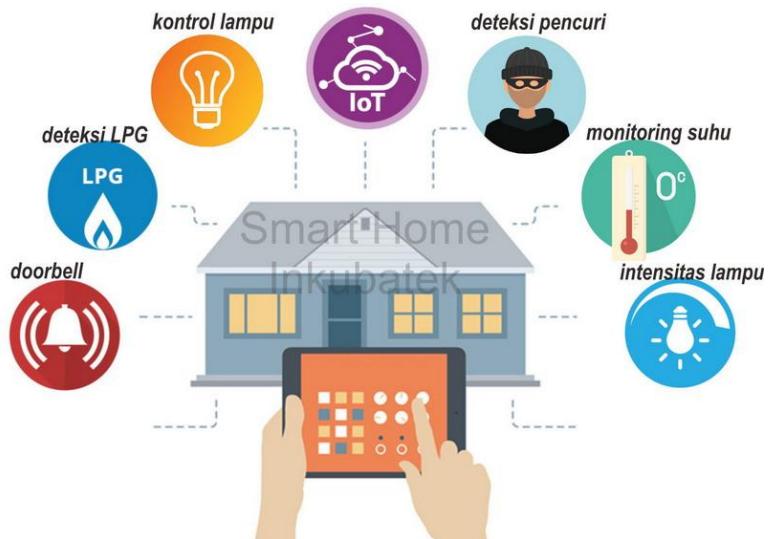
[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

XXIV.Project IoT 19 : Smart Home dengan Android

Pada proyek kali ini kita akan melibatkan beberapa device untuk membuat sebuah system *Smart Home*. Beberapa peralatan rumah tangga dan

IoT Starter Kit Inkubatek

keadaan lingkungan rumah akan kita kontrol dan monitor melalui HP Android.



Hardware IoT Starter Kit yang dipakai :

Device	Fungsi (analogi)
Sensor suhu LM35	Temperatur ruangan rumah
Relay 1	Kontrol ON – OFF lampu luar
Relay 2	Kontrol ON – OFF lampu ruang dalam
LED 4	Intensitas lampu kamar tidur
Sensor gas MQ2	Deteksi kebocoran gas
Buzzer	Bel pintu (doorbell)
Sensor PIR	Deteksi orang/pencuri

Buat rangkaian sensor dll di IoT Starter Kit Inkubatek:

- Sensor PIR :

Sensor PIR	IoT Starter Kit
VCC	3V
OUT	D3
GND	GND

IoT Starter Kit Inkubatek

- Sensor MQ2 :

Sensor MQ2	IoT Starter Kit
A0	-
D0	D5
GND	GND
VCC	+5V

- Sensor LM35 :

- Jumper J6 : terpasang (ON)
- CN5 hubungkan ke A0

- Modul Relay :

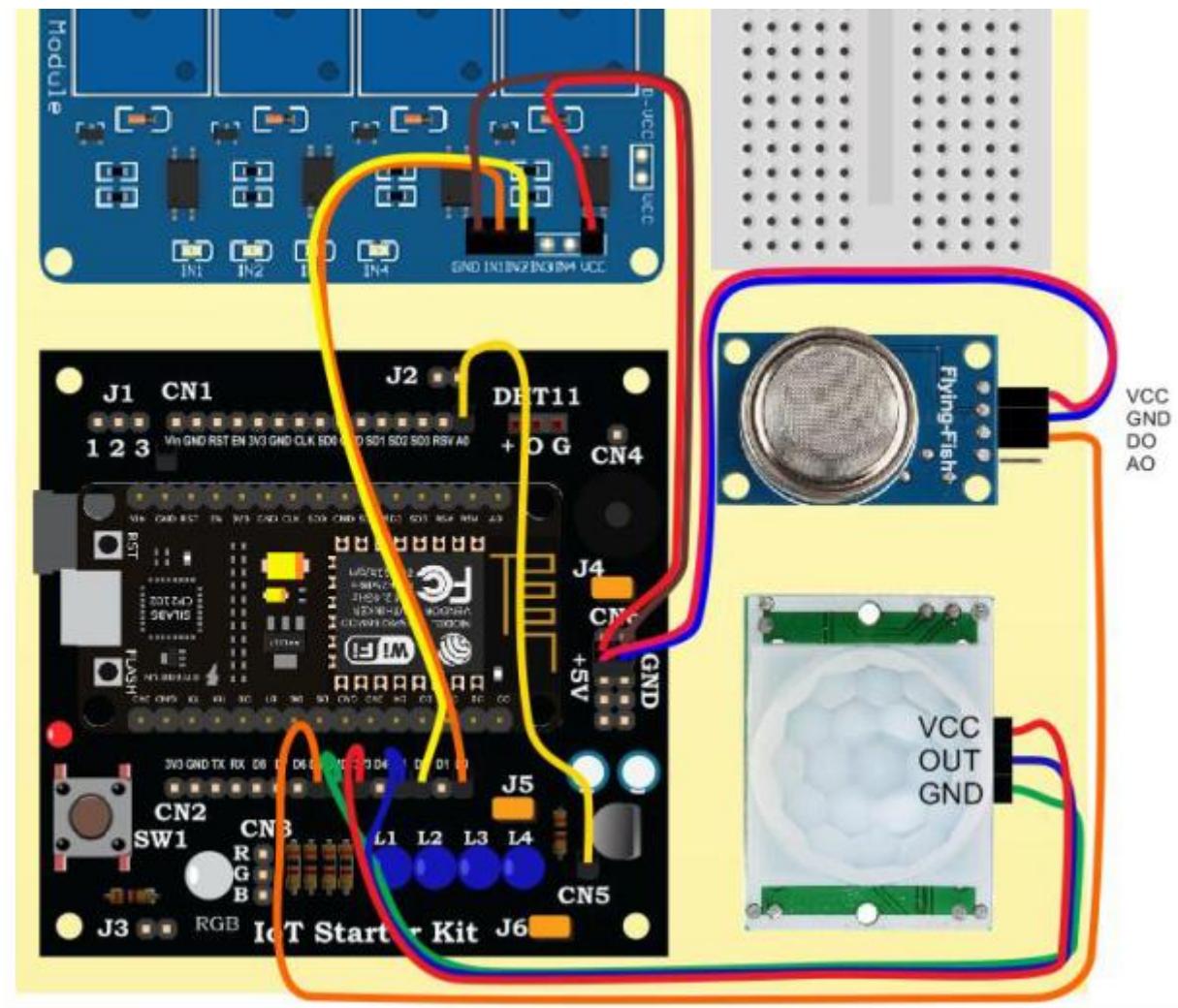
Relay modul	IoT Starter Kit
1	D0
2	D2
3	-
4	-
VCC	+5V
GND	GND

- Buzzer :

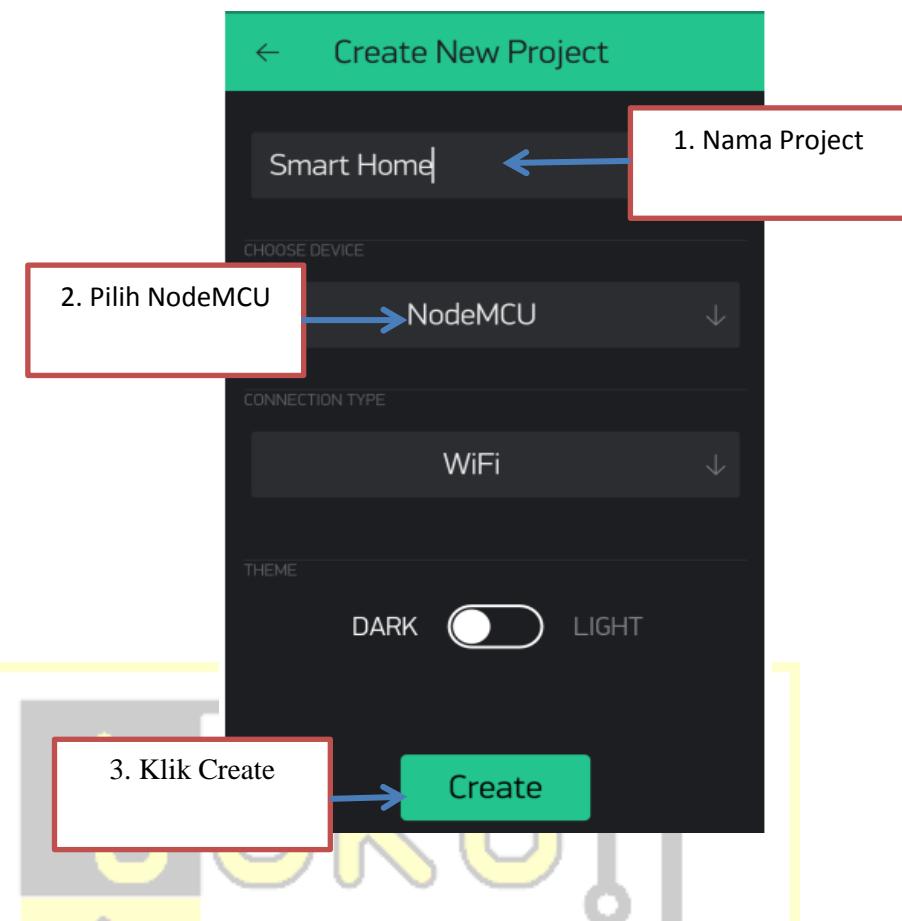
- Jumper J4 : terpasang (ON)
- Otomatis terhubung ke D1 (tanpa kabel)

- LED 4 :

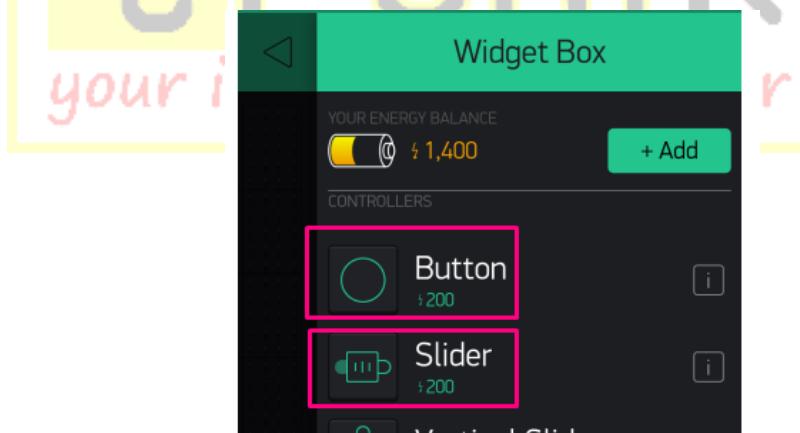
- Jumper J5 : terpasang (ON)
- Otomatis terhubung ke D8 (tanpa kabel)



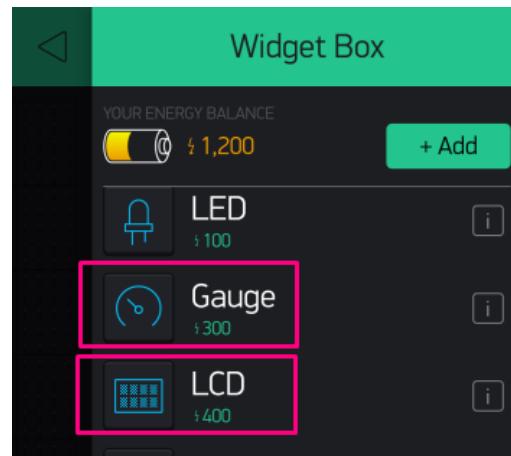
Hardware sudah beres kita lanjutkan ke aplikasi di Android. Jalankan Blynk kemudian buat proyek baru “ + New Project” (atau dapat juga mengedit proyek sebelumnya). Beri nama proyek “**Smart Home**” kemudian pada CHOOSE DEVICE pilih NodeMCU.



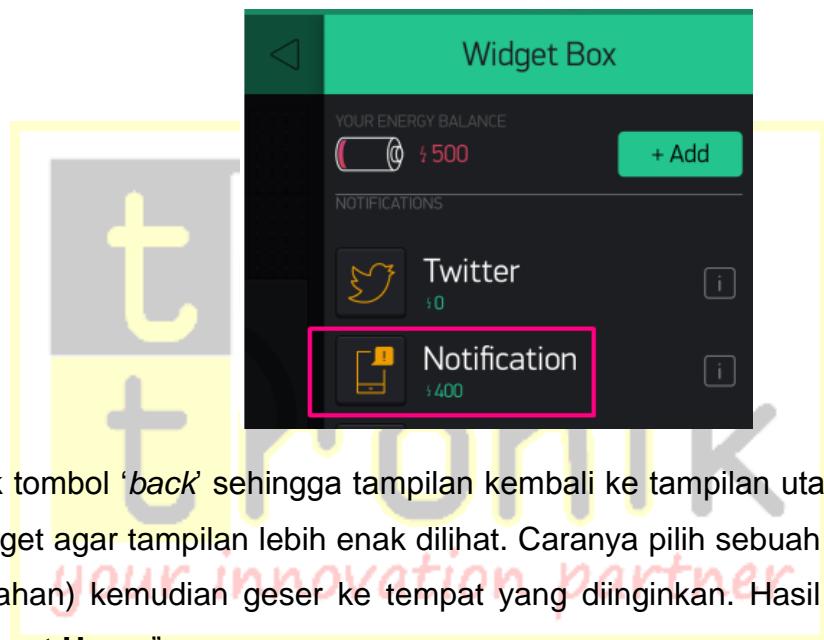
Klik Create sehingga Token terkirim ke email anda. Buka dan catat kode token nya. Tambahkan widget Button (3) dan sebuah Slider.



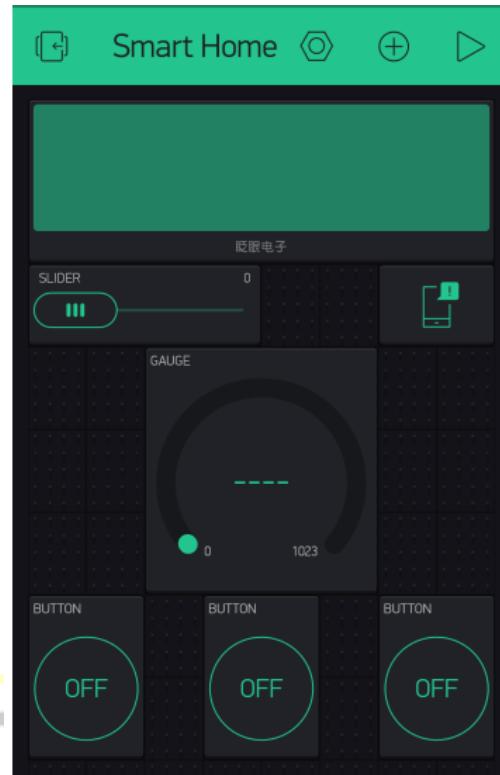
Tambahkan juga sebuah Gauge dan LCD



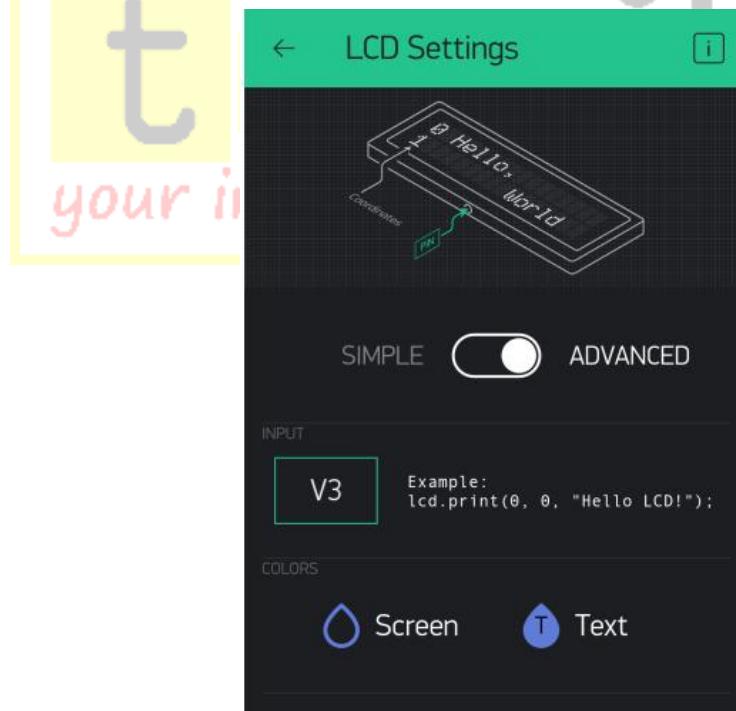
Terakhir tambahkan sebuah notifikasi (Notification)



Klik tombol 'back' sehingga tampilan kembali ke tampilan utama. Atur lokasi widget agar tampilan lebih enak dilihat. Caranya pilih sebuah widget (sentuh + tahan) kemudian geser ke tempat yang diinginkan. Hasil akhir tampilan **"Smart Home"** :



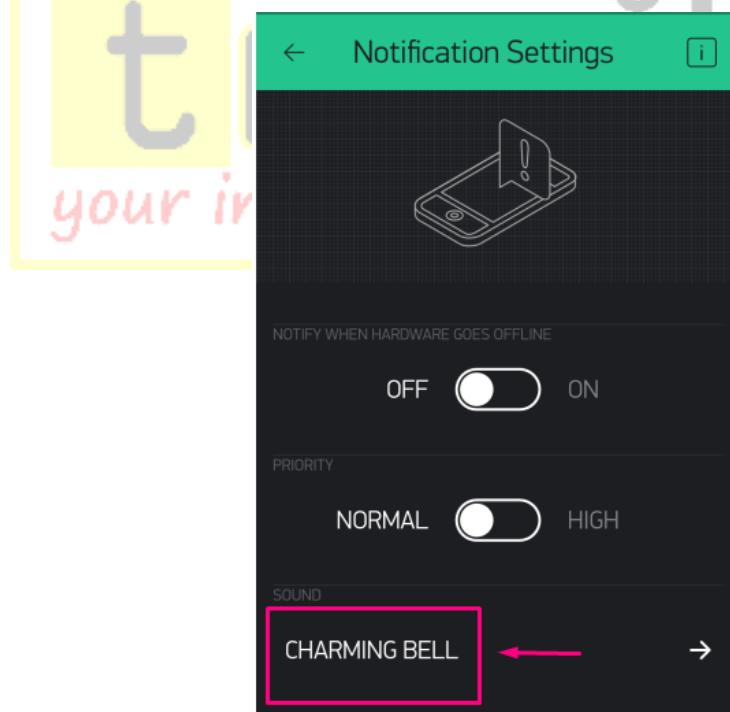
Saatnya kita setting masing – masing widget. Mulai dari LCD. Set ke "ADVANCED" kemudian INPUT pilih **Virtual V3**.



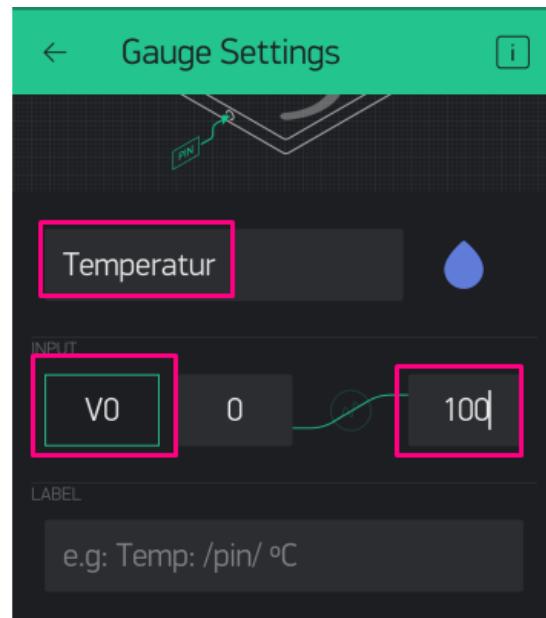
Lanjut ke widget Slider. Klik widget slider kemudian beri label/nama "Lampu Tidur". OUTPUT pilih Digital D8.



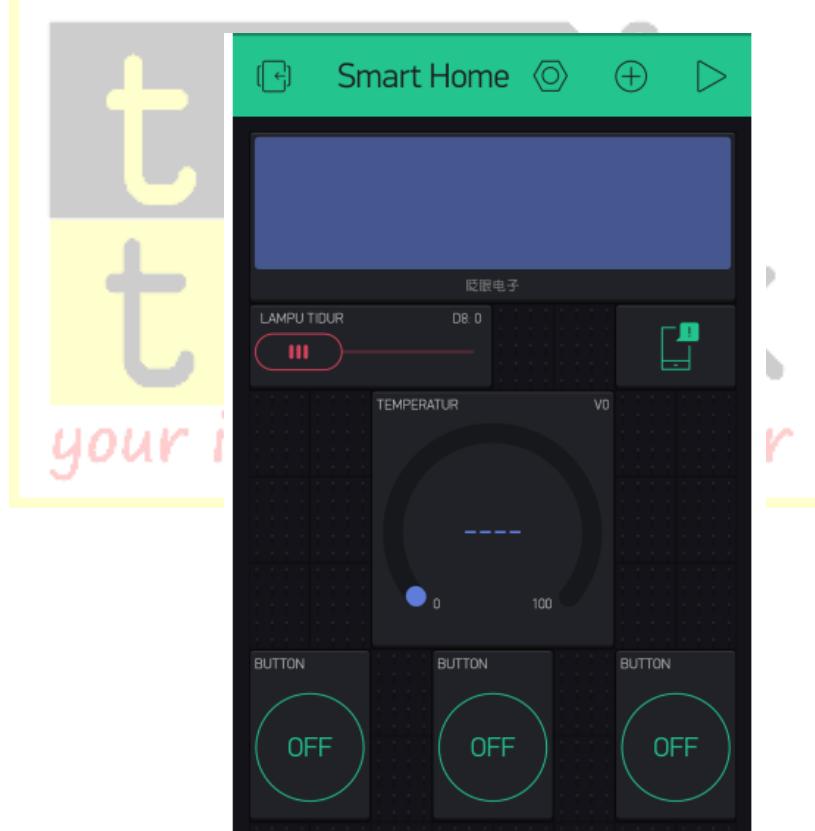
Lanjut ke widget Notification. Berfungsi untuk notifikasi ketika ada orang masuk rumah (sensor PIR) atau ada kebocoran gas (sensor MQ2). Kalau suara alarm (nada) tidak perlu diganti (default) maka widget Notification tidak perlu diseting. Pilih SOUND untuk mengganti nada-nya.



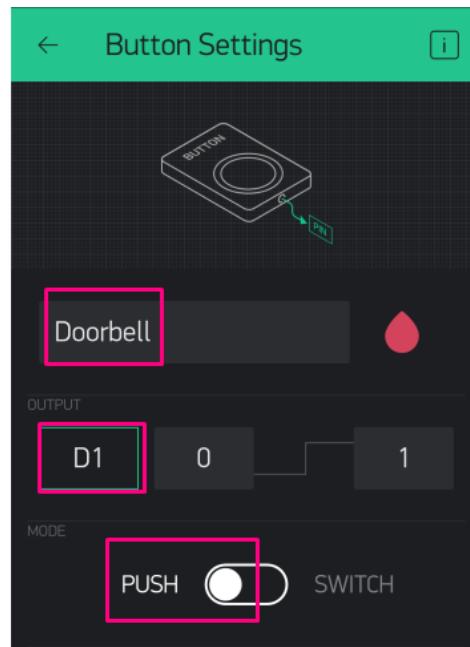
Lanjut widget Gauge. Beri label “Temperature” kemudian INPUT pilih **Virtual V0** dengan nilai maksimal 100.



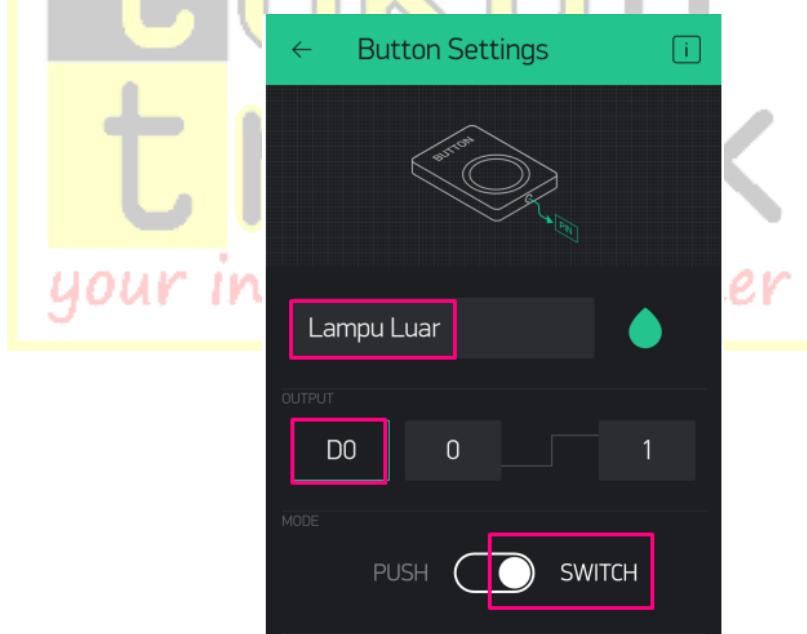
Kita lihat dulu sebentar hasilnya ..



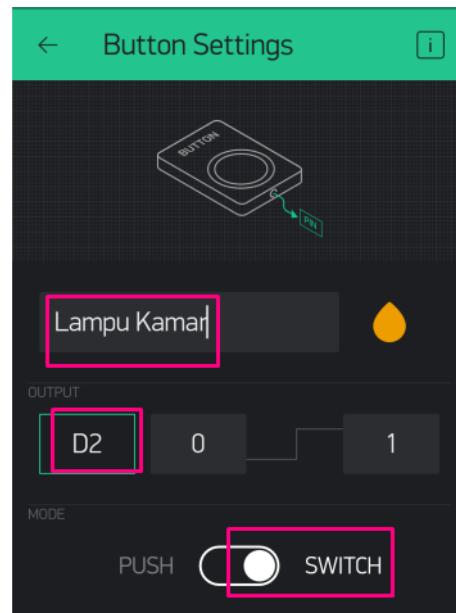
Masih ada 3 BUTTON yang belum diseting. Klik BUTTON 1 (paling kiri). Fungsinya untuk ‘bell pintu’, riilnya terhubung ke Buzzer di pin D1. Beri label/nama “Doorbell” kemudian OUTPUT pilih Digital D1.



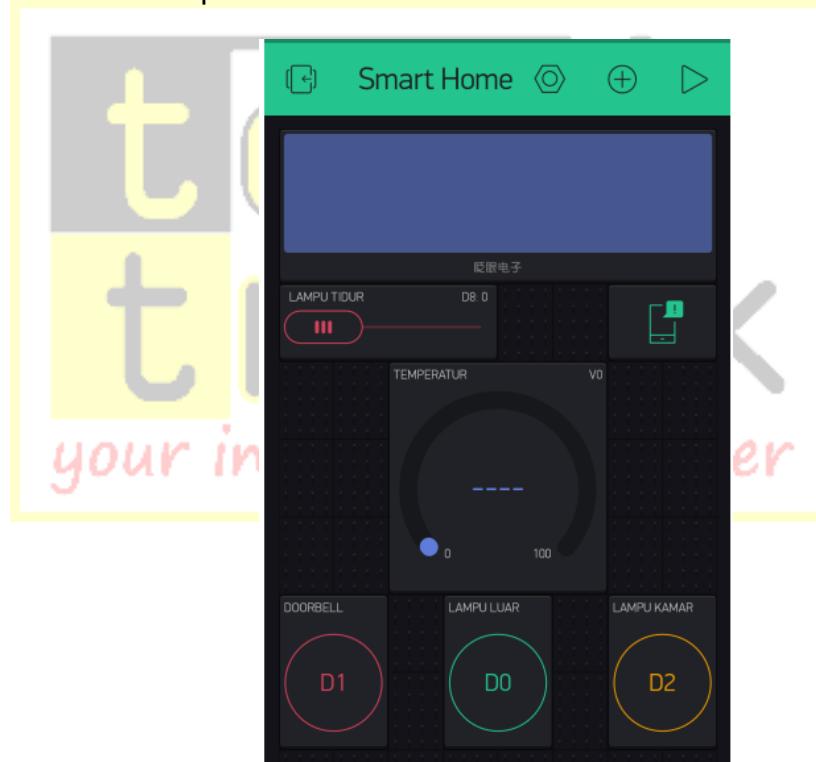
BUTTON ke-2 (tengah) berfungsi untuk control lampu luar. Beri label “Lampu Luar” dengan OUTPUT Digital D0 kemudian MODE seting ke “SWITCH”



BUTTON ke-3 berfungsi untuk control lampu dalam ruangan. Beri label “Lampu Kamar” dengan OUTPUT Digital D2 kemudian MODE seting ke “SWITCH”



Hasil terakhir tampilan Smart Home :



Alhamdulillahsekarang kita buat program di Arduino IDE. Buka lembar baru (*File → New*) kemudian ketik program berikut. **Jangan lupa ganti Token, SSID & Password WiFi.**

```
*****
```

```
* Program :Project 19 Smart Home
```

```
* Input : LM35, PIR, MQ2
* Output : Buzzer, Relay, LED
* IoT Starter Kit Inkubatek
* www.tokotronik.com
* *****/
```

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SPI.h>
#include <SimpleTimer.h>

//Ganti dengan Token anda
char auth[] = "8499boageoe94101b9e1e5d433445512";
// Ganti dengan WiFi anda
char ssid[] = "FirnasAbe";
char pass[] = "podowingi";
WidgetLCD lcd(V3);
int adco;
float millivolts;
float celsius;

SimpleTimer timer;

int sensor_in;
#define pirPin D3 // sensor PIR
int pirValue;
#define MQ2Pin D5 // sensor MQ2
int MQ2Value;
void setup()
{
    Serial.begin(115200);
    delay(50);
    timer.setInterval(1000, sendData);
```

```
Blynk.begin(auth, ssid, pass);
lcd.clear();
lcd.print(0, 0, "== Smart Home ==");
lcd.print(0, 1, "Inkubatek Jogja");
delay(1000);
pinMode(pirPin, INPUT);
pinMode(MQ2Pin, INPUT);
}

void sendData()
{
adco= analogRead(Ao);
millivolts= (adco/1024.0) * 3300;
celsius= millivolts/10;
Blynk.virtualWrite(0, celsius);
}

void getPirValue(void)
{
pirValue = digitalRead(pirPin);
if (pirValue)
{
Serial.println("==> Motion detected");
lcd.clear();
lcd.print(0, 0, " WASPADA !!! ");
lcd.print(0, 1, "Objek Terdeteksi");
Blynk.notify("Object terdeteksi.....");
while(digitalRead(pirPin));
delay(1000);
}
}

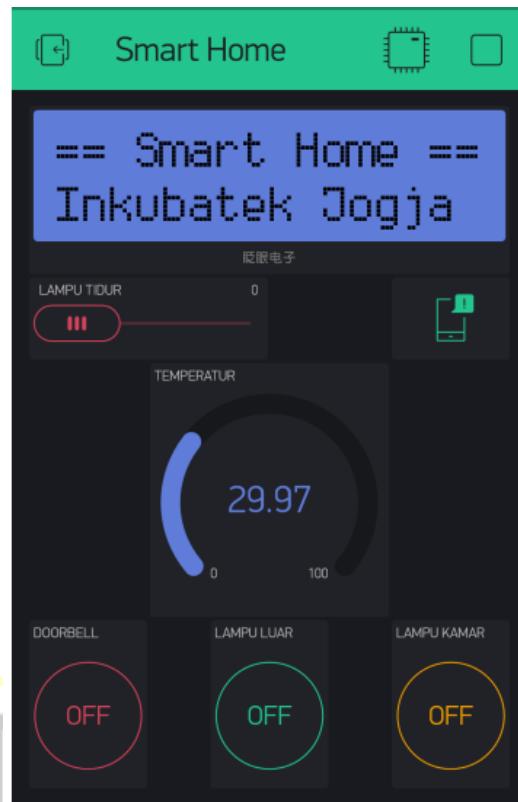
void getMQ2Value(void)
{
```

```
MQ2Value = digitalRead(MQ2Pin);
if (!MQ2Value)
{
    Serial.println("==>Gas terdeteksi");
    lcd.clear();
    lcd.print(0, 0, " WASPADA !!! ");
    lcd.print(0, 1, "Gas Bocor....");
    Blynk.notify("Kebocoran Gas terdeteksi.....!");
    delay(1000);
}

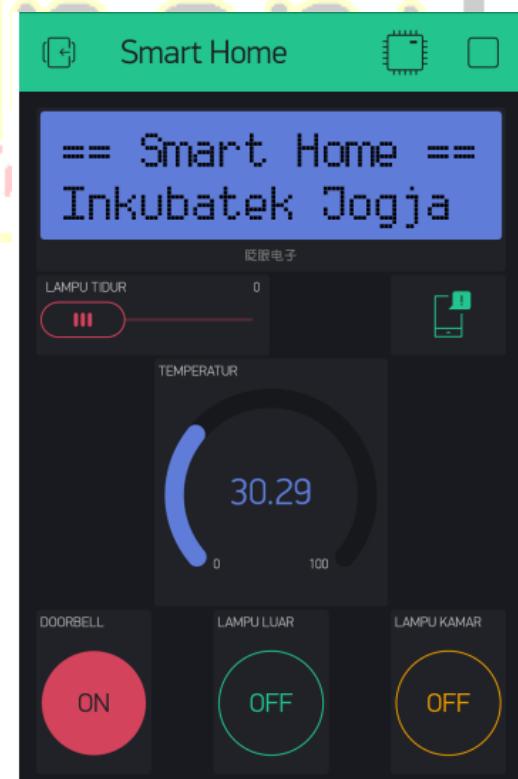
void loop()
{
    Blynk.run();
    timer.run();
    getPirValue();
    getMQ2Value();
}
```

Jangan lupa untuk mengganti token, SSID dan Password WiFi anda.
Save kemudian *Upload*, pastikan tidak ada error.

Jalankan aplikasi Smart Home anda.



Coba tes tekan tombol Doorbell, jika berjalan lancar maka buzzer akan berbunyi.



Coba juga sensor PIR (agak sensitive) serta peripheral yang lain.

[jika mengalami kendala dalam pembuatan aplikasi dg Blynk silahkan baca : TUTORIAL MEMBUAT APLIKASI IoT DI ANDROID DG BLYNK.PDF yang ada di CD]

Alhamdulillah selesai juga kita belajar membuat aplikasi IoT dengan **IoT Starter Kit** Inkubatek. Pengembangan pada aplikasi yang lain masih banyak. Teman – teman dapat mencari referensi di internet.

Trouble shooting dan lain sebagainya silakan cek juga di CD.

Terimakasih.

Inkubatek Jogja @2017

