

KOMUNIKASI DATA SISTEM UPLINK DAN DOWNLINK BLYNK

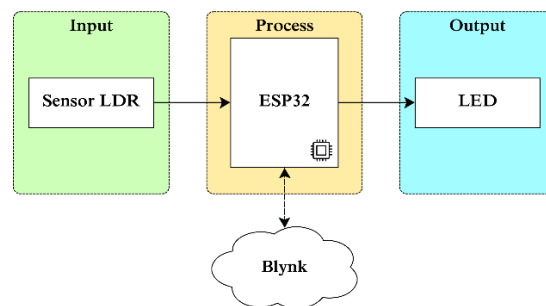
A. Tujuan

1. Memahami konsep integrasi *uplink* dan *downlink* dalam komunikasi dua arah pada sistem IoT berbasis Blynk dan ESP32.
2. Menerapkan kombinasi *uplink* dan *downlink* untuk membangun sistem kendali dan pemantauan berbasis *cloud*.
3. Menggunakan sensor, aktuator dan fungsi pemrograman dalam satu sistem terpadu.
4. Menganalisis kinerja sistem IoT dua arah terhadap perintah kendali serta keakuratan data sensor yang dikirim ke server Blynk.

B. Teori Dasar

Dalam penerapan *Internet of Things* (IoT), kemampuan perangkat untuk berkomunikasi dua arah menjadi penting. Kombinasi antara sistem *uplink* dan *downlink* memungkinkan perangkat tidak hanya mengirim data ke server (*uplink*), tetapi juga menerima perintah dari server atau pengguna (*downlink*).

Pada sistem berbasis ESP32 dan Blynk, *uplink* digunakan untuk mengirimkan data dari perangkat ke aplikasi, misalnya data nilai sensor. Sedangkan *downlink* digunakan untuk menerima perintah dari aplikasi ke perangkat, seperti mengaktifkan LED atau pompa seperti pada Gambar 1.



Gambar 1. Diagram Blok Sistem

Melalui kombinasi *uplink* dan *downlink* sistem dapat melakukan:

- Pemantauan data secara *real-time* melalui antarmuka Blynk.
- Kontrol perangkat secara langsung berdasarkan perintah dari pengguna melalui Blynk.
- Otomatisasi sistem melalui sensor dan aktuator, misalnya menyalakan kipas jika suhu melebihi ambang tertentu.

C. Alat & Bahan

- | | |
|---------------|---|
| a. ESP32 | e. Breadboard |
| b. LED | f. Kabel jumper |
| c. Relay | g. Laptop terinstal Arduino IDE dan library |
| d. Sensor LDR | h. Kabel USB Type C |

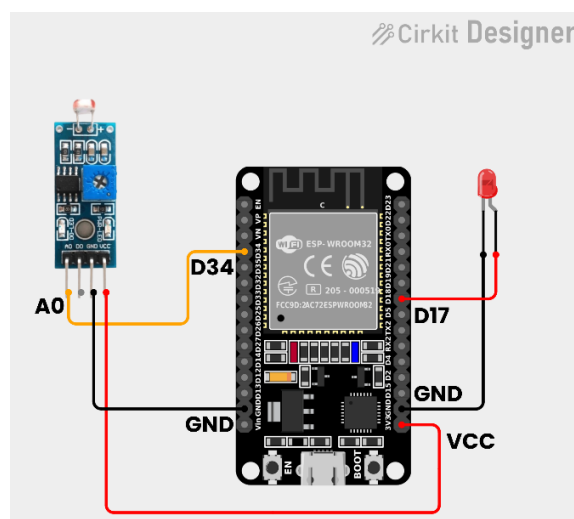
D. Keselamatan Kerja

1. Lakukan, atau laksanakan praktikum di tempat yang bersih dan kering.
2. Menggunakan laptop, atau komputer dengan baik, dan benar.
3. Jangan sentuh langsung bagian kit praktikum yang dialiri listrik.
4. Pastikan *power supply* kit praktikum dalam keadaan mati ketika proses instalasi, atau perakitan kit praktikum.
5. Selalu perhatikan *port* kit praktikum. Jangan sampai tertukar, terbalik, atau salah penempatan dalam instalasi, atau perakitan pada kit praktikum.
6. Gunakan komponen pada kit praktikum sebagaimana mestinya.
7. Jika terjadi *short circuit*, segera matikan *power supply* pada kit praktikum.
8. Mengikuti langkah-langkah praktikum seperti yang tertera pada langkah kerja serta memperhatikan keselamatan

E. Langkah Kerja

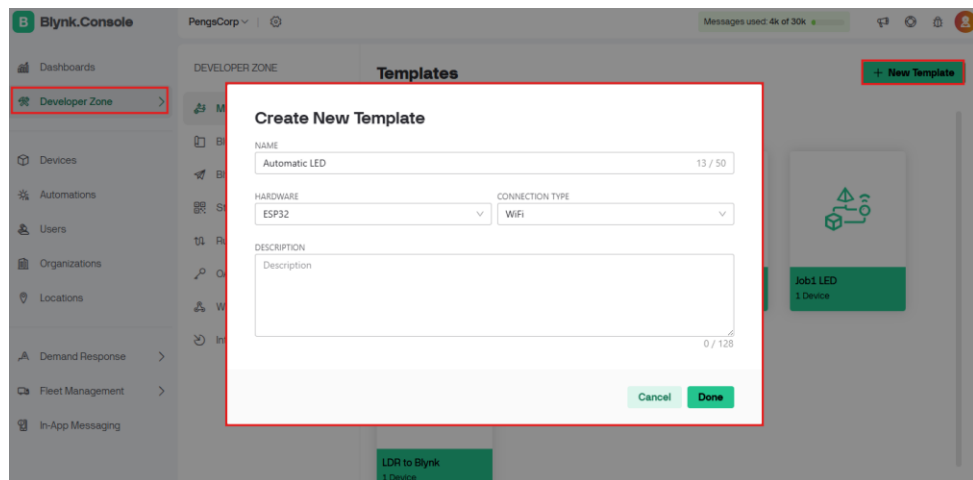
8.1 Sistem Lampu Otomatis dengan Sensor LDR

- a. Buat rangkaian seperti gambar berikut.

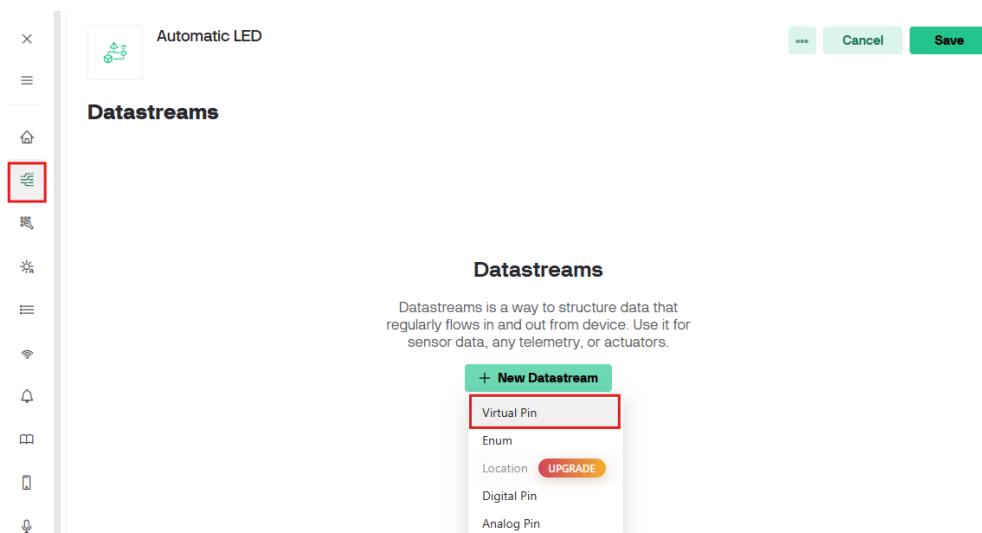


- b. Konfigurasi Blynk terlebih dahulu untuk membuat *virtual* platform.

- c. Masuk ke menu **Developer Zone** lalu pilih **New Template**, isi sesuai dengan kebutuhan lalu klik **Done**.



- d. Selanjutnya, masuk ke menu **Datastreams** > klik **New Datastream** > pilih **Virtual Pin**.

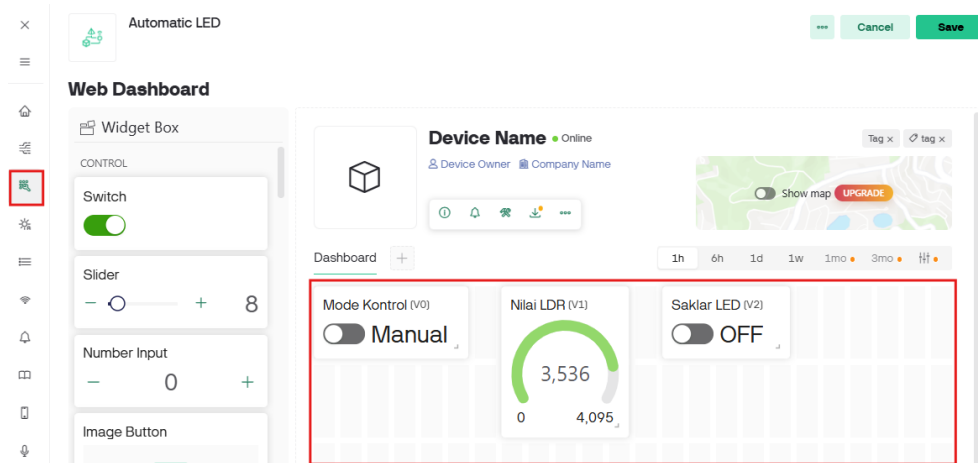


- e. Buat Virtual Pin sesuai dengan kebutuhan, untuk sistem ini digunakan tiga virtual pin seperti yang ditampilkan pada Tabel 1.

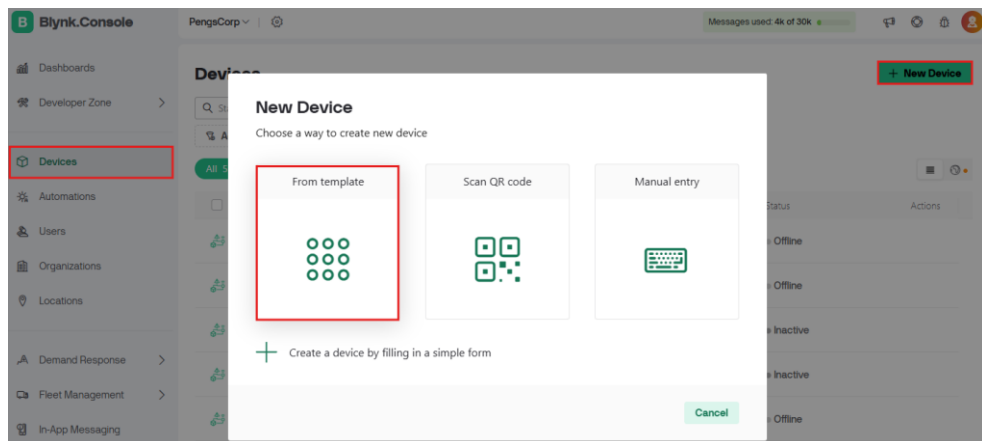
Tabel 1. Konfigurasi virtual pin blynk

Name	Pin	Data Type	Value	Fungsi
Mode Kontrol	V0	Integer	0 - 1	Memilih mode kontrol (manual atau otomatis)
Nilai LDR	V1	Integer	0 - 4095	Mengambil data sensor LDR
Saklar LED	V2	Integer	0 - 1	Kontrol LED secara manual

- f. Selanjutnya masuk ke menu **Dashboard** untuk membuat panel *monitoring* dan kontrol. Pilih sesuai dengan kebutuhan seperti yang sudah *disetting* melalui Datastreams, jika sudah klik **Save**.



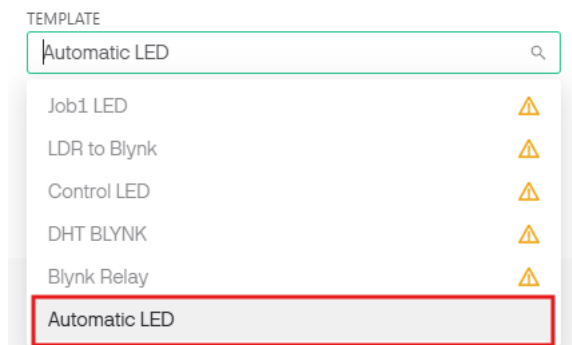
- g. Selanjutnya masuk ke menu **Devices** untuk mengatur perangkat *virtual*.
Klik **New Device** > pilih **From template**.



- h. Pilih Template yang telah dikonfigurasi, lalu klik **Create**.

New Device

Create new device by filling in the form below



- i. Simpan kode tersebut untuk dikonfigurasi melalui Arduino IDE.

New Device Created!



```
#define BLYNK_TEMPLATE_ID "TMPL6Hf8fII4L"
#define BLYNK_TEMPLATE_NAME "Automatic LED"
#define BLYNK_AUTH_TOKEN "VFw4I1KB1--xwg_Bn_l8ebpgIIdb9bQR"
```

Template ID, Template Name, and AuthToken should be declared at the very top of the firmware code.

- j. Konfigurasi Blynk sudah selesai, langkah selanjutnya konfigurasi program di Arduino IDE, masukkan kode berikut.

```
#define BLYNK_TEMPLATE_ID "TMPL6Hf8fII4L"
#define BLYNK_TEMPLATE_NAME "Automatic LED"
#define BLYNK_AUTH_TOKEN "VFw4I1KB1--xwg_Bn_l8ebpgIIdb9bQR"

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

// Ganti dengan WiFi Anda
char ssid[] = "fxx";
char pass[] = "11111111";

const int ldrPin = 34;
const int ledPin = 17;

int ldrValue = 0;
bool ledState = false;
bool autoMode = true; // Default: mode otomatis
int manualControl = 0; // Nilai kontrol manual dari Blynk

#define VPIN_MODE V0 // Virtual pin untuk switch mode (auto/manual)
#define VPIN_LDR V1 // Virtual pin untuk menampilkan nilai LDR
#define VPIN_MANUAL V2 // Virtual pin untuk tombol manual LED

void setup() {
  Serial.begin(9600);
  pinMode(ldrPin, INPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
}

// Mode otomatis/manual dari Blynk (Switch)
BLYNK_WRITE(VPIN_MODE) {
  autoMode = param.asInt();
}

// Kontrol LED manual dari Blynk
BLYNK_WRITE(VPIN_MANUAL) {
  manualControl = param.asInt();
}
```

```

}

void loop() {
  Blynk.run();

  ldrValue = analogRead(ldrPin);

  // Tampilkan nilai LDR ke Serial dan Blynk
  Serial.print("LDR Value: ");
  Serial.println(ldrValue);
  Blynk.virtualWrite(VPIN_LDR, ldrValue);

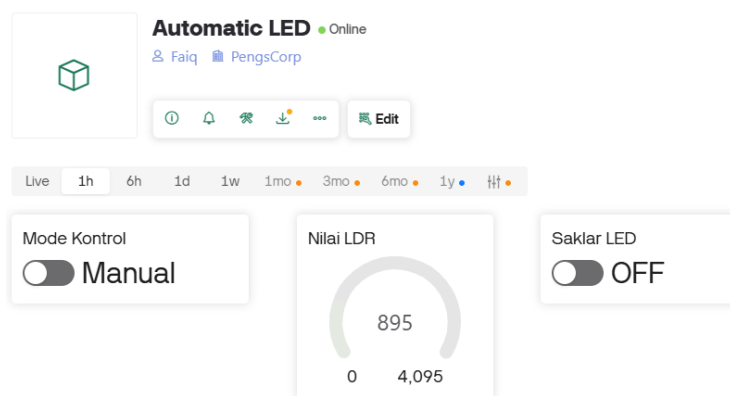
  // Logika kontrol LED
  if (autoMode) {
    // Mode otomatis: LED nyala jika terang
    if (ldrValue > 2500) {
      digitalWrite(ledPin, HIGH);
      ledState = true;
    } else {
      digitalWrite(ledPin, LOW);
      ledState = false;
    }
  } else {
    // Mode manual: kontrol LED via tombol Blynk
    digitalWrite(ledPin, manualControl ? HIGH : LOW);
    ledState = manualControl;
  }

  // Tampilkan status LED ke Serial
  Serial.print("LED Status: ");
  Serial.println(ledState ? "ON" : "OFF");

  delay(1000);
}

```

- k. Hubungkan ESP32 ke laptop, sebelum Verify atau Compile code pastikan Board ESP32 sudah terhubung di Arduino IDE, selanjutnya Upload.
- l. Setelah berhasil Upload, masuk ke Blynk apabila sudah tersambung maka status akan berubah menjadi **Online**.



F. Soal

1. Tampilkan output berdasarkan dari instruksi Langkah Kerja! Tambahkan analisis dan prinsip kerjanya di laporan praktikum!
2. Modifikasi rangkaian menggunakan input sensor DHT dan output aktuator LED (sejumlah 3 LED), asumsikan LED sebuah kecepatan kipas sesuai skenario berikut:
 - Suhu <29 output 1 LED
 - Suhu 29 – 34 output 2 LED
 - Suhu >34 output 3 LED
3. Dokumentasikan hasil pengerjaan pada lembar Kerja Praktikum. Sertakan video hasil praktikum dan source code-nya!
4. Upload lembar kerja Laporan Praktikum (dalam format PDF) dan video dokumentasi di Google Drive.
<https://drive.google.com/drive/folders/1aK1wROwYA5LkRB38jrDirvBtd6uQlPt3?usp=sharing>
5. Ketentuan pengumpulan drive sebagai berikut:
 - Gunakan akun students UNNES
 - Buat folder dengan format “NIM_Nama Lengkap”
 - Upload lembar kerja Laporan Praktikum dan dokumentasi video