LAMPIRAN

```
#include "FastLED.h"

#define NUM_LEDS 114
#define DATA_PIN 13
#define CLOCK_PIN 13

CRGB leds[NUM_LEDS];
bool ledsOn = true; // Variabel global untuk status lampu

// Telegram Library
#include <AsyncTelegram2.h>
// Definisi zona waktu
#define MYTZ "CET-1CEST,M3.5.0,M10.5.0/3"
struct tm sysTime;

#include <WiFiClientSecure.h>
WiFiClientSecure client;
#ifdef ESP8266
#include <ESP8266WiFi.h>
Session   session;
X509List  certificate(telegram_cert);
#endif

AsyncTelegram2 myBot(client);

// variabel untuk menyimpan data pesan telegram
TBMessage msg;

const char* ssid = "WR Elite";
const char* pass = "kamarpenuh";
const char* token = "7029434585:AAHtVbNpoNsnDAqiEZDsJltibBD1qKrMZUM";

int64_t userid = 1380021615; // id user

void setup() {
  // Set console baud rate
  Serial.begin(9600);
  delay(2000);
  FastLED.addLeds<WS2812B, DATA_PIN, GRB>(leds, NUM_LEDS);
  initWifi(); //Inisialisai
  initTelegram(); //Inisialisai Telegram
}

void loop() {
  unsigned long onDuration = 43200;  // 5 detik on
  unsigned long offDuration = 43200; // 5 detik off

  static unsigned long previousMillis = 0;
  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= (ledsOn ? onDuration : offDuration) * 1000) {
    ledsOn = !ledsOn;
    previousMillis = currentMillis;
    if (ledsOn) {
      //sendMessage("Lampu telah menyala.");
      myBot.sendMessage(msg, "Lampu telah menyala.");
    } else {
      //sendMessage("Lampu telah mati.");
      myBot.sendMessage(msg, "Lampu telah mati.");
    }
  }

  if (!ledsOn) {
    for (int i = 0; i < NUM_LEDS; i++) {
      leds[i] = CRGB(0, 0, 0);
    }
  } else {
    int numBlueLEDs = 4;
    int numRedLEDs = 2;
    int repetitions = NUM_LEDS / (numBlueLEDs + numRedLEDs);

    for (int rep = 0; rep < repetitions; rep++) {
      for (int i = rep * (numBlueLEDs + numRedLEDs); i < rep * (numBlueLEDs +
numRedLEDs) + numBlueLEDs; i++) {
        leds[i] = CRGB(0, 0, 255);
      }

      for (int i = rep * (numBlueLEDs + numRedLEDs) + numBlueLEDs; i < rep *
(numBlueLEDs + numRedLEDs) + numBlueLEDs + numRedLEDs; i++) {
        leds[i] = CRGB(255, 0, 0);
      }
    }
  }

  FastLED.show();
  // Handle incoming messages from Telegram
  handleTelegramMessages();
}

void initWifi() {
  WiFi.setAutoConnect(true);
  WiFi.mode(WIFI_STA);

  // terhubung ke titik akses
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print('.');
    delay(500);
  }
  Serial.println("\nWiFi connected");
  Serial.println(WiFi.localIP());
}

void handleTelegramMessages() {
  //  if (bot.getUpdates(bot.last_message_received + 1)) {
  //    String chat_id = String(bot.messages[0].chat_id);
  //    String text = bot.messages[0].text;
  //
  //    if (text == "/status") {
  //      // Proses perintah status di sini
  //      String status = (ledsOn) ? "Lampu menyala" : "Lampu mati";
  //
  //      // Kirim pesan respons ke bot Telegram
  //      bot.sendMessage(chat_id, status, "");
  //    }
  //  }
  // jika ada pesan masuk...
  if (myBot.getNewMessage(msg)) {
    MessageType msgType = msg.messageType;

    // Menerima pesan teks
    if (msgType == MessageText) {
      String msgText = msg.text;
      Serial.print("Text message received: ");
      Serial.println(msgText);

      if (msgText == "/status") {
        //Proses perintah status di sini
        String status = (ledsOn) ? "Lampu menyala" : "Lampu mati";
        myBot.sendMessage(msg, status);
      }

      if (msgText == "/start") {
        String welcome = "Hello\n";
        welcome += "Ini adalah bot untuk mengetahui kondisi Lampu\n\n";
        welcome += "/status : Mengetahui status lampu\n";
        myBot.sendMessage(msg, welcome);
      }
    }
  }
}

void initTelegram() {
#ifdef ESP8266
  // Sinkronkan waktu dengan NTP, untuk memeriksa sertifikat Telegram dengan
benar
  configTime(MYTZ, "time.google.com", "time.windows.com", "pool.ntp.org");
  //Atur sertifikat, sesi, dan beberapa properti klien dasar lainnya
  client.setSession(&session);
  client.setTrustAnchors(&certificate);
  client.setBufferSizes(1024, 1024);
#elif defined(ESP32)
  // Sinkronkan waktu dengan NTP
  configTzTime(MYTZ, "time.google.com", "time.windows.com", "pool.ntp.org");
  client.setCACert(telegram_cert);
#endif

  // Atur properti bot Telegram
  myBot.setUpdateTime(1000);
  myBot.setTelegramToken(token);

  // Periksa apakah semuanya baik-baik saja
  Serial.print("\nTest Telegram connection... ");
  myBot.begin() ? Serial.println("OK") : Serial.println("NOK");

  // https://core.telegram.org/bots/api#formatting-options
  myBot.setFormattingStyle(AsyncTelegram2::FormatStyle::HTML /*
MARKDOWN */);

  // Kirim pesan selamat datang ke pengguna jika sudah siap
  char welcome_msg[128];
  snprintf(welcome_msg, sizeof(welcome_msg),
           "BOT @%s online.\n/start for command list.",
           myBot.getBotName());
  // Periksa userid dengan bantuan bot @JsonDumpBot atau @getidsbot (bekerja juga dengan grup)
  // https://t.me/JsonDumpBot  or  https://t.me/getidsbot
  myBot.sendTo(userid, welcome_msg);
}
```