

```

#include <ESP8266WiFi.h> // for ESP8266
#include <BlynkSimpleEsp8266.h> // for ESP8266
#include <SimpleTimer.h>
#define BLYNK_MSG_LIMIT 0 // In theory this disables the "anti-flood governor"...
                             might only work on Local Server

BlynkTimer timer;

char auth[] = "T_FKKbEVNavVz6GWB0e-Ljlq_3Fgyxfa"; // Auth Token
char ssid[] = "I'm AweSome"; // Your WiFi network name
char pass[] = "~Arrow818"; // Your WiFi network password
char server[] = "blynk-cloud.com"; // Cloud Server address
int port = 8080; // MCU Hardware Device port

int LEDArduino = D5;

void setup() {
  pinMode(LEDArduino, OUTPUT); // Setup GPIO2 (D4 the ESP On-Board LED) as an
  output pin

  // Connect to your WiFi network, then to the Blynk Server
  WiFi.begin(ssid, pass);
  Blynk.config(auth, server, port);
  Blynk.connect();

  // This dual nested timer routine will constantly blink both a Virtual LED and the
  onboard LED of an ESP8266 Dev Board
  timer.setInterval(30000L, []() { // Start 1st Timed Lambda Function - Turn ON LED
  every 30 seconds (repeating)
    timer.setTimeout(180L, []() { // Start 2nd Timed Lambda Function - Turn OFF LED 1
    second later (once per loop)
      Blynk.virtualWrite(V0, 0); // Turn Virtual LED OFF
      digitalWrite(LEDArduino, HIGH); // Turn ESP On-Board LED OFF
    }); // END 2nd Timer Function
    Blynk.virtualWrite(V0, 255); // Turn Virtual LED ON
    digitalWrite(LEDArduino, LOW); // Turn ESP On-Board LED ON
  }); // END 1st Timer Function
}

BLYNK_WRITE(V0) {
  int pinValue = param.asInt();
  Serial.print("Pin number: ");
  Serial.println(LEDArduino);
  Serial.println(pinValue);
  if (pinValue == 1) {
    digitalWrite(LEDArduino, HIGH); // Turn LED on.
  } else {

```

```
digitalWrite(LEDArduino, LOW); // Turn LED off.  
}  
  
}  
  
void loop() {  
  Blynk.run();  
  timer.run();  
}
```