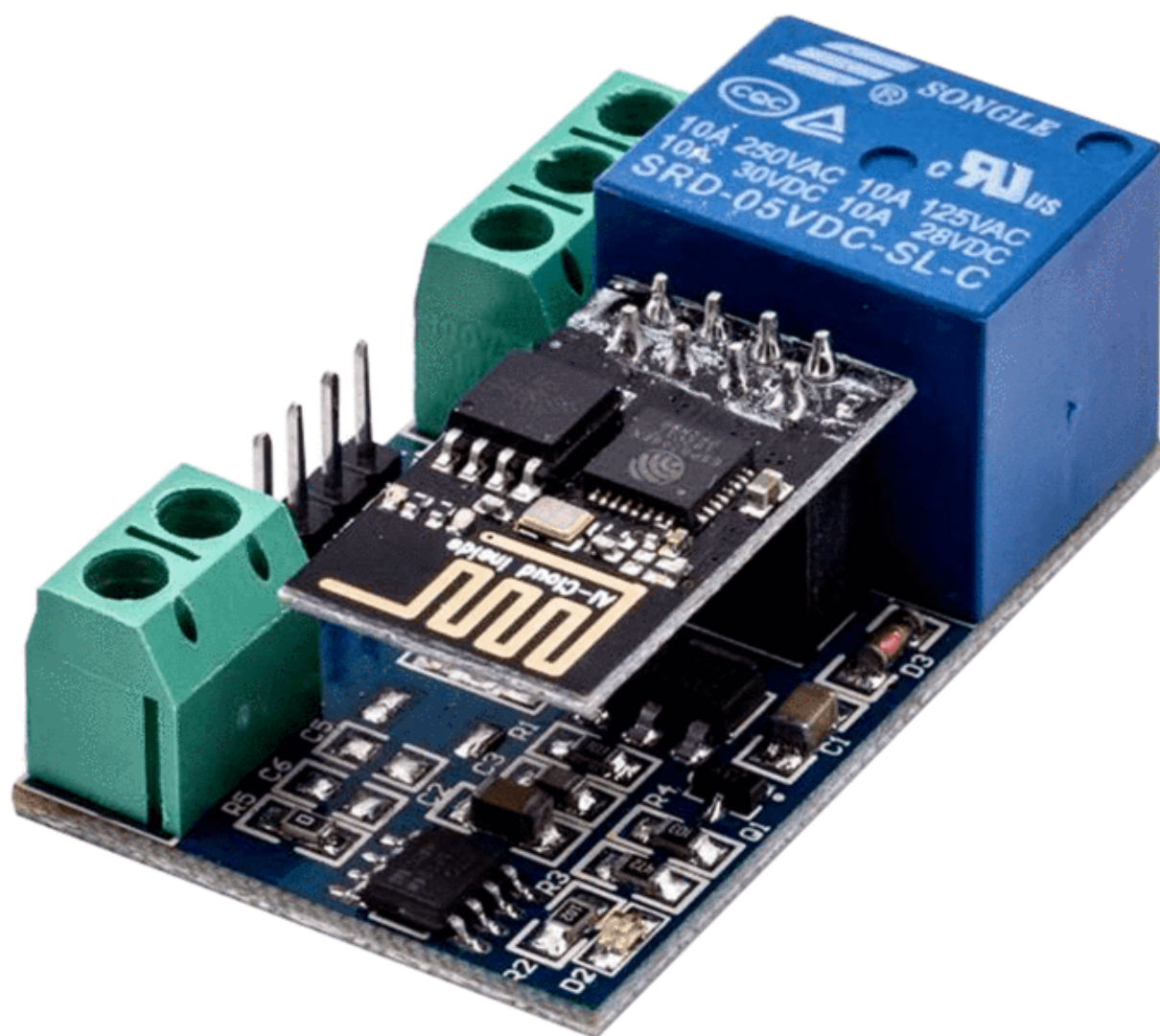


AZ-Delivery

Benvenuto!

Grazie per aver acquistato il nostro AZ-Delivery ESP8266-01S con modulo Relé. Nelle pagine seguenti, ti illustreremo come utilizzare e configurare questo pratico dispositivo.

Buon divertimento!



Az-Delivery

Questo modulo è un relè controllato tramite wifi. Il relè può accendere o spegnere i dispositivi sia in AC che in DC (10A 250V - AC, e 10A 30V - DC).

Un relè è un interruttore elettromagnetico azionato da una corrente elettrica relativamente piccola che può accendere o spegnere una corrente elettrica molto più grande. Il cuore di un relè è un elettromagnete, una bobina di filo che diventa un magnete temporaneo quando l'elettricità scorre attraverso di esso. Puoi pensare a un relè come a una specie di leva elettrica: lo accendi con una piccola corrente e accende ("fa leva") un altro apparecchio usando una corrente molto più grande.

Perché è utile? Come suggerisce il nome, molti sensori sono dei componenti di apparecchiature elettroniche incredibilmente sensibili e producono solo piccole correnti elettriche. Ma spesso ne abbiamo bisogno per guidare pezzi più grandi di apparati che usano correnti più grandi. I relè colmano il divario, consentendo alle piccole correnti di attivare quelle più grandi. Ciò significa che i relè possono funzionare come interruttori (accendendo e spegnendo le cose) o come amplificatori (convertendo piccole correnti in più grandi).

Questo ha quindi due parti, una funge da alimentatore con relè, e l'altra funge da microcontrollore con interfaccia seriale e wifi. Queste due parti possono essere disconnesse l'una dall'altra. La seconda parte è in realtà il modulo ESP8266-01S, quindi può essere sostituito in caso di malfunzionamento.

Az-Delivery

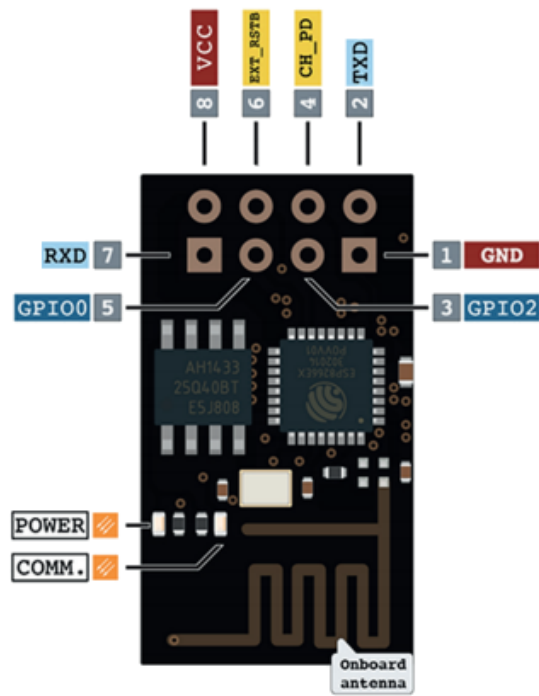
Collegate insieme queste due parti creano un ottimo modulo per l'IoT. Puoi rendere la tua casa una casa intelligente. Con esso puoi accendere/spegnere dispositivi o luci in casa via Internet(wifi)

Il modulo ESP8266 è un System on a Chip (SoC), prodotto dalla società cinese Espressif. Consiste in un microcontrollore Tensilica L106 a 32 bit e un ricetrasmittitore wifi. Ha 11 pin GPIO (General Purpose Input/Output) e anche un ingresso analogico. Ciò significa che puoi programmarlo come un qualsiasi normale Arduino o qualsiasi altro microcontrollore. E la cosa migliore dell'ESP8266 è che con esso ottieni una comunicazione wifi, quindi puoi usarlo per connetterti alla tua rete wifi, a Internet, ospitare un server web con pagine Web reali, lasciare che il tuo smartphone si colleghi ad esso, ecc.

L'ESP8266 è un fantastico chip WiFi che può essere utilizzato in diverse applicazioni di home automation. Grazie al potente processore da 80 MHz e all'ampia memoria da 1 MB, l'ESP8266 può funzionare anche in modo autonomo.

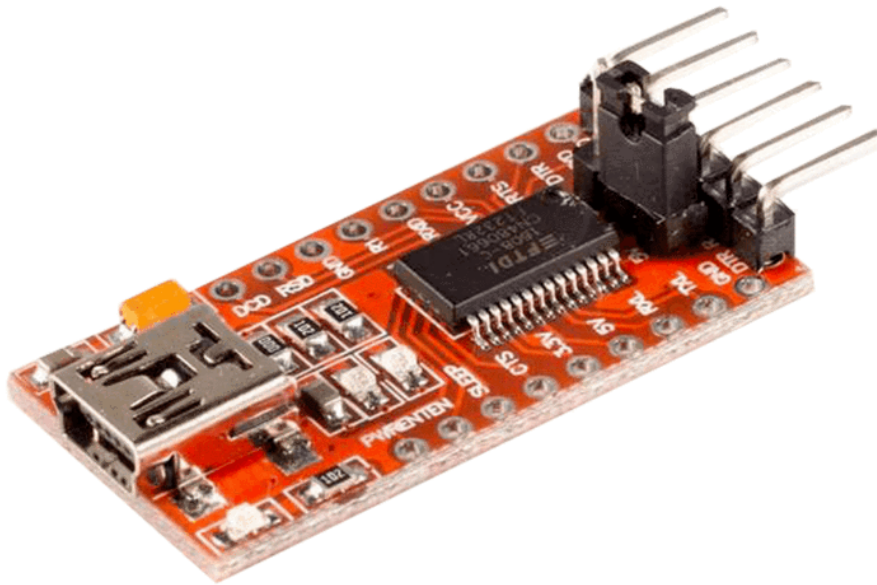
Az-Delivery

Esistono molte schede di sviluppo diverse basate sull'ESP8266, quindi l'ESP8266-01S è una di queste. Ha 8 pin (designazione dei pin nell'immagine qui sotto



Come puoi vedere, possiamo controllare due pin GPIO, GPIO0 e GPIO2. Il pin di controllo del relè è collegato al pin GPIO2 (tramite transistor). Quindi tutto ciò che dobbiamo fare è accendere o spegnere il pin GPIO2 del modulo ESP8266 per accendere o spegnere il dispositivo collegato al relè.

Per programmare l'ESP8266-01S dobbiamo usare i pin RX e TX e collegarli al dispositivo con interfaccia seriale come il modulo FTDI o l'Arduino se non hai FDTI.



Adattatore FTDI

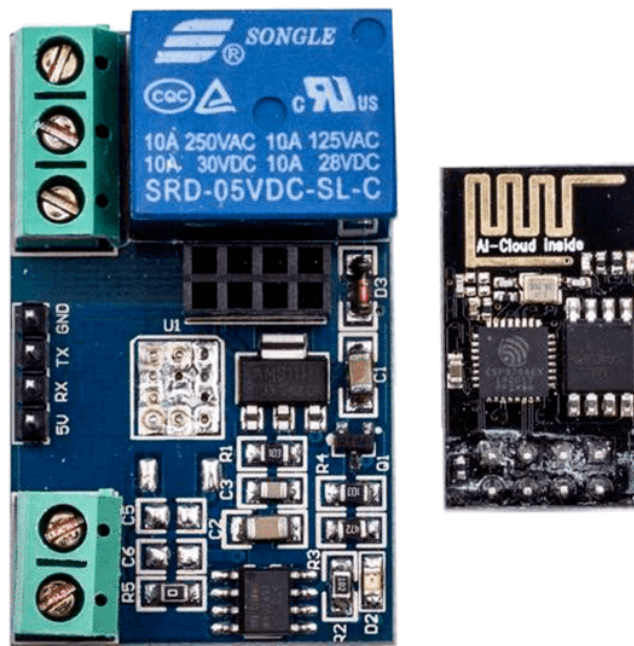
<https://www.az-delivery.de/products/ftdi-adapter-ft232rl?ls=en>

Esistono due modi per programmare l'ESP8266-01S. Il primo consiste nell'utilizzare i comandi AT (ATtentionc commands) e il secondo è in realtà la programmazione del chip ESP8266 utilizzando l'IDE Arduino.

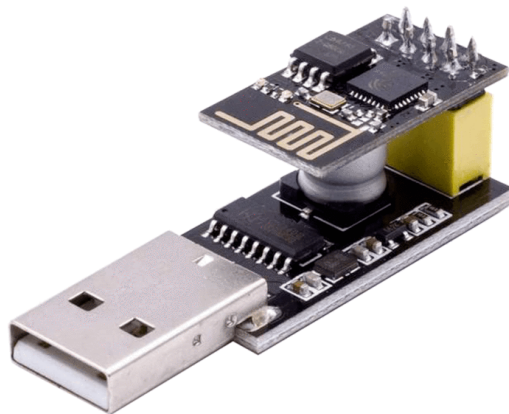
Poiché il secondo modo offre molta più libertà in ciò che è possibile creare, in questo eBook tratteremo solo questo modo di programmare L'ESP8266-01S.

Az-Delivery

Per programmare L'ESP8266-01S usando Arduino IDE dobbiamo disconnettere L'ESP8622-01S dal modulo e collegarlo con FTDI o Arduino Uno. Il modo migliore è utilizzare i Cavetti jumper da maschio a femmina, disponibile anche un adattatore USB o una scheda di breakout della breadboard.

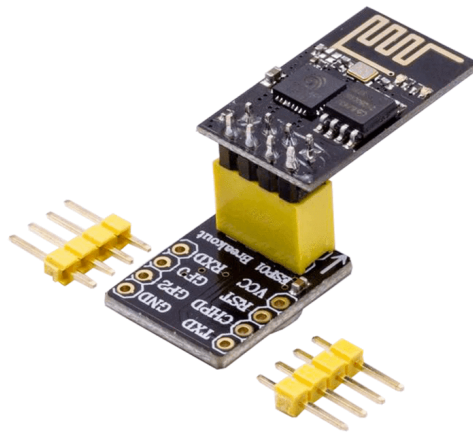


ESP8266-01S scollegato dal modulo



Adattatore USB con ESP8266-01S

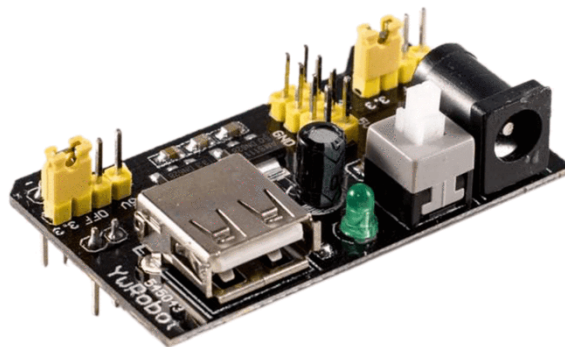
<https://www.az-delivery.de/products/esp8266-01s-mit-usb-adapter?ls=en>



Breadboard breakout board con ESP8266-01S

<https://www.az-delivery.de/products/esp8266-01s-mit-breadboardadapter?ls=en>

Useremo la breadboard breakout board e Arduino Uno, perché è facile per noi, ma qualsiasi modo tu scelga è simile. Assicurati di alimentare l'ESP8266-01S con un alimentatore separato, perché il regolatore di tensione a bordo della maggior parte delle schede Arduino non è abbastanza potente per alimentare sia la scheda Arduino che la scheda ESP8266-01S. Puoi utilizzare il nostro alimentatore per breadboard MB102. È un alimentatore molto semplice e maneggevole che accetta input da 6 a 12V DC e può produrre sia +3.3V che +5V.

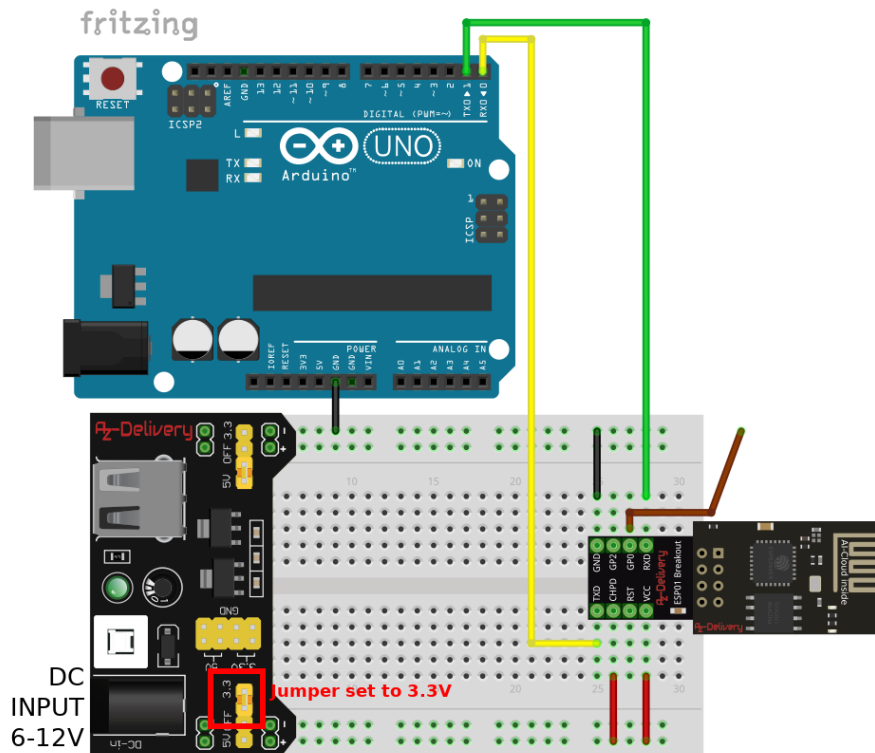


Alimentatore per breadboard MB102

<https://www.az-delivery.de/products/mb102-breadboard?ls=en>

Az-Delivery

Se si utilizza Arduino Uno come adattatore seriale, collega tutto come nello schema di collegamento in basso:



Imposta il jumper di voltaggio dell'MB102 a 3.3V!

Pin ESP > Pin Arduino

TXD > D0 (RXD)

RXD > D1 (TXD)

GND > GND

Filo giallo

Filo verde

Filo nero

Collega anche gli altri pin ESP così:

Pin VCC e CH_PD alla +3.3V

GND a GND

Fili rossi

Filo nero

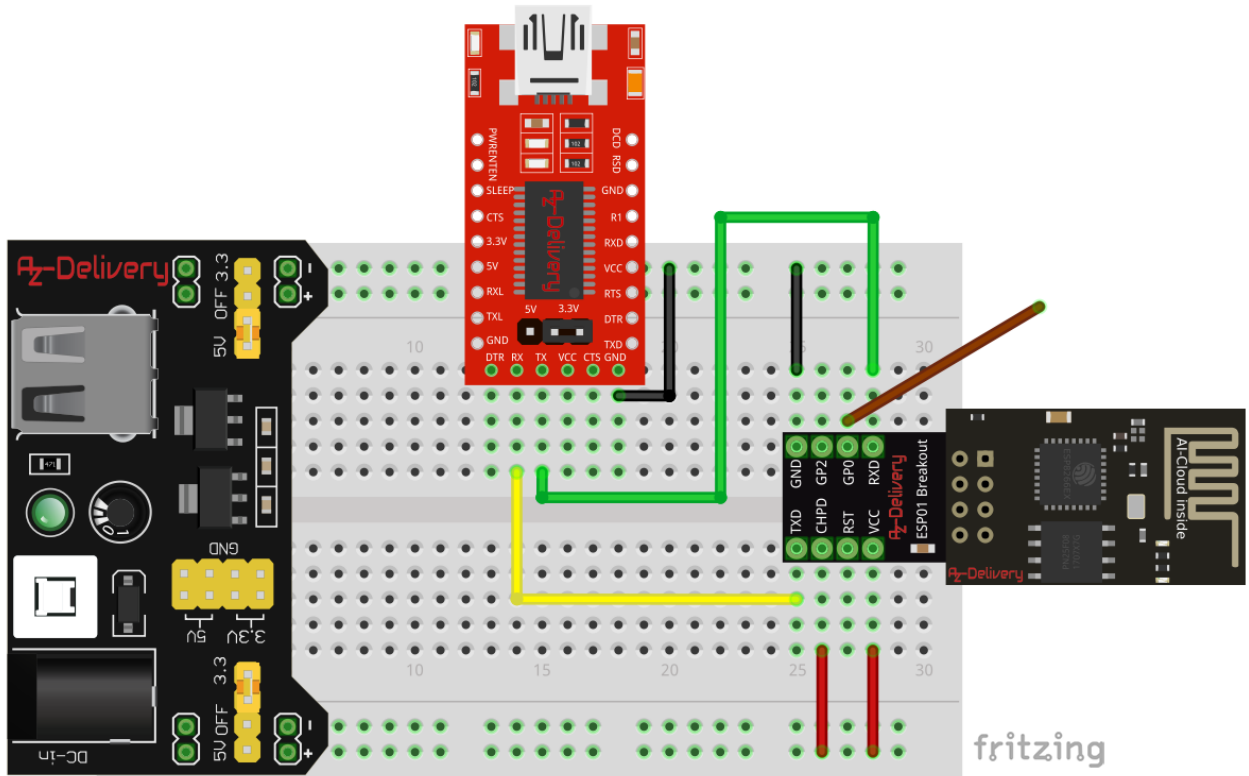
Pin GPIO0 scollegato per modalità normale,

e per modalità programmazione collega GPIO0 al GND

Filo marrone

Az-Delivery

E se utilizzi l'adattatore FTDI, collega tutto come nello schema di collegamento in basso:



Imposta il jumper di voltaggio dell'MB102 a 3.3V!

Pin ESP > Pin FTDI

TXD > RX

RXD > TX

GND > GND

Collega anche gli altri pin ESP così::

Pin VCC e CH PD alla +3.3V

GND a GND

Pin GPIO0 scollegato per modalità normale,

e per modalità programmazione collega GPIO0 al GND

Filo giallo

Filo verde

Filo nero

Fili rossi

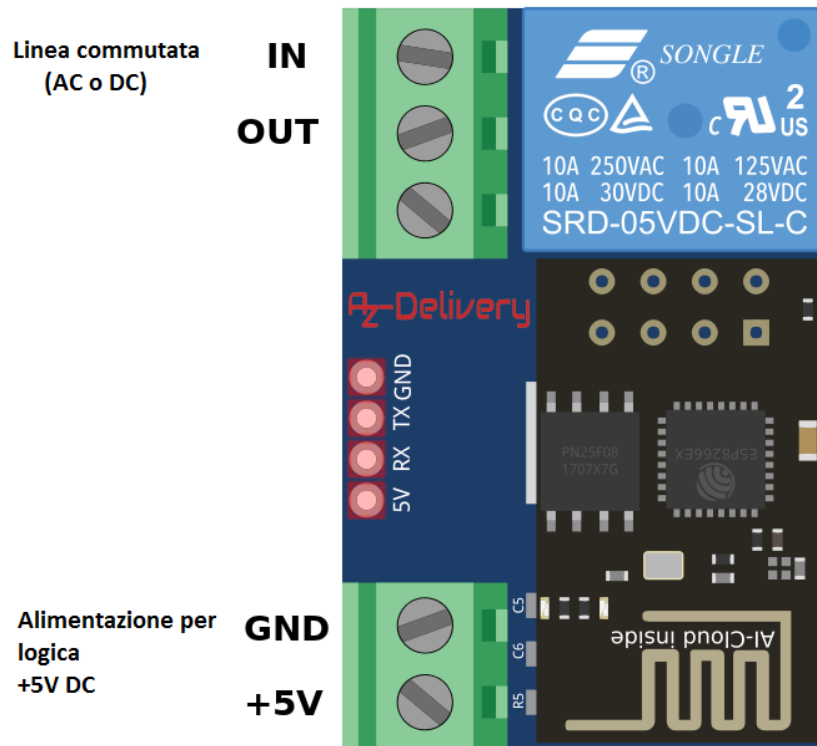
Filo nero

Filo marrone

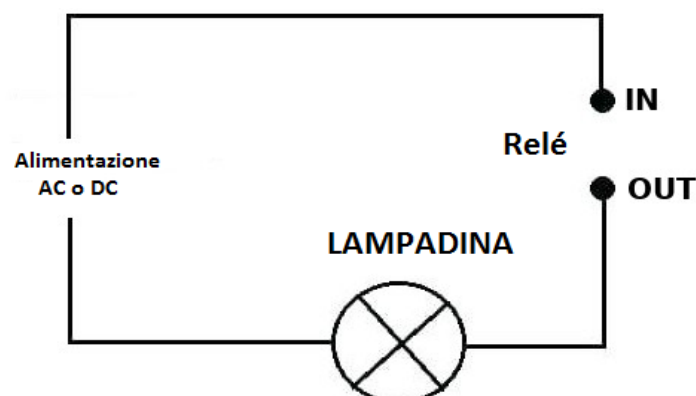


Per la modalità normale il pin GPIO0 deve essere scollegato all'avvio (filo marrone), ma quando vogliamo programmare l'ESP8266-01S, dobbiamo prima spegnere il modulo (scollegandolo dall'alimentazione), collegare il pin GPIO0 a GND e quindi ricollegarlo all'alimentazione (avviarlo). Per tornare alla modalità normale, è necessario spegnere il modulo, quindi scollegare il pin GPIO0 (lasciarlo scollegato; collegato a nulla) e riaccendere il modulo, ricollegarlo all'alimentazione (avviarlo).

Come collegare il modulo completo



Ora ci sono due parti su questo modulo, una che fa da interruttore e l'altra è quella logica. Per alimentare la parte logica abbiamo bisogno di una tensione di + 5 V CC e GND (terra) (angolo in basso a sinistra della foto sopra). E per creare un circuito di commutazione, basta tagliare qualsiasi linea che devi commutare e inserire un'estremità del taglio nel terminale a vite IN e l'altra estremità del taglio nel terminale a vite OUT (angolo in alto a sinistra della foto). Ecco un esempio di circuito per l'accensione di una lampadina in casa:

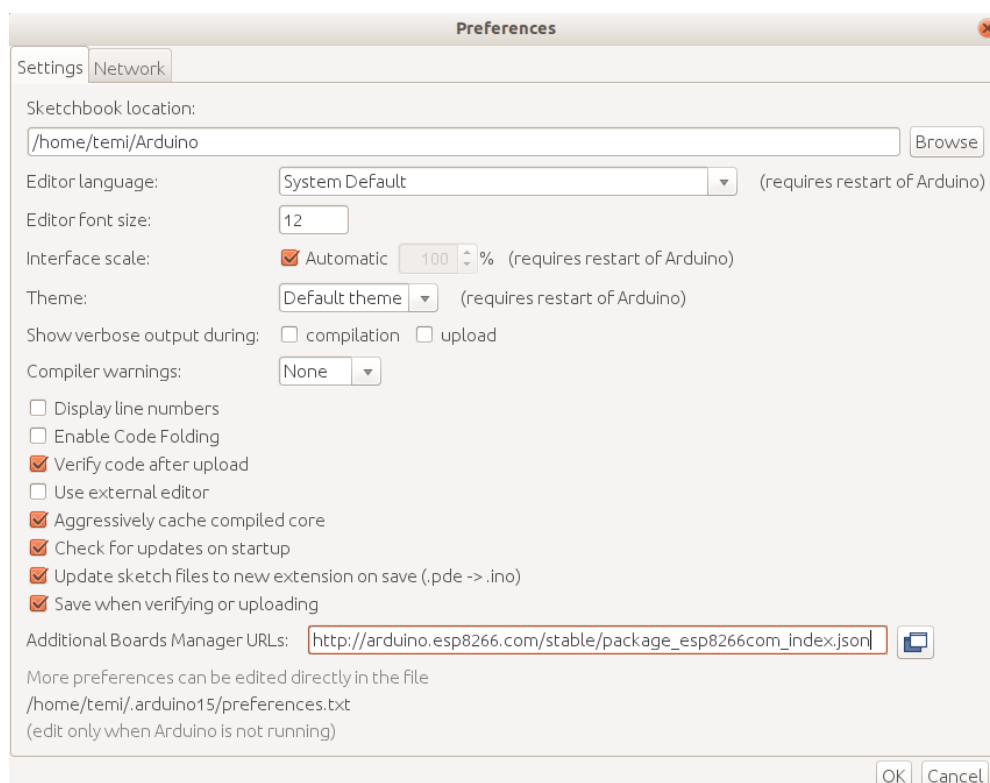


Az-Delivery

Software

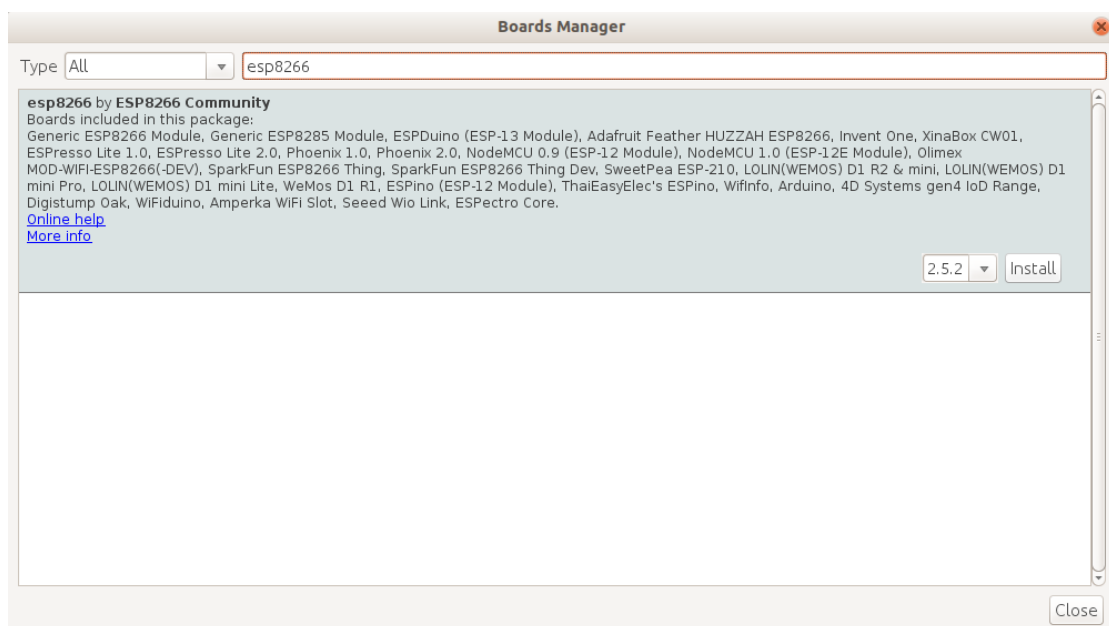
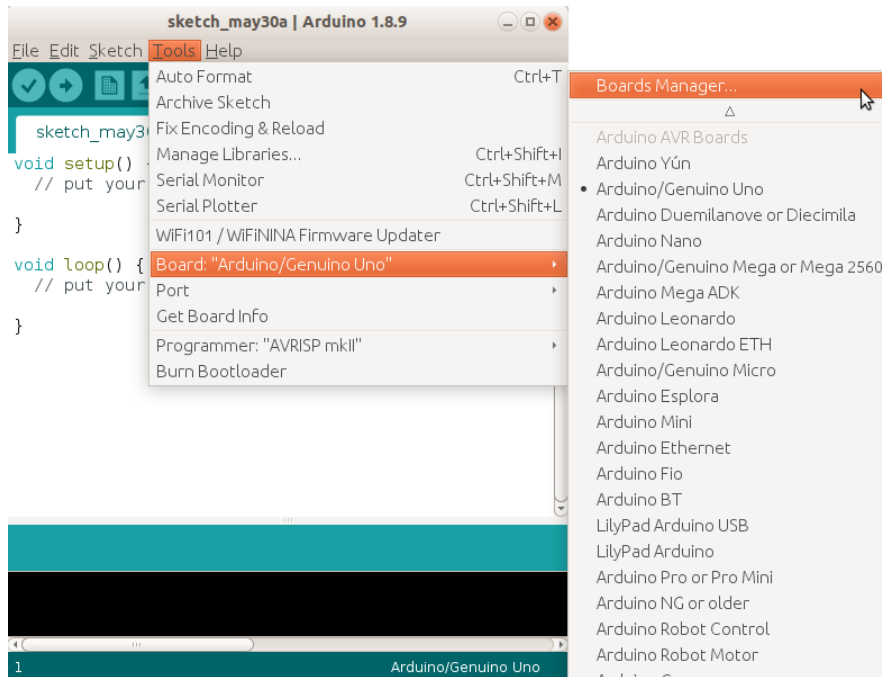
Ora per usare l'ESP8266 con Arduino IDE dobbiamo prima configurare l'Arduino IDE. Il primo passo è scaricare e installare l'Arduino IDE. Se lo LA stai già utilizzando, ottimo, in caso contrario, vai su <https://www.arduino.cc/en/Main/Software> scaricala e installala.

Per programmare l'ESP8266-01S, è necessario un plug-in per l'IDE Arduino, che può essere scaricato da GitHub [GitHub](#) manualmente, ma è più semplice aggiungere l'URL nell'IDE di Arduino. Apri l'IDE, vai su *File > Preferences*, copia l'URL http://arduino.esp8266.com/stable/package_esp8266com_index.json nel campo URL "Additional Board Manager" . (Puoi aggiungere più URL, separandoli con virgole.)



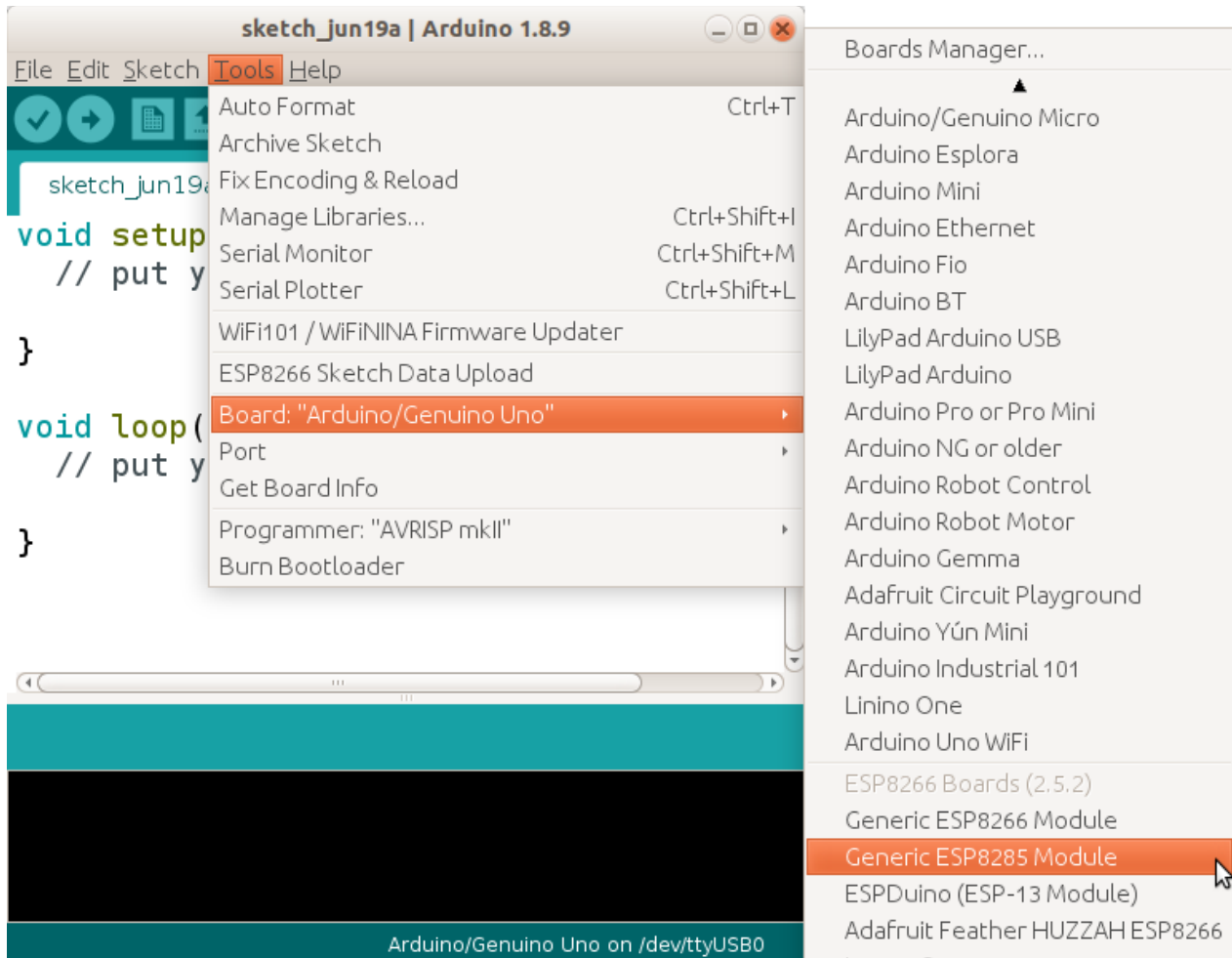
Az-Delivery

Vai poi su *Tools > Board > Board Manager* e cerca "esp8266". Seleziona la versione più recente e fai clic su Installa. Dopo questo molti esempi di sketch, verranno installate le schede.



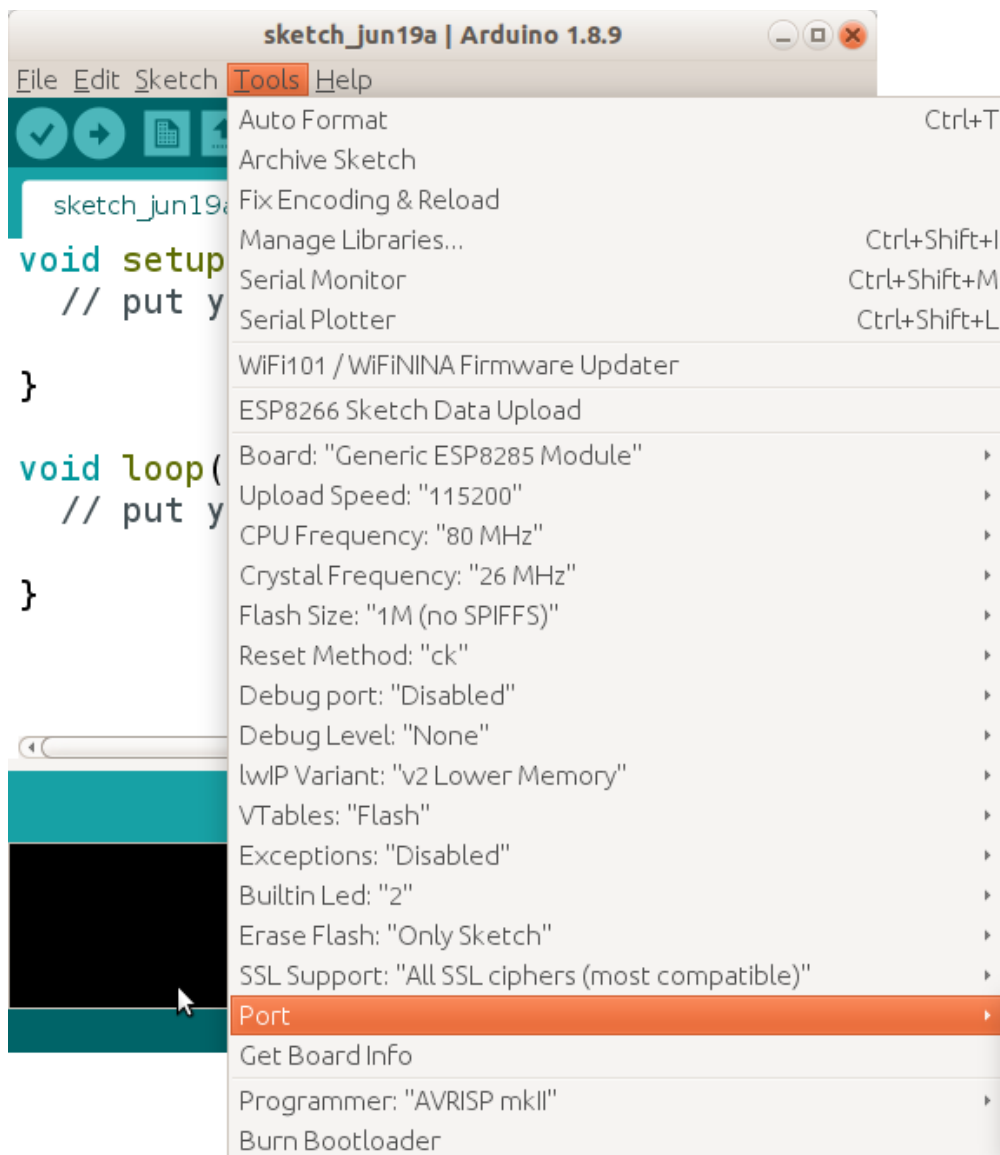
Az-Delivery

Riavvia la tua Arduino IDE. Ora dobbiamo impostare quale scheda programmare. *Vai su Tools > Board > {board name} e scegli "Generic ESP8266 Module".*



Az-Delivery

Quando scegli questa scheda ci sono anche altre opzioni che puoi impostare. Quando si aprono i menu a discesa *"Tools"* dopo che avrai scelto *"Generic ESP8266 Module"*, ci saranno diverse opzioni che puoi impostare. Ai fini di questo eBook lascia tutto così com'è, perché non entreremo nei dettagli. Basta scegliere la porta su cui è connessa la tua scheda.



Az-Delivery

Ora siamo pronti per programmare il modulo ESP8266-01S. Ecco lo sketch:

```
#include <ESP8266WiFi.h>

const char* ssid = "ssid";//type your ssid
const char* password = "password"; //type your password


int relayPin = 2; // GPIO2 of ESP8266
WiFiServer ESPserver(80);//Service Port
void setup() {
    Serial.begin(115200);
    pinMode(relayPin, OUTPUT);
    digitalWrite(relayPin, HIGH);
    Serial.println();
    Serial.println();
    Serial.print("Connecting to: ");
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    delay(5000);
    /* The following four line of the
       code will assign a Static IP Address to
       the ESP Module. If you do not want this,
       comment out the following four lines.*/

    IPAddress ip(192,168,0,254);
    IPAddress gateway(192,168,0,1);
    IPAddress subnet(255,255,255,0);
    WiFi.config(ip, gateway, subnet);
    delay(5000);
}
```

Az-Delivery

```
while(WiFi.status() != WL_CONNECTED) {
    delay(100);
    Serial.print("*");
}
Serial.println("");
Serial.println("WiFi connected");
// Start the server
ESPserver.begin();
Serial.println("Server started");
// Print the IP address
Serial.print("The URL to control ESP8266: ");
Serial.print("http://");
Serial.print(WiFi.localIP());
}

void loop() {
    // Check if a client has connected
    WiFiClient client = ESPserver.available();
    if(!client) {
        return;
    }
    // Wait until the client sends some data
    Serial.println("New Client");
    while(!client.available()) {
        delay(1);
    }

    // Read the first line of the request
    String request = client.readStringUntil('\r');
    Serial.println(request);
    client.flush();
}
```

Az-Delivery

```
// Match the request
int value = LOW;
if(request.indexOf("/RELAYON") != -1) {
    Serial.println("LAMP is ON");
    digitalWrite(relayPin, LOW);
    value = LOW;
}
if(request.indexOf("/RELAYOFF") != -1) {
    Serial.println("RELAY is OFF");
    digitalWrite(relayPin, HIGH);
    value = HIGH;
}

// Return the response
client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println(""); // IMPORTANT
client.println("<!DOCTYPE HTML>");
client.println("<html>");
client.print("Status of the RELAY: ");
if(value == LOW) {
    client.print("ON");
}
else {
    client.print("OFF");
}
delay(1);
//client.stop();
Serial.println("Client disconnected");
Serial.println("");
}
```

Az-Delivery

Ora spieghiamo alcuni punti:

Innanzitutto queste due variabili sono il nome e password della tua rete wifi domestica su cui si collegherà il modulo ESP8266-01St:

```
const char* ssid = "ssid";  
const char* password = "password";
```

Quindi definiamo su quale pin relay GPIO è collegato:

```
int relayPin = 2;
```

Dopo di che definiamo il server oggetto e la porta per esso:

```
WiFiServer ESPserver(80);
```

Nella funzione di configurazione avviamo la connessione wifi. Per avviare la connessione usiamo:

```
WiFi.begin(ssid, password);
```

e attendiamo 5 secondi per stabilire la connessione.

Le successive quattro righe sono per l'indirizzo IP statico. Ne abbiamo bisogno per l'app Android. È molto più semplice comunicare con il server con IP e app statici perché ogni volta che il server si connette alla rete wifi otterrà un nuovo indirizzo IP. Pertanto, se utilizziamo un indirizzo IP statico, il server Web avrà sempre lo stesso IP.

```
IPAddress ip(192,168,0,254);  
IPAddress gateway(192,168,0,1);  
IPAddress subnet(255,255,255,0);  
WiFi.config(ip, gateway, subnet);
```

e quindi attendere 5 secondi affinché avvengano le modifiche.

Az-Delivery

Dopo di che aspettiamo che avvenga la connessione wifi (mentre il ciclo è in corso) e lo stampiamo sull'interfaccia seriale. Successivamente, avviamo l'ESPserver con:

```
ESPserver.begin();  
and print the IP address  
Serial.print(WiFi.localIP());
```

Nella funzione loop controlliamo prima se il client (la nostra app Android) è connesso al server e se è collegato lo stampiamo sull'interfaccia seriale.

Dopodiché attendiamo l'input del client, attendiamo che l'app invii *"/RELAYON"* or *"/RELAYOFF"*, e quando lo invia, lo stampiamo sull'Interfaccia Seriale e attiviamo o disattiviamo il relè.

Il set di istruzioni che inizia con `client.println` sono per lo stato relé. Questo viene reinviato all'app Android.

E alla fine della funzione loop aspettiamo che il client si disconnetta e lo stampiamo sull'interfaccia Seriale.

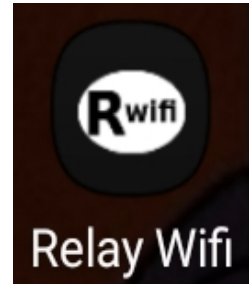
App Android "Relé via Wifi"

Questa app è stata sviluppata utilizzando l'app inventor 2 del MIT: ai2.appinventor.mit.edu/

È stata sviluppata per controllare i relè usando l'ESP8266-01S, come questo modulo di cui scriviamo. L'app è gratuita da usare per chiunque nel modo che preferisce.

Puoi scaricarla da Google Play, basta cercare "Relay via Wifi", o usare il seguente link:

https://play.google.com/store/apps/details?id=appinventor.ai_cvijanovic.RelayWifi



Come usare l'app?

Innanzitutto, devi impostare un indirizzo IP, scrivi l'indirizzo IP statico che abbiamo impostato nello sketch Arduino. L'app non ha convalida per l'indirizzo IP tranne per il giusto numero di caratteri che sono in input (da 7 a 15 caratteri in un indirizzo IP valido). Quindi assicurati di scrivere un IP valido. Dopo aver inserito l'indirizzo IP premi il pulsante "Connect". Se hai inserito un IP valido potrai usare i pulsanti "Turn ON" e "Turn OFF", che controllano lo stato del relé.

Relay Control

IP Address:

Connect

HELP

Turn ON

Turn OFF

Relay Status

Az-Delivery

C'è un altro pulsante chiamato "*HELP*". Premendo il pulsante di aiuto, si aprirà una nuova finestra, spiegherà alcune cose per le persone che non hanno letto questo eBook.

C'è un'altra opzione per controllare il relé in questo modo, ma senza l'app per Android. Se utilizzi il laptop (o qualsiasi altro dispositivo che utilizza il browser per connettersi a Internet, Android o iPhone, ecc.) per connetterti alla stessa rete Wi-Fi di ESP8266-01S, puoi scrivere nel browser:

<http://192.168.0.254/RELAYON>

per accendere il relé e

<http://192.168.0.254/RELAYOFF>

per spegnere il relé.

Ce l'hai fatta, ora puoi usare il modulo per i tuoi progetti.



E ora è tempo di imparare e di creare dei Progetti da solo. Lo puoi fare con l'aiuto di molti script di esempio e altri tutorial, che puoi trovare in internet.

Se stai cercando dei prodotti di alta qualità per il tuo Arduino e Raspberry Pi, AZ-Delivery Vertriebs GmbH è l'azienda giusta dove potrai trovarli. Ti forniremo numerosi esempi di applicazioni, guide di installazione complete, e-book, librerie e l'assistenza dei nostri esperti tecnici.

<https://az-delivery.de>

Buon divertimento!

Impressum

<https://az-delivery.de/pages/about-us>