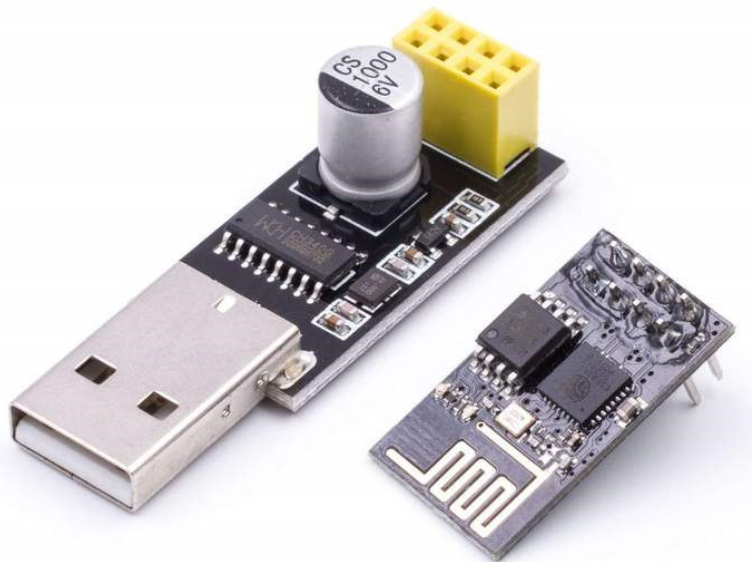


Benvenuto!

E grazie per aver acquistato ESP8266-01 di AZ-Delivery. Nelle prossime pagine ti guideremo durante i primi passi della programmazione.

Buon divertimento!



AZ-Delivery
Ihr Experte für Mikroelektronik

ESP8622-01 ha una vasta portata grazie a WLAN 802.11 b/g/n Standard e può essere utilizzato universalmente. Il modulo supporta 3 modalità operative: Router WLAN (AP), Client WLAN (STA) ed entrambi allo stesso tempo (AP + STA)!

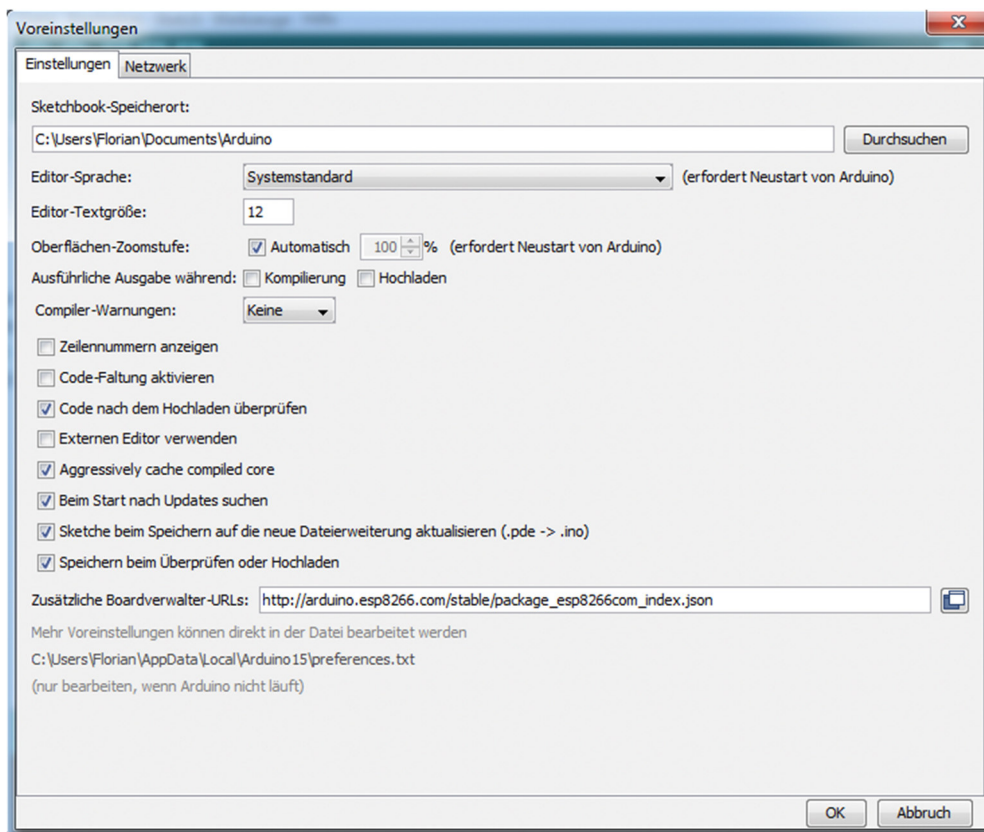
Il potente processore da 80 MHz e la memoria da 1 MB rendono ESP8266-01 adatto a diversi tipi di utilizzo. In confronto al fratello maggiore (ESP8266-12E) è limitato da 2 pin GPIO.


Preparazione del software:

Se ancora non disponete del software Arduino che vediamo installato in questo passaggio, è possibile scaricarlo su <https://www.arduino.cc/en/Main/Software#> e installarlo sul PC. Invece, i driver per il CH340 dovrebbero essere già installati, in caso contrario puoi trovarli qui: [CH340](#).

Una volta soddisfatti tutti i requisiti di base, iniziamo con l'installazione del software. Prima di tutto, il software Arduino ha bisogno di tutte le informazioni su ESP8266-01S, possiamo farlo inserendo il seguente indirizzo sotto "Preferenze" > "URL amministratore di bordo aggiuntivo":

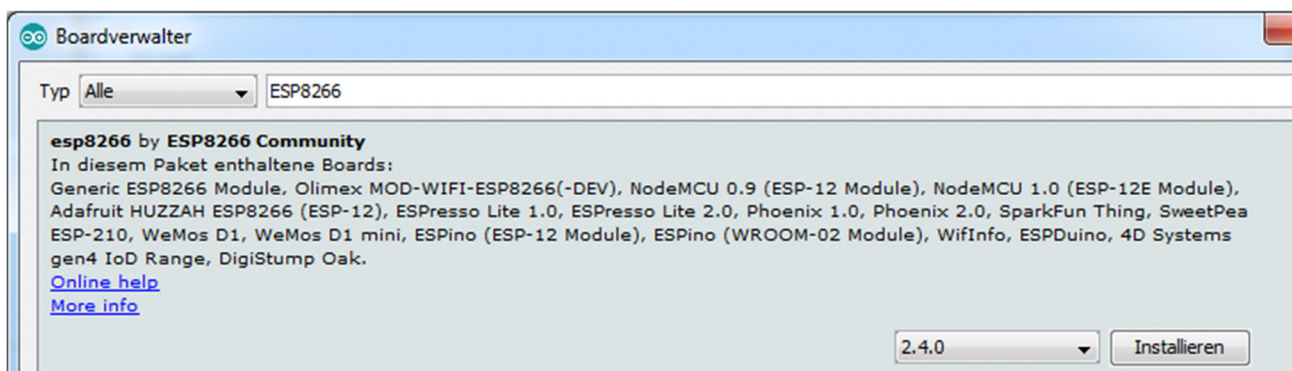
[->http://arduino.esp8266.com/versions/2.4.1/package_esp8266com_index.json](http://arduino.esp8266.com/versions/2.4.1/package_esp8266com_index.json)



Se hai già inserito un link, fai clic sul pulsante  e aggiungi una nuova riga nella finestra.

Confermare la voce con "OK".

Una volta terminato, vai su "Strumenti"> "Scheda"> "Amministratore scheda" e installare la libreria ESP8266. In "amministratore scheda" inserire "ESP8266" nella barra di ricerca in alto a destra e troverai il pacchetto di ESP8266 Community. Selezionalo e fai clic su Installa.



Dopo l'installazione uscirà la scritta INSTALLATO.

esp8266 by ESP8266 Community Version 2.4.0 INSTALLED

In diesem Paket enthaltene Boards:

Generic ESP8266 Module, Olimex MOD-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Adafruit HUZZAH ESP8266 (ESP-12), ESPresso Lite 1.0, ESPresso Lite 2.0, Phoenix 1.0, Phoenix 2.0, SparkFun Thing, SweetPea ESP-210, WeMos D1, WeMos D1 mini, ESPino (ESP-12 Module), ESPino (WROOM-02 Module), WifInfo, ESPDuino, 4D Systems gen4 IoT Range, DigiStump Oak.

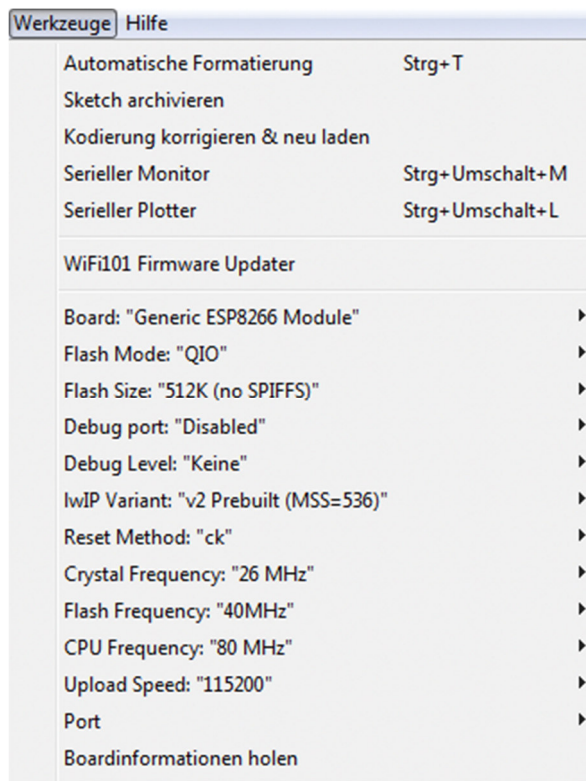
[Online help](#)

[More info](#)

Il prossimo passo è selezionare la scheda giusta:
Strumenti>

Scheda: "Moduli generici ESP8266"

Porta: "COMxx" (qui la tua porta dell'adattatore seriale)



Ora che tutte le impostazioni di base sono attive, si procede con il cablaggio. Per poter programmare ESP8266-01, bisogna prima impostarlo in modalità di programmazione.

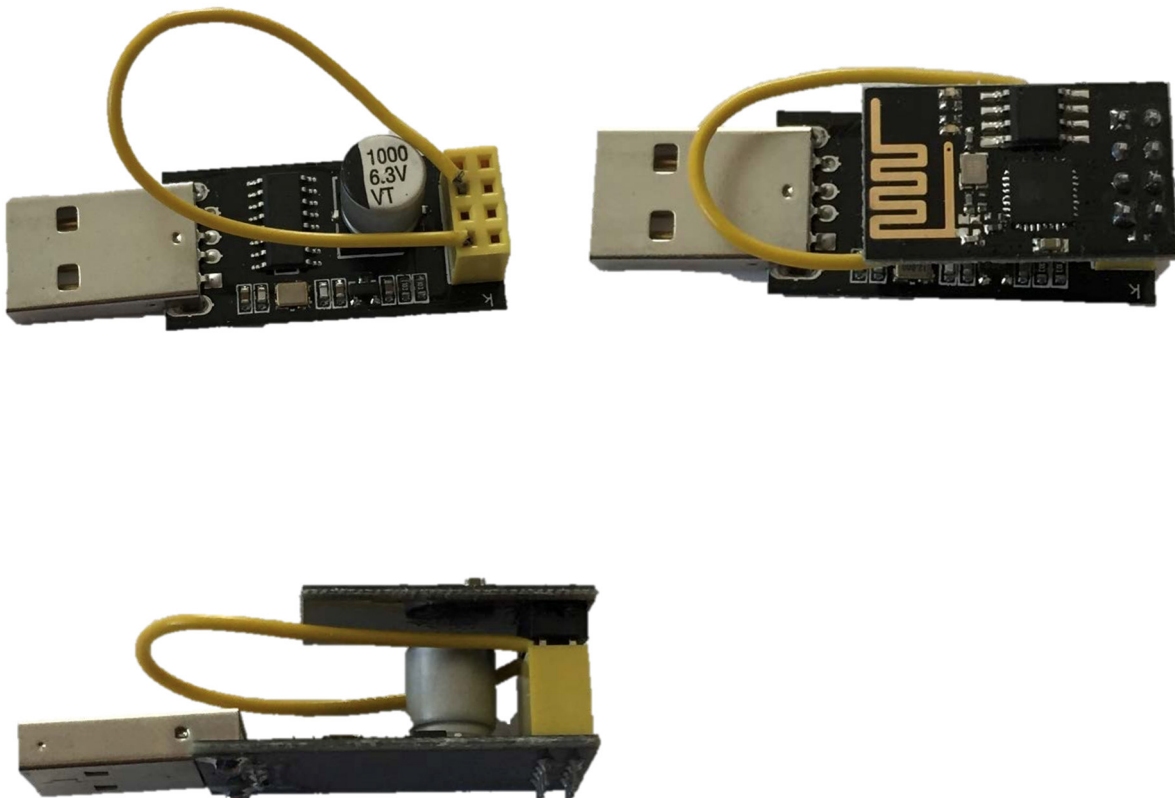
Cablaggio del modulo con l'adattatore seriale:



GPIO02 deve essere collegato a terra durante l'avvio per consentire al chip di accedere alla modalità di programmazione.

Per fare questo, devi creare un ponte tra GPIO2 e GND.

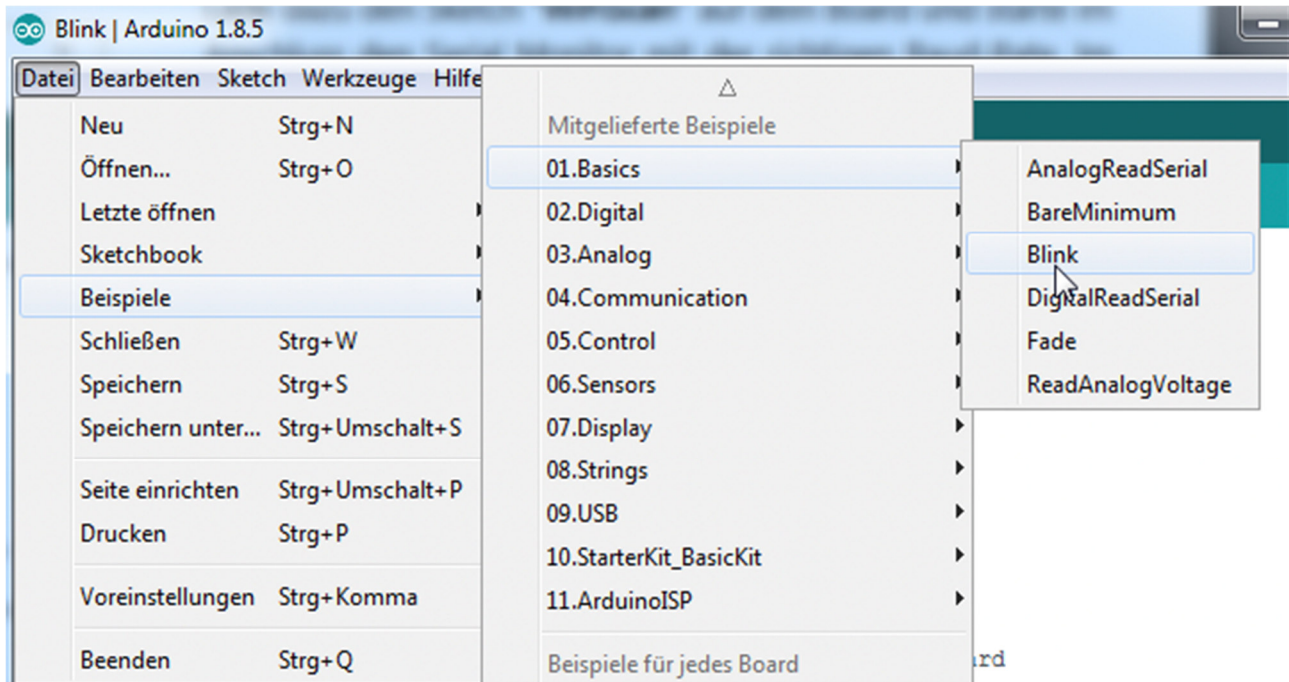
Questo ponte deve essere rimosso dopo la programmazione.



Il codice Arduino:

Ora che il cablaggio è stato eseguito e l'adattatore è stato collegato, scriviamo il nostro primo codice. Facciamo lampeggiare il LED direttamente sull'ESP8266.


Per fare ciò, vai su File> Esempi> 01.Basics> Lampeggiante.



```
void setup() {  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH);  
  delay(1000);  
  digitalWrite(LED_BUILTIN, LOW);  
  delay(1000);  
}
```


Se il LED con "LED_BUILTIN" non dovesse lampeggiare, allora devi sostituire LED_BUILTIN con la porta 1 e riprovare:

```
void setup() {  
  pinMode(1, OUTPUT);  
}  
void loop() {  
  digitalWrite(1, HIGH);  
  delay(1000);  
  digitalWrite(1, LOW);  
  delay(1000);  
}
```

Dopo aver aperto o modificato il codice, fai clic su  e verifica il programma:

Se tutto è corretto e il programma non contiene errori possiamo caricarlo su ESP8622-01.

```
Der Sketch verwendet 247055 Bytes (23%) des Programmspeicherplatzes. Das Maximum sind 1044464 Bytes.  
Globale Variablen verwenden 32868 Bytes (40%) des dynamischen Speichers, 49052 Bytes für lokale Variablen
```

Per questo facciamo clic su 

Se il nostro ESP8266-01 non fosse in modalità programmazione, mostrerebbe il seguente messaggio:

```
error: espcomm_upload_mem failed  
error: Failed to open COM11  
error: espcomm_open failed  
error: espcomm_upload_mem failed  
error: espcomm_upload_mem failed
```

In tal caso, scollegare semplicemente l'alimentazione brevemente e ricollegarla o ripristinarla e controllare se il GPIO02 è collegato a terra.

Una volta riavviato il caricamento  dovrebbe funzionare.

Dopo la programmazione, come già descritto, scollegare nuovamente le linee contrassegnate in blu chiaro!

Ora il LED lampeggia ogni secondo, ma questa è solo una piccola parte di ciò che può fare ESP8266-01.

Utilizzo di ESP8266-01 come modulo WLAN per Arduino

Nozioni di base sulla comunicazione tramite comandi AT

Il modulo WLAN esp8266 AZ-Delivery viene fornito preinstallato con un firmware che consente ai dispositivi collegati, come un Arduino, di comunicare con esso tramite i comandi AT. Il set di istruzioni AT è relativamente vecchio (circa 30 anni) ed è stato originariamente sviluppato per la comunicazione con i modem. Nonostante ciò, la sua semplicità ed economia di risorse lo rendono ideale per la comunicazione con ESP8266-01.

Tutti i comandi AT iniziano con "**AT** +" seguiti dal comando effettivo. Inoltre, è importante che tutti i comandi AT terminino con un'interruzione di riga (CR e NL). Per questo motivo, se si invia un comando sulla connessione seriale tra ESP8266 e Arduino, è necessario utilizzare sempre println per inviare l'interruzione di riga e indicare al modulo di eseguire il comando.

I comandi AT sono disponibili nella scheda tecnica di ESPRESSIF:

https://www.espressif.com/sites/default/files/documentation/4a-esp8266_at_instruction_set_en.pdf

Download di un sito Web

In questo esempio, vogliamo comunicare a ESP8266-01 AZ-Delivery i comandi AT per scaricare un sito Web. Per fare ciò, il modulo WLAN deve prima essere impostato sulla modalità "Client WLAN" tramite il comando **AT+CWMODE=1**. Successivamente può essere collegato a una rete Wi-Fi tramite **AT+CWJAP="SSID","Password"**.

Ora è possibile utilizzare **AT+CIPSTART="TCP","website.de",80** per connettersi al server di destinazione. Bisogna poi dire al modulo WiFi tramite **AT+CIPSEND=NUMBER**, quanto lunga è la richiesta che si vuole inviare al server. Poi devi aggiungere alcuni caratteri, perché le interruzioni di riga devono essere contate alla fine.

La richiesta al server si basa sul principio GET /Paddy_to_Website.html. A questo punto potrebbe essere necessario inserire spazi finché non verrà raggiunto il numero di caratteri specificato in precedenza, quindi ESP8266 invia automaticamente la richiesta al server e invia automaticamente la risposta (codice HTML del sito Web) a Arduino. Finché non verrà raggiunto il numero di caratteri specificato, ESP8266 non farà nulla. Se la query è più lunga di quanto specificato, verrà visualizzato "busy".

A causa della complessità di questa procedura, si dovrebbe prima provare su PC tramite la console seriale e poi tradurre la procedura in codice Arduino, altrimenti la risoluzione dei problemi sarà molto difficile.

Ce l'hai fatta, ora puoi programmare e realizzare i tuoi progetti con ESP8266-01!

È arrivato il momento di sperimentare.

E per più hardware, ovviamente, visita il nostro negozio online su:

<https://az-delivery.de>

Buon divertimento!

Impressum

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