Pin	Name	Description
1	GND	Ground
2	U0TXD	UART0 Transmit
3	GPIO2	Has internal pull-up
4	CHIP_EN	Chip Enable, active high
5	GPIO0	Has internal pull-up
6	EXT_RSTB	External reset signal, active low, has no pull-up? Spurious blue LED activity when attaching a DMM between GND and RST to check voltage.
7	U0RXD	UART0 Receive, has internal pull-up
8	VDD	+3.3V power input

I comandi AT

Giusto per accennare ai comandi AT parto dalla definizione presa su wikipedia:

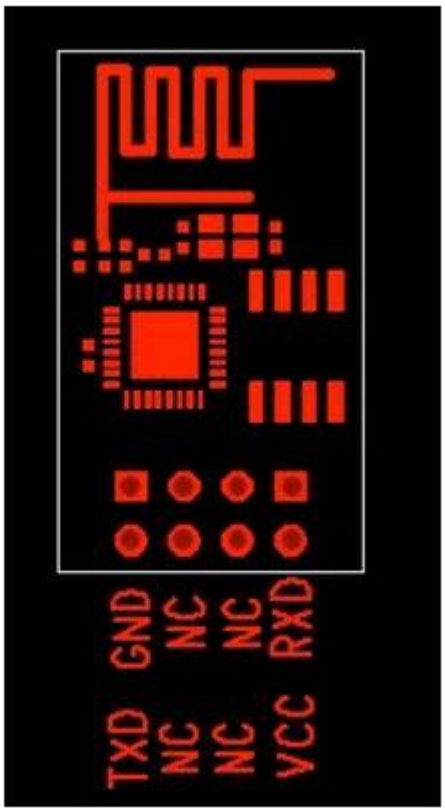
*“La maggior parte dei modem fonici utilizza i **comandi AT Hayes**, uno specifico insieme di comandi originalmente sviluppato per il modem Hayes Smartmodem da 300 baud.*

La stringa di inizializzazione consiste di una serie di comandi che prepara il modem per la comunicazione, impostando caratteristiche come il tipo di connessione, i tempi di attesa, la rilevazione delsegnale di occupato eccetera. Nei modelli di modem più recenti l’inizializzazione viene gestita da programmi con una qualche interfaccia grafica, in modo che l’utente non debba digitare questi comandi e neanche conoscerne l’esistenza.”

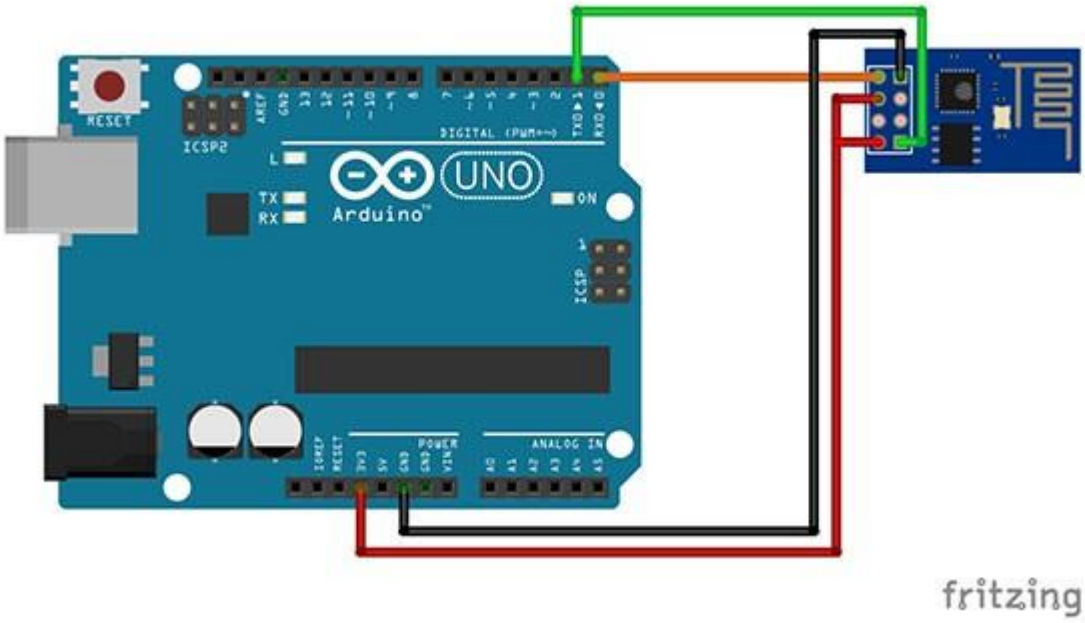
Per il mio test ho utilizzato alcuni dei comandi AT disponibili e supportati dall’ESP8266 prendendo come riferimento i comandi riportati in uno dei link suggeriti al paragrafo precedente.

Colleghiamo l’ESP 8266 ad arduino

il collegamento del modulo è molto semplice



io ho realizzato uno schema di connessione diretta ai pin 0 ed 1 di arduino in modo da utilizzare quest'ultimo come semplice passacarte verso l'ESP8266:



Upload sketch to arduino

lo sketch che ho utilizzato è il più semplice di tutti, lo sketch vuoto:

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

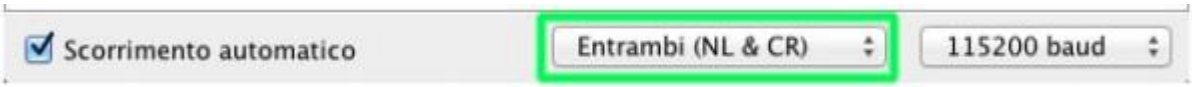
la sua funzione è semplicemente permettere ad arduino di non far nulla e non interferire con la comunicazione seriale che transita direttamente al modulo.

Il primo test dell'ESP8266

Per il primo test devi utilizzare, come detto, i comandi AT in modo da comprendere quali operazioni potrai fare con il tuo ESP8266 direttamente connesso ad Arduino.

Inizia con aprire il monitor seriale dell'IDE, dopo aver collegato arduino al computer ed al modulo ESP8266 come descritto sopra.

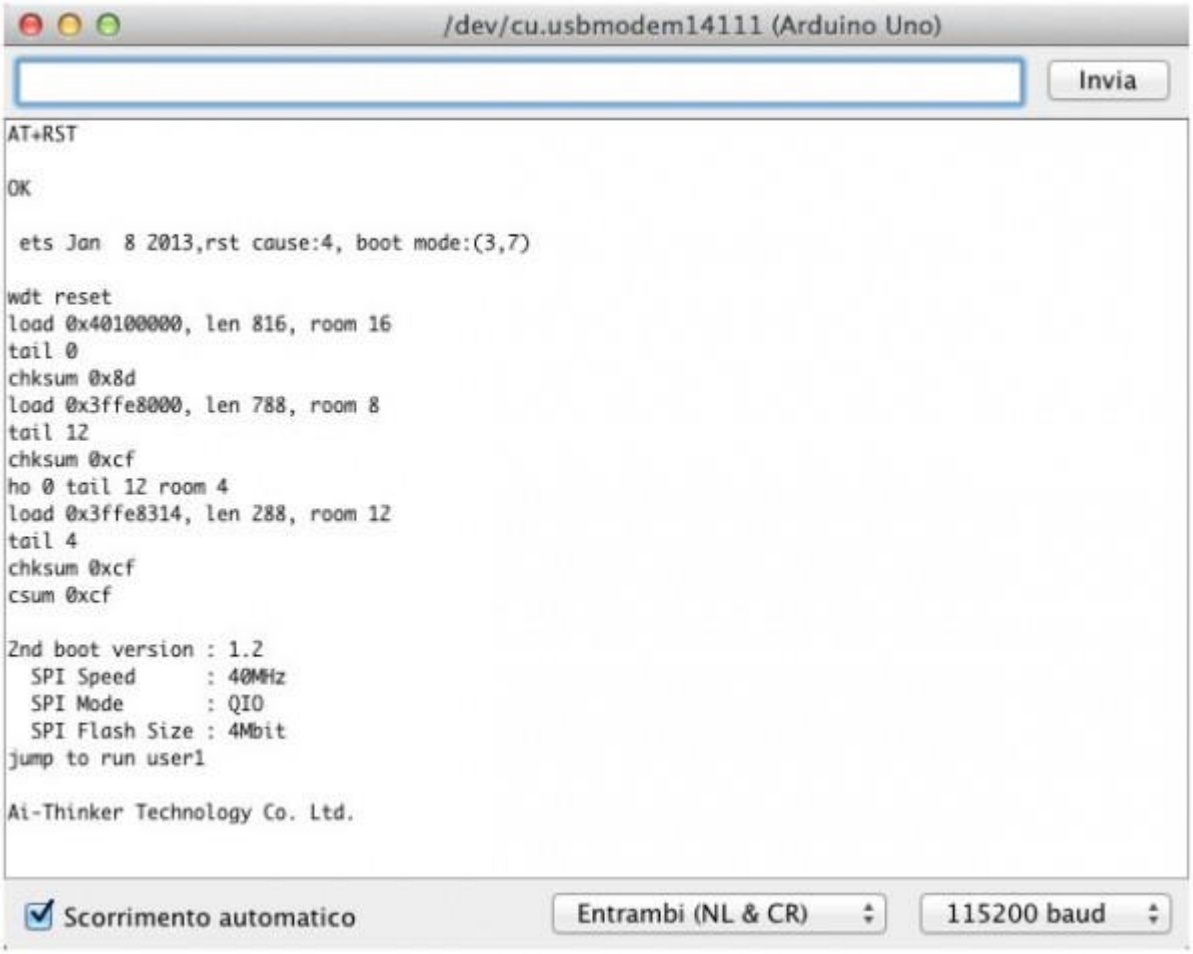
Il monitor seriale devi configurarlo in modo che ad ogni invio di stringa seriale aggiunga al termine sia il carattere di fine riga sia quello di ritorno a capo, come in figura:



inoltre è necessario che la velocità di comunicazione (baud) sia la medesima tra il modulo ed il monitor seriale.

Le velocità possibili per questo modulo sono 3: 9600, 57600 e 115200, ancora non mi è chiarissimo come distinguere quale velocità vada impostata per ciascun modello, posso dirti che dopo 2 prove la terza è quella che mi ha dato esito positivo e quindi ho impostato la comunicazione a 115200:

trovata la giusta impostazione di comunicazione, in termini di baud, puoi inviare un reset al modulo con il comando AT+RST:



a cui seguono una serie di informazioni restituite direttamente sul monitor seriale.

Puoi interrogare il modulo per farti restituire alcune informazioni quali la versione del firmware (AT+GMR):

```
AT+GMR
AT version:0.21.0.0
SDK version:0.9.5

OK
```

e successivamente ho chiesto al modulo di impostare la modalità operativa:

- 1 Sta (Station – CLient)
- 2 AP (Access Point)
- 3 Both (Entrambe)

usando il comando AT+CWMODE?, che permette sia di eseguire richieste di questo tipo sia di impostare la modalità desiderata, ho richiesto di visualizzarmi la modalità impostata:

```
AT+CWMODE=1

OK
AT+CWMODE?
+CWMODE:1

OK
```

ora sia che la modallità in cui sta operando è Client (1) puoi passare a verificare quali reti WiFi ci sono nei paraggi e sono captate dal modulo, per farlo usi il comando AT+CWLAP:

```
AT+CWLAP
+CWLAP:(3,"Almanet",-57,"d4:ca:6d:67:96:bf",1)
+CWLAP:(4,"Alice-38125071",-52,"00:23:8e:4c:2a:c8",7)

OK
```

e scopri sia le reti disponibili sia i Mac Address delle interfacce che espongono le reti wifi.

Ora puoi chiedere al tuo modulo di eseguire la Join alla rete wifi:

```
AT+CWJAP="Alice-38125071","[illegible]"

OK
AT+CIFSR
+CIFSR:STAIP,"192.168.2.238"
+CIFSR:STAMAC,"18:fe:34:a3:01:fa"
```

con il comando AT+CWJAP che accetta come primo parametro l'SSID e come secondo parametro la password di accesso.

Infine per verificare che la join sia avvenuta correttamente ho chiesto al modulo di restituirmi l'Ip ricevuto dal router con il comando AT+CIFSR e poi ho confrontato il risultato, che vedi nella schermata precedente, con quanto presente sul router:

802.11g	**54 Mbps**	**Eccellente**	**18:fe:34:a3:01:fa**	**192.168.2.238**	**Connesso**

Fonte: http://www.mauroalfieri.it