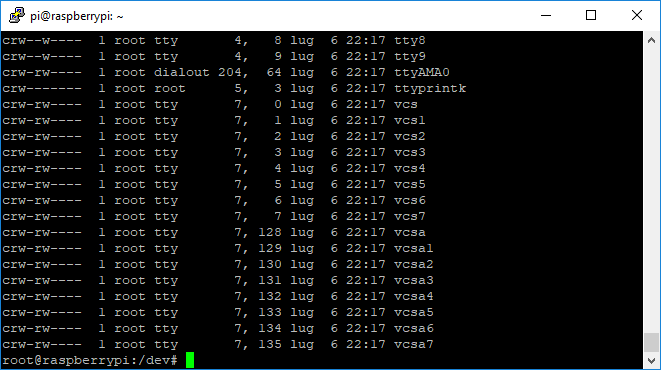
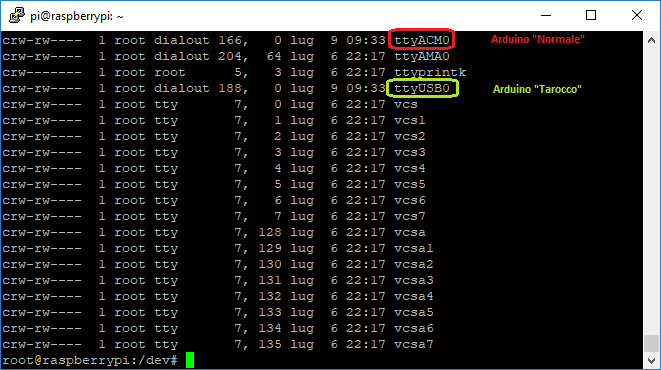
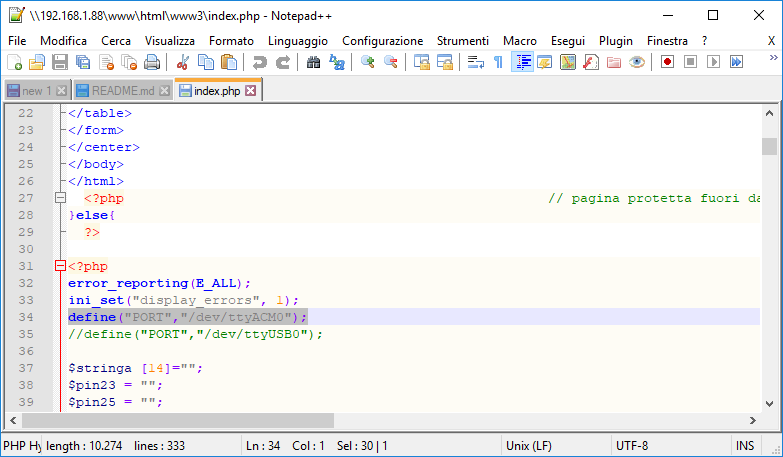
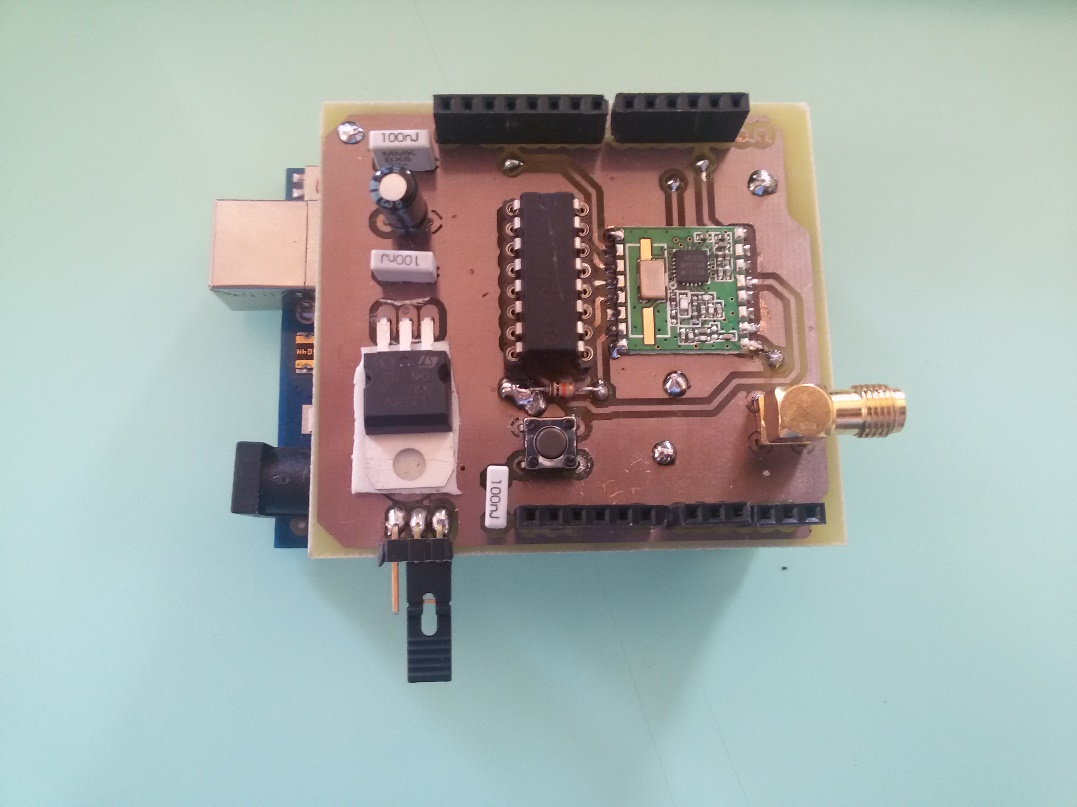
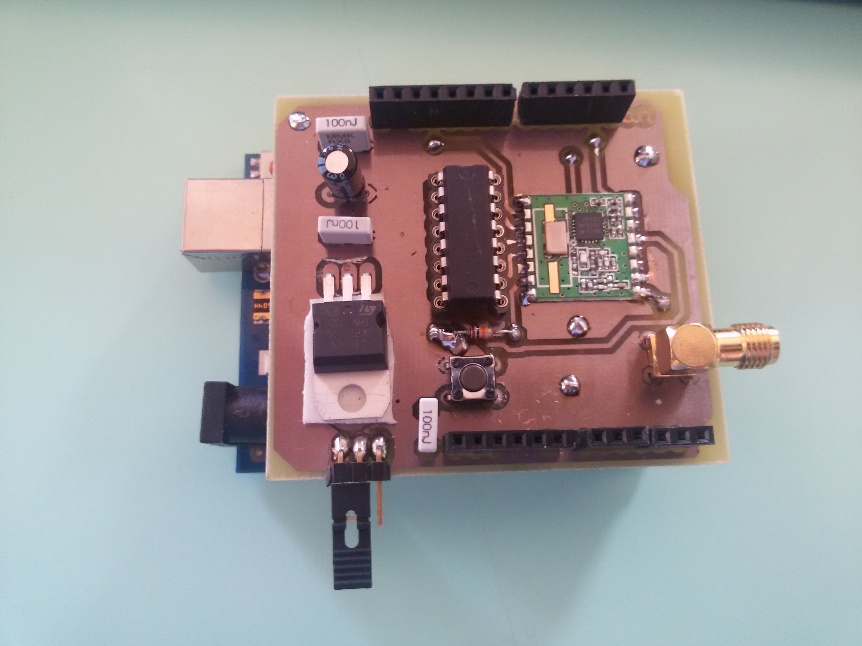
GUIDA AL SETUP DEL PROGETTO

CONFIGURAZIONE DELL’HOST (nel mio caso un RaspberryPi)

1. Installare un server web che supporti il linguaggio php (nel mio caso apache2)
2. Assicurarsi di essere in ambiente linux e di disporre di un account con privilegi amministrativi
3. Usa il comando “cd /dev” per spostarti nella cartella contenente tutti i dispositivi di I/O  
   
4. Visualizza solo quelli che ci interessano impartendo il comando “ls -al | grep tty”  
   
5. Collega il tuo arduino all’host tramite USB
6. Ri-esegui il comando; noterai che si è aggiunto un dispositivo, solitamente chiamato ttyACM0 se l’arduino ha il classico controller USB ATmega16U2; vedrai qualcosa di simile a ttyUSB0 in caso di un clone con chip USB CH340G o similari:  
     
   Consiglio di fare il login all’interfaccia grafica, se disponibile, e installare l’ide di arduino per evitare problematiche future
7. Eseguire “sudo adduser www-data dialout” per aggiungere il daemon di apache al gruppo dialout e nel caso l’output del comando “ls -al | grep tty”  
   non fornisse come output root e dialout sulla porta interessata, come evidenziato nell’immagine sottostante,  
     
   procedere cambiando l’owner della porta con il comando “chown root:dialout ttyACM0”
8. Cambiare la porta seriale in base a quella utilizzata nel file index.php
9. Copiare il contenuto della cartella libraries in C:\Users\%username%\Documents\Arduino
10. Caricare gli sketch su arduino uno e mega2560
11. Per la shield servono i seguenti componenti per ogni arduino:
    1. Un modulo RF22B o RFM23B (il 22B ha più potenza in uscita)
    2. Un level shifter, in questo caso il CD74HC4050EE4
    3. Un pulsante da PCB (opzioanle)
    4. Una resistenza da 10kΩ
    5. 3 condensatori in poliestire da 100nF; 1 elettrolitico da 10uF
    6. Un Regolatore di tensione, in questo caso l’LD1117V33C
    7. Un connettore SMA

ATTENZIONE: La shield funziona perfettamente su arduino uno ma quando la si usa sul Mega, bisogna impostare i pin 11, 12 e 13 come INPUT per evitare cortocircuiti e ponticellare i pin come di seguito:   
11 ---> 51, 12 ---> 50, 13 ---> 52  
In questa maniera, si preserva il funzionamento dell’SPI, perdendo i pin 11, 12 e 13; lo slave select rimane invariato.  
Inoltre la shield collegata all’arduino collegato al raspberry, deve avere il jumper tra il condensatore a fianco e massa; questo serve ad inibire il reset dell’arduino all’apertura della comunicazione seriale:



**POSIZIONE**

**NULLA**

**RESET**

**INIBITO**

1. Il tutto dovrebbe funzionare, in caso contrario potete creare una segnalazione qui su GitHub perché possa essere utile anche ad altri; altrimenti potete entrare nel mio [***gruppo telegram***](https://t.me/joinchat/Dzkdyw6bsxirKAu0QJYPQg)