



Jsonrpc: la richiesta

Tutti i parametri trasferiti di ogni tipo sono singoli oggetti, serializzati usando JSON. Una richiesta è una chiamata a uno specifico metodo disponibile sul sistema remoto; deve contenere tre specifiche proprietà:

- **method** - Una stringa col nome del metodo da invocare.
- **Params** - Un array di oggetti come parametri al metodo invocato.
- **id** - Un valore di qualsiasi tipo, usato per riferire la risposta alla richiesta a cui si sta rispondendo.



Jsonrpc: la risposta

Il server che riceve la richiesta deve rispondere con una risposta valida a tutte le richieste ricevute. Una risposta deve contenere le proprietà descritte qui sotto:

- result - I dati ritornati dal metodo invocato. Se c'è un errore invocando il metodo, il valore deve essere null.
- error - Uno specifico codice di errore se l'invocazione del metodo ha dato luogo a un errore, altrimenti null.
- id - L'id della richiesta a cui si sta rispondendo.



Json-rpc un modo per fare tutto...

Questo un esempio di interrogazione e risposta di un sensore di temperatura

- SEND: {"jsonrpc":"2.0", "method":"getjson", "params":{"node":1, "type":"TMP", "driver":"I2C", "address":72},"id": 0}
- RECEIVE: {"jsonrpc":"2.0","result":{"B12101":30633},"id":0}

B12101 indica che il numero che segue è una temperatura in centesimi di gradi Kelvin, quindi 33.18 C.

Le remote procedure disponibili sono documentate sul wiki del progetto:
http://www.raspibo.org/wiki/index.php/Gruppo_Meteo/RemoteProcedures



MQTT (Message Queue Telemetry Transport)

- è un protocollo publish/subscribe particolarmente leggero, adatto per la comunicazione M2M attraverso un tramite detto broker
- Il mittente di un messaggio si limita a "pubblicare" il proprio messaggio al broker. I destinatari si rivolgono a loro volta al broker "abbonandosi" alla ricezione di messaggi.
- Il meccanismo di sottoscrizione consente ai subscriber di precisare a quali messaggi sono interessati tramite un pattern (topic).
- Client e broker si scambiano messaggi di polling per monitorare lo stato delle comunicazioni; sono previsti messaggi "will and testament"



Caratteristiche moduli GSM, Master, Satellite e Bluetooth

- Firmware unico configurabile al tempo della compilazione
- Parametri di configurazione run time salvati su EEPROM
- Debug on serial insieme a json-rpc
- WatchDog con timeout di 8 sec
- Modalità sleep con interrupt sulla radio per modulo satellite
- RTC sincronizzato tramite NTP / GPS / http



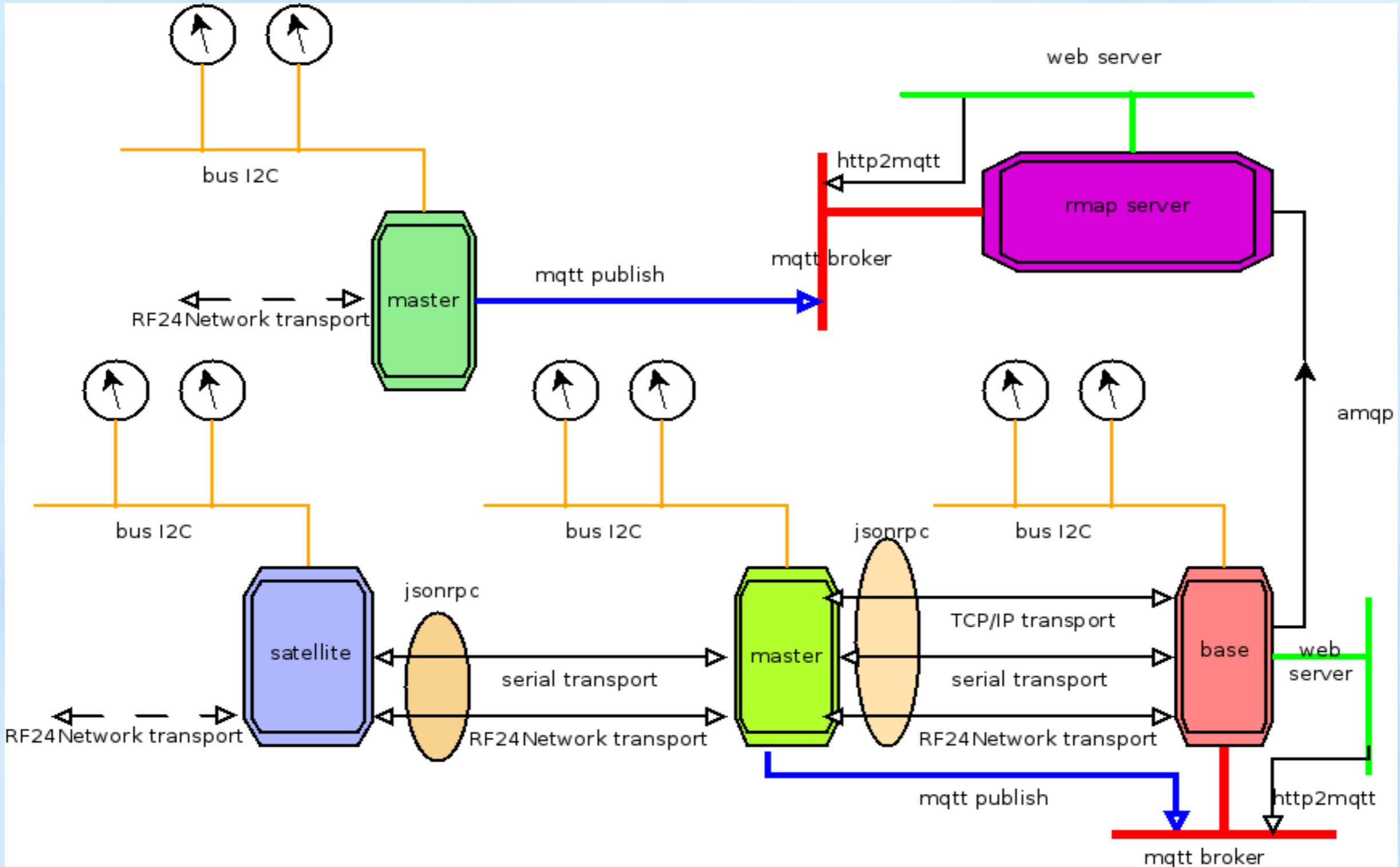
Gestione dei sensori

Libreria di "driver" per sensori

- Esistono attualmente due versioni, una in C++ e una in python
- Porta la gestione della sensoristica ad un livello di astrazione più alto. Aggiungere un nuovo tipo di sensore consiste nell'estendere una classe con quattro metodi per effettuare la lettura di quello specifico sensore:
 - int **setup**(int address);effettua eventuali settaggi necessari al funzionamento del sensore; esempio per temperatura: numero di bit di risoluzione, operazione di misura one-shot
 - int **prepare**(unsigned long* waittime); impartisce al sensore il comando per effettuare una singola misurazione torna il tempo in millisecondi di attesa necessario
 - int **get**(int* value); torna i valori della misurazione
 - jsonObject* **getJSON**() = 0; torna i valori in formato json



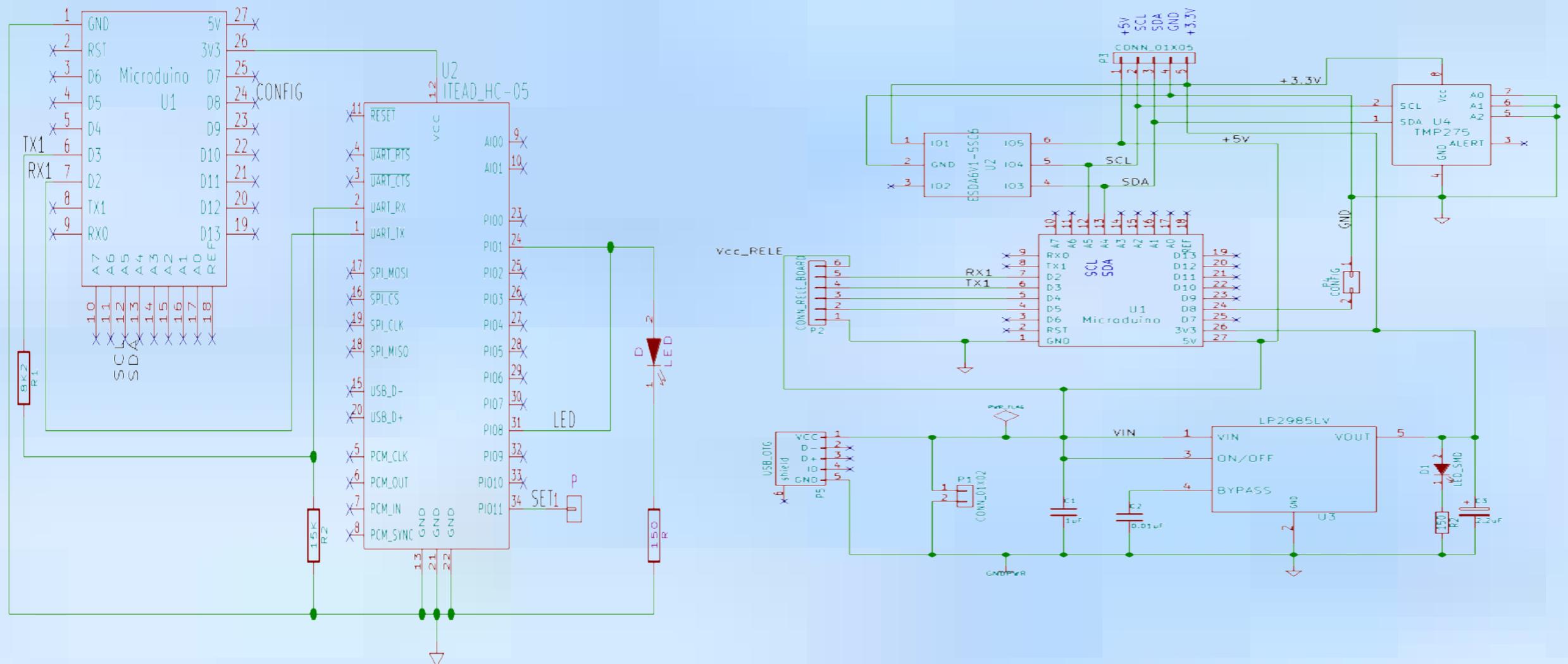
Schema comunicazione tra moduli





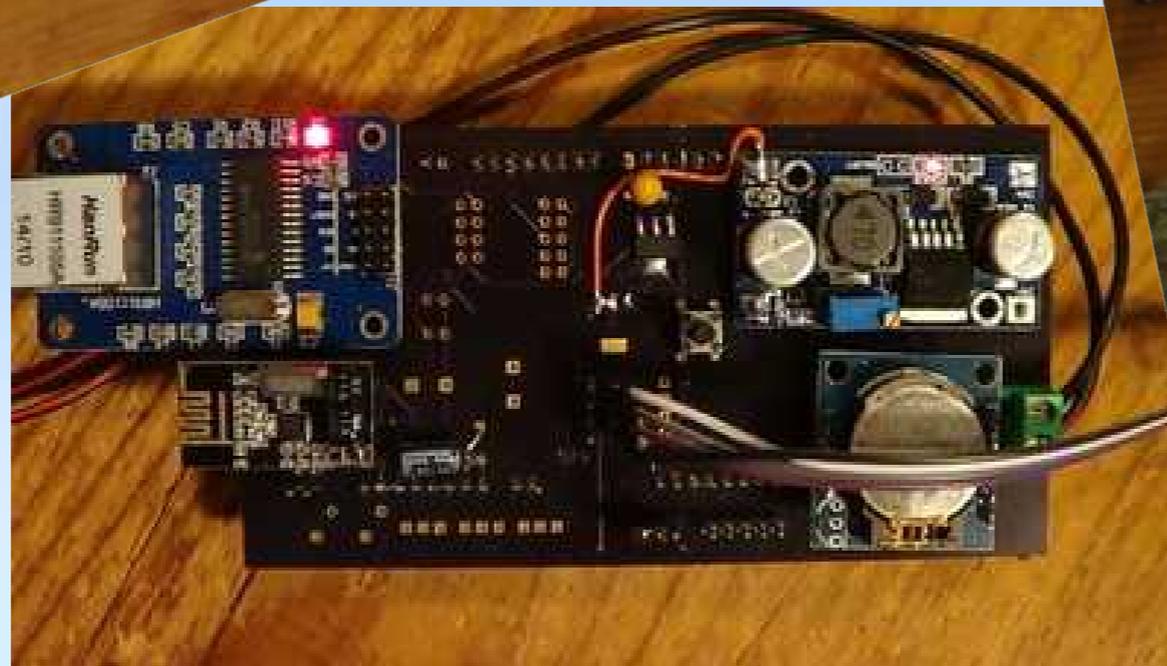
KiCad

- KiCad è una suite open source di software Electronic Design Automation (EDA) per il disegno di schemi elettrici e circuiti stampati (PCB). Ha un ambiente di sviluppo integrato (IDE) con editor di schematici, generazione della distinta base, sbroglio circuitale del PCB e visualizzatore di file Gerber.





Prototipi versione 0 e 1





Board con microcontroller versione 1 circa 50€ /100€

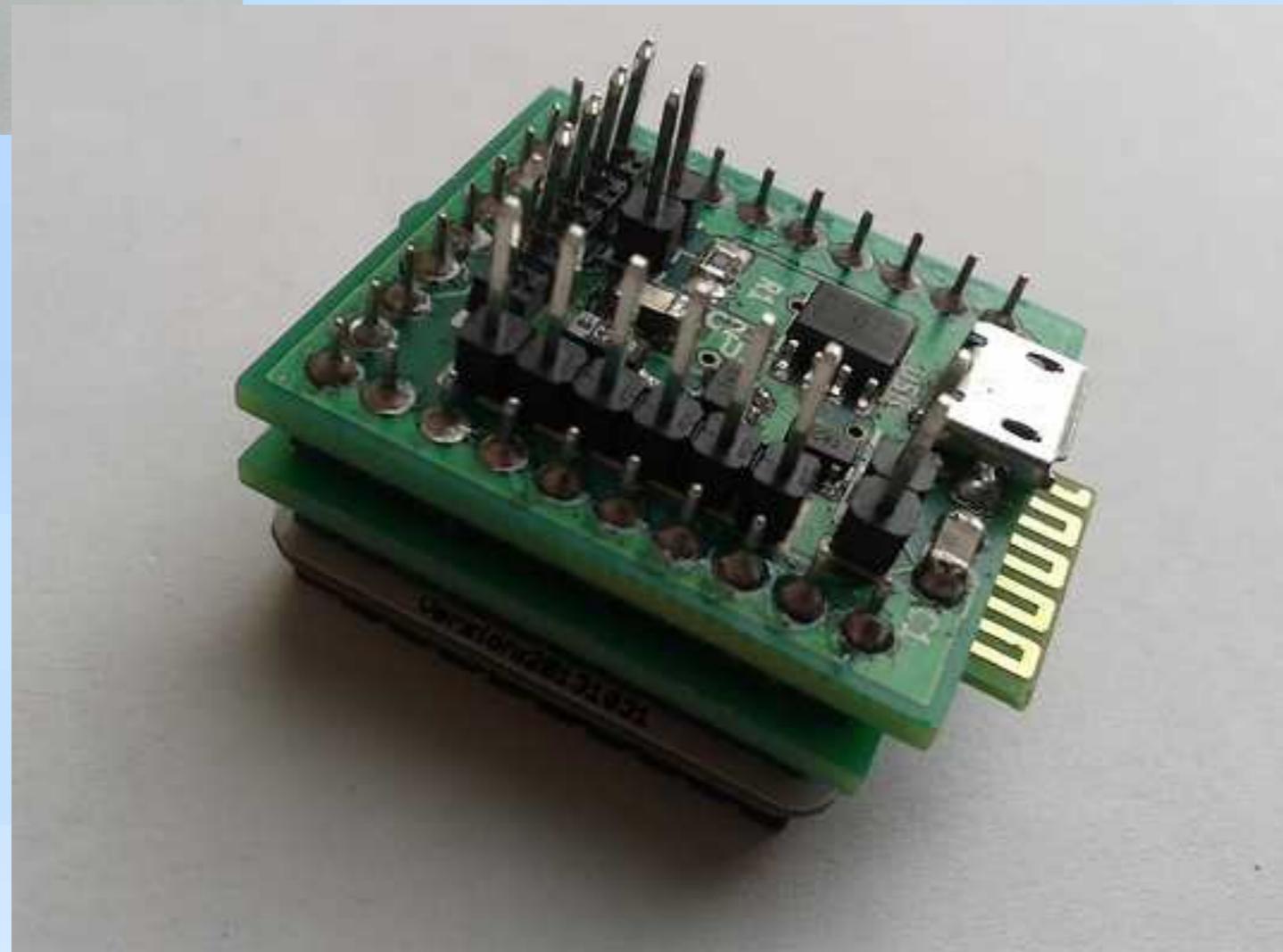
- Modulo master
 - Microduino core+ 1284p / arduino mega 2560
 - Breadborad/circuito stampato
 - scheda RTC
 - scheda mini ethernet ENC60 /microduino ENC / scheda SIM900
 - scheda radio RF24
 - cavo power over ethernet
 - stabilizzatore dc/dc switched
 - sensori
- Modulo satellite Modulo Bluetooth
 - microduino core+ 644p @5V 16MHz
 - microduino RF24 con antenna / scheda Bluetooth
 - microduino seriale USB FT232R
 - sensori



Modulo Stima-Bluetooth

Il modulo è composto da tre board:

- board microduino core+
- board stima-I2C
- board stima-bluetooth



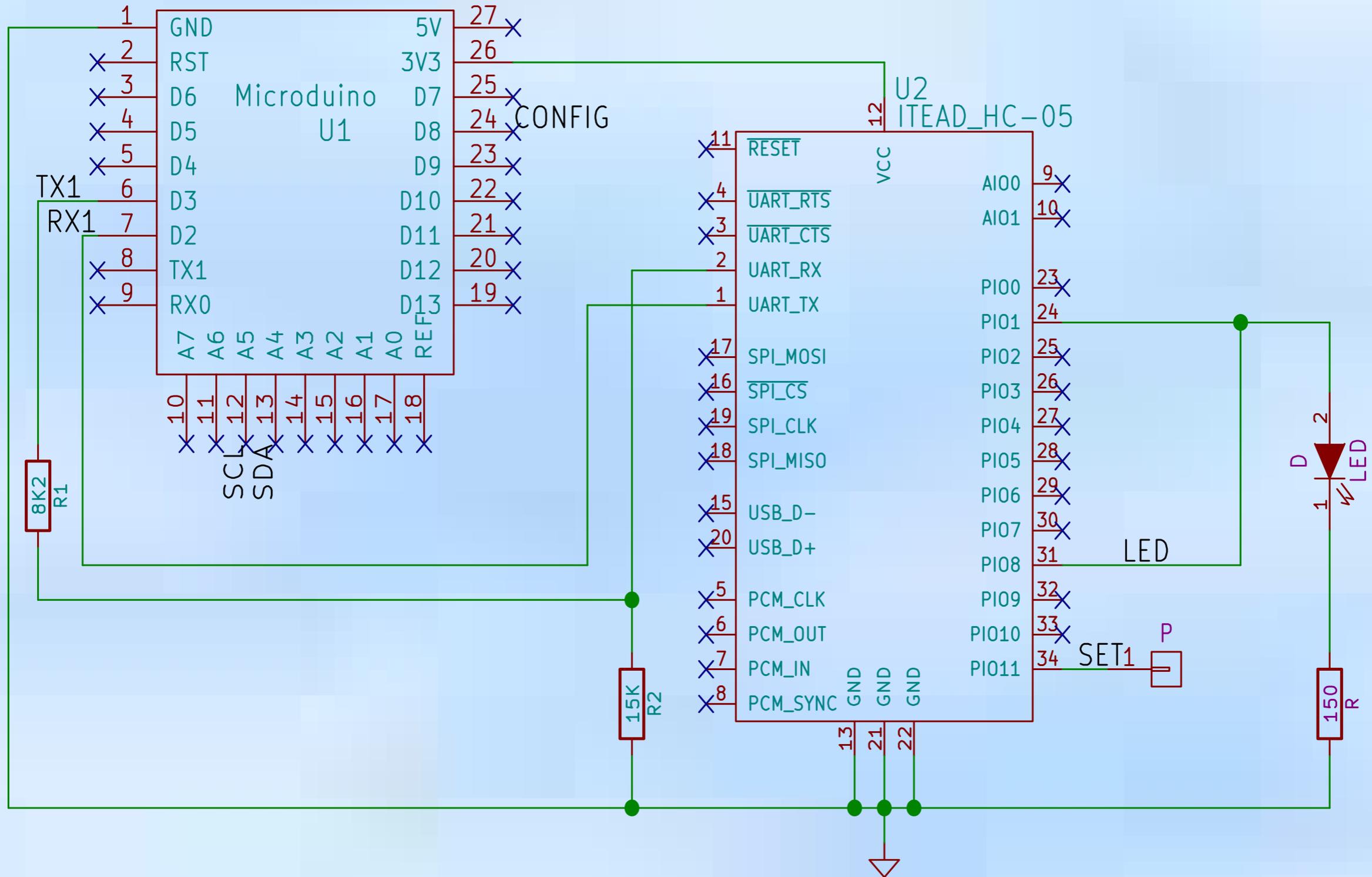
Bologna, 2017-07-20

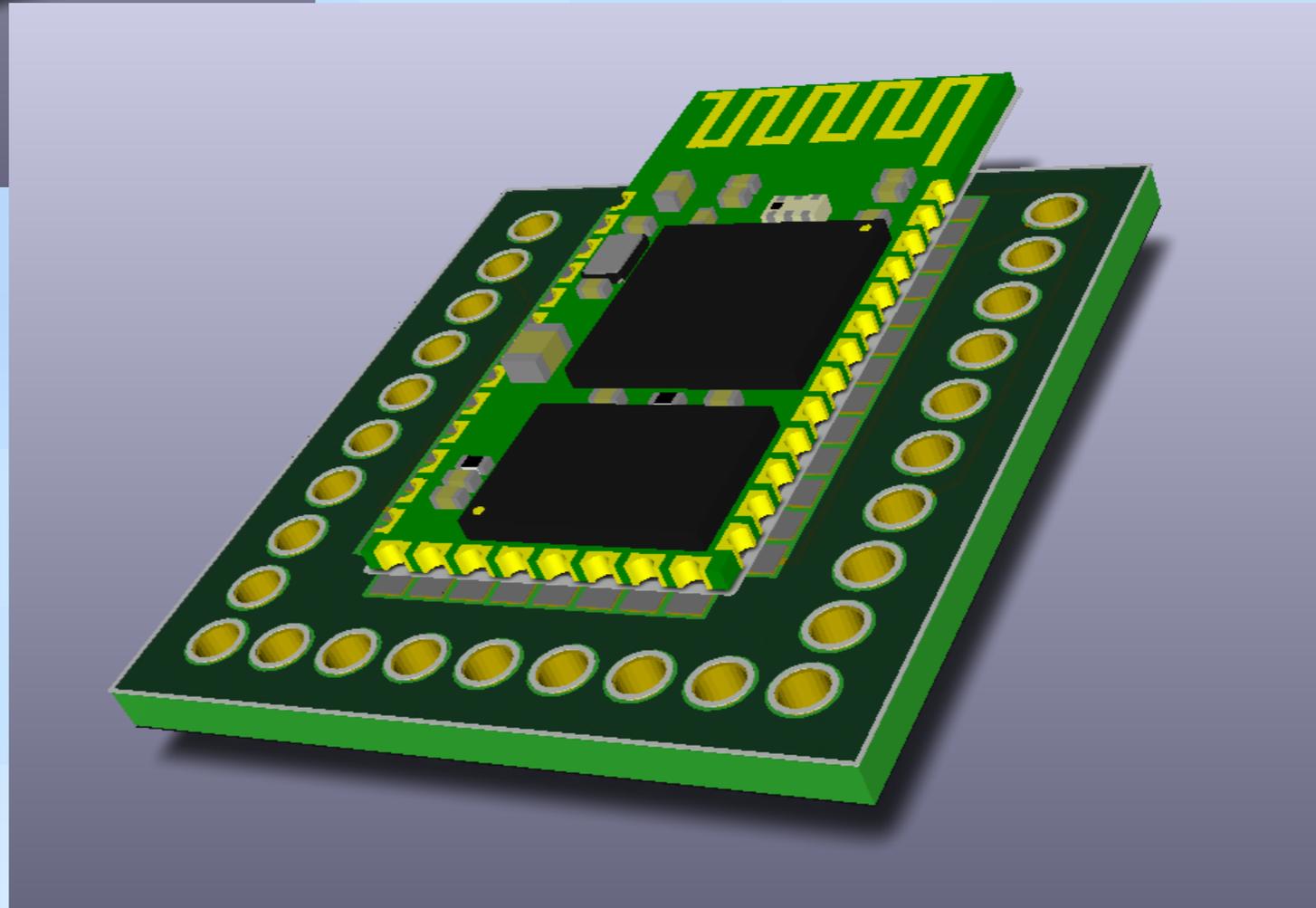
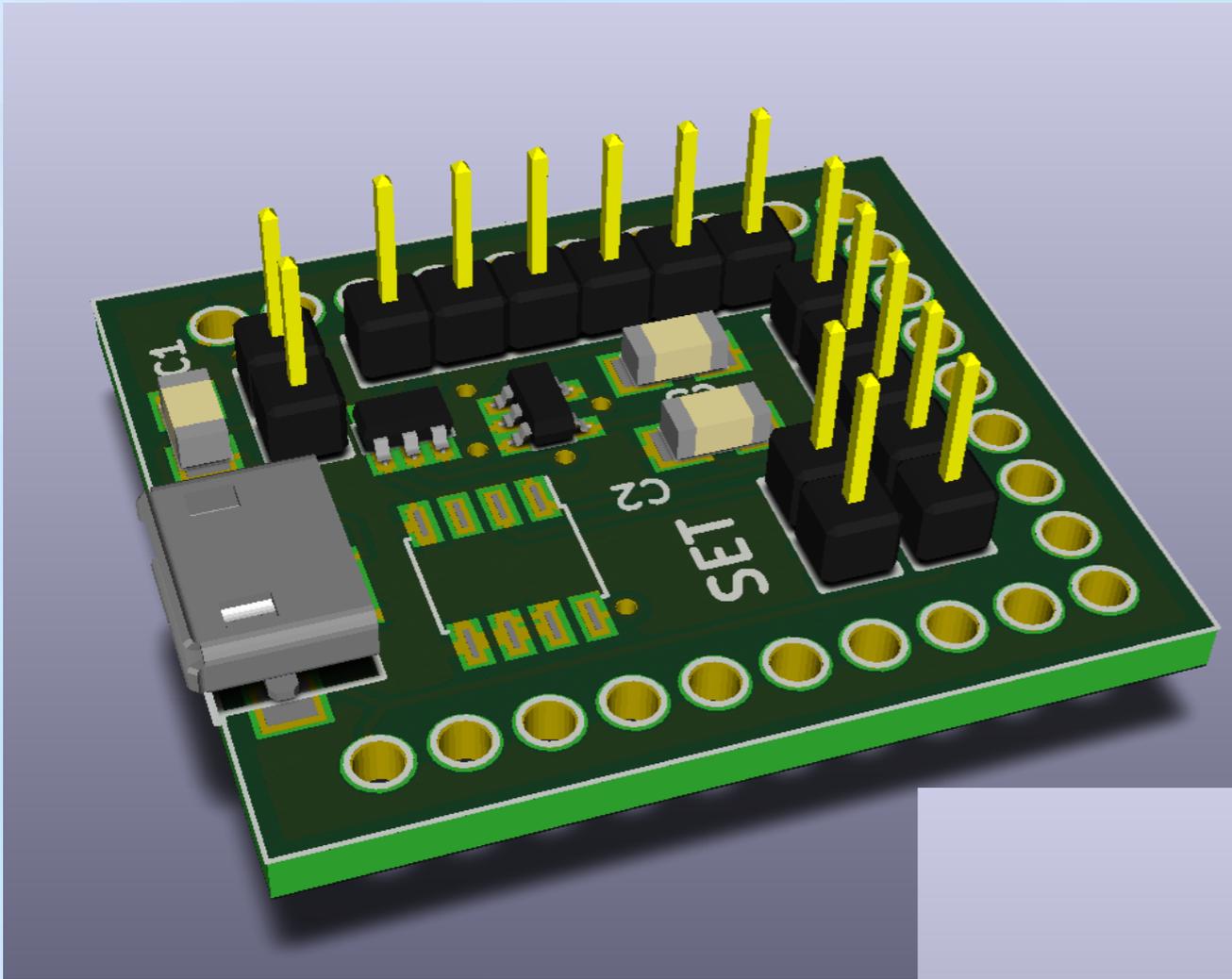
Paolo Patruno, Rete di Monitoraggio Ambientale Partecipativo

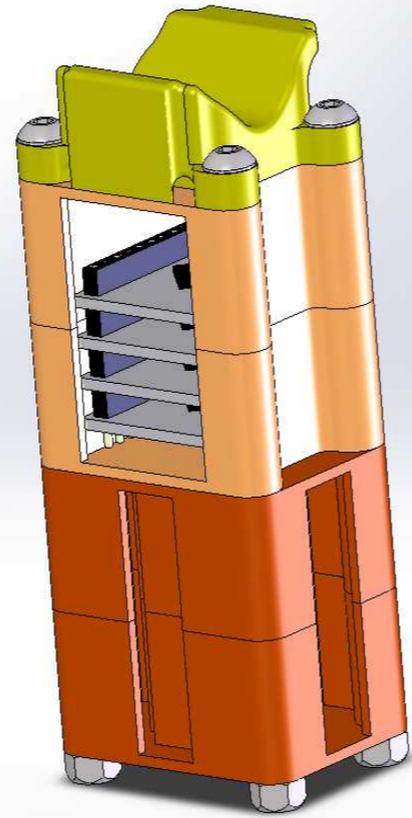
ARPAE-SIMC progetto RMAP – [HTTP://rmap.cc](http://rmap.cc)



Board Stima-bluetooth







Bologna, 2017-07-20

Paolo Patruno, Rete di Monitoraggio Ambientale Partecipativo

ARPAE-SIMC progetto RMAP - [HTTP://rmap.cc](http://rmap.cc)



Modulo Stima-Master

- Board microduino core+ 1284
il componente principale e' un Atmega1284P a 16Mz ha 128K di memoria Flash, e 16K di SRAM
- Board microduino ENC
connettore RJ45, stabilizzatore switching per il PoE e l'integrato ENC28J60
- Board STIMA-I2C
- Board microduino nRF24 (opzionale)



Modulo Stima-Satellite

Il modulo satellite è un modulo "passivo", rimane in attesa di richieste via radio da un modulo Stima-Master o STIMA-GSM/GPRS, interroga i sensori i2c e ne trasmette le misurazioni al modulo Master. E' utilizzabile solo se sul modulo Master o GSM/GPRS avete montato la board microduino nRF24. E' composto dalle seguenti schede:

- Board microduino core+ 644
- Board microduino nRF24
- Board microduino nRF24



Modulo Stima-GSM/GPRS

Questo modulo utilizza la rete di telefonia mobile. Vista la minore stabilità di questa rete di comunicazione il modulo è in grado di salvare permanentemente i dati per un lungo periodo su una SD.

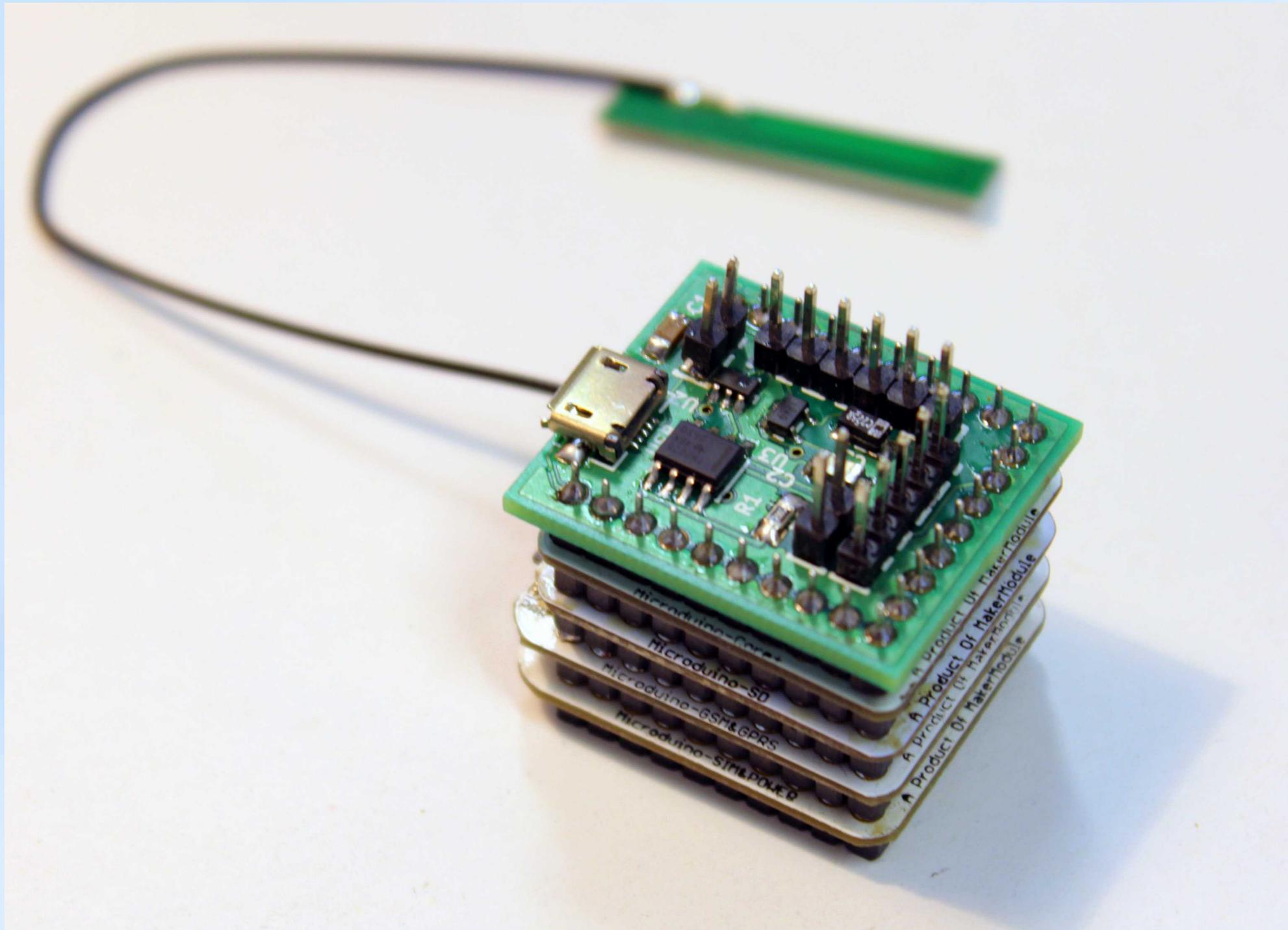
- **Board microduino GPRS/GSM**

Basato su sim800, un modulo quad-band GSM/GPRS, lavora con le frequenze GSM850MHz, EGSM900MHz, DCS1800MHz and PCS1900MHz. SIM800 GPRS multi-slot class 12/ class 10 e supporta gli schemi di codifica GPRS CS-1, CS-2, CS-3 e CS-4. GPRS uplink/downlink transfer rate massimi di 85.6 kbps

- **Board microduino core+ 1284**
- **Board microduino nRF24 (opzionale)**
- **Board microduino SD**

Il modulo memorizza i dati su microsd. Quando la connessione al server viene ristabilita, i dati vengono inviati a destinazione

- **Board STIMA-I2C**



Bologna, 2017-07-20

ARPAE-SIMC progetto RMAP – [HTTP://rmap.cc](http://rmap.cc)

Paolo Patruno, Rete di Monitoraggio Ambientale Partecipativo