

STIMA: MISURAZIONE DI PARAMETRI METEO E AMBIENTALI



Tre nuovi moduli per l'acquisizione e trasmissione al server RMAP dei parametri rilevati, utilizzabili anche per applicazioni domotiche come chiudere una tenda da sole in caso di pioggia.

di PAOLO PATRUNO
e DANIELE CASTELLARI

Dopo avervi spiegato che cos'è il sistema modulare STIMA e come esso si integra nel progetto RMAP (Rete di Monitoraggio Ambientale Partecipativo), in queste pagine presentiamo ulteriori strumenti per acquisire parametri dall'ambiente che ci circonda; abbiamo sviluppato tre moduli con caratteristiche differenti che si adattano a situazioni di installazione e di utilizzo eterogenee per tipo di alimentazione e tipologia di comunicazione dei dati. I moduli presentati in questa puntata sono tre e vanno ad aggiungersi allo STIMA-Bluetooth presentato nella scorsa puntata:



L'OCCORRENTE PER L'APPLICAZIONE

- **Microduino core+ 1284p 5V**
- **Microduino core+ 644p 5V**
- **Microduino-ENC (RJ45 POE Included)**
- **Microduino ft232R**
- **Microduino-GPRS/GSM**
- **Microduino SDcard**
- **Microduino-nRF24**
- **board Stima Bluetooth**
- **board Stima I2C**
- **SET CAVI POE**
- **Display LCD 20x4 con interfaccia I2C**
- **MODULO 4 RELÈ 5VDC 10A**
- **MODULO 2 RELÈ 5 VDC 10A**
- **Modulo 1 relè 5 Vdc 10A**

Stima-Master da collegare all'Ethernet, Stima-Satellite collegabile via radio al modulo Master, Stima-GSM/GPRS che invia i dati su GSM e salva i dati su SD-Card. Il software che qui presentiamo per la pubblicazione e visualizzazione dei dati è utilizzabile su differenti sistemi operativi: la geolocalizzazione, autenticazione e pubblicazione dei dati rilevati manualmente o a vista è stata testata su Linux, Windows, OS X e Android. In parte, l'utilizzo è stato spiegato nella scorsa puntata; in queste pagine utilizzeremo l'applicazione grafica per la configurazione dei moduli STIMA, compresa la geolocalizzazione, che è una parte fondamentale. Le configurazioni hardware sono molteplici e utilizzano differenti board; sono compatibili con i moduli hardware maggiormente diffusi e conosciuti dai maker e generalmente a basso costo. Qui utilizzeremo la configurazione "impilabile" delle board Microduino. L'unico firmware sviluppato si adatta all'hardware tramite il preprocessore C; molte opzioni quindi sono definite a tempo di compilazione, mentre altre durante l'esecuzione e salvate sulla EEPROM per poi essere rilette all'avvio.

MQTT

I moduli STIMA che qui esaminiamo utilizzano MQTT per pubblicare sul broker rmap.cc i propri dati e metadati. MQTT (acronimo di Message Queue Telemetry Transport) è un protocollo publish/subscribe particolarmente leggero, adatto per la comunicazione M2M tra dispositivi con poca memoria o potenza di calcolo e server (o message broker). HTTP è il protocollo più diffuso, ma dal punto di vista dei device che devono trasmettere dei dati, può invece non essere la scelta migliore: nell'ambito dell'Internet of Things si stanno affermando altri protocolli, soprattutto "binari", che non solo hanno un pattern diverso noto come publish/subscribe, ma soprattutto garantiscono un trasferimento minimo di byte "inutili" rispetto al contenuto informativo. Tra i più noti c'è, appunto, MQTT, nato proprio per la "telemetria", molto leggero ed affidabile (garantisce tre livelli differenti di QoS) soprattutto su reti non perfette in termini di stabilità della connessione. Nello schema publish/subscribe, mittenti e destinatari di messaggi dialogano attraverso un intermediario detto broker: il mittente di un messaggio si limita a "pubblicare" il proprio messaggio sul broker; i destinatari si rivolgono a loro volta al broker "abbonandosi" alla ricezione di messaggi. Quindi il broker inoltra ogni messaggio inviato da un publisher a tutti i subscriber interessati a quel messaggio. Il meccanismo di sottoscrizione consente ai subscriber di precisare a quali messaggi sono interessati tramite un pattern (topic). Pub/Sub disaccoppia un client che sta inviando un particolare messaggio da un altro client (o molti clients), che sta ricevendo il messaggio. Questo significa che

publisher e subscriber non conoscono l'uno l'esistenza dell'altro. Client e broker si scambiano messaggi di polling per monitorare lo stato delle comunicazioni; se il broker non riceve una PINGREQ o qualsiasi altro pacchetto da uno specifico client, chiuderà la connessione e invierà l'ultimo messaggio "will and testament" (se il client ne ha specificato uno). Il meccanismo di keep alive può essere disattivato.

IL FIRMWARE STIMA

Presentiamo ora le caratteristiche generali del firmware utilizzato nei moduli STIMA.

Stazioni fisse o mobili

È possibile installare stazioni sia fisse (la cui posizione non cambia nel tempo) che mobili, sia terrestri che marine; le stazioni sono quelle che acquisiscono i parametri ambientali. Per aggiornare le posizioni delle stazioni mobili viene utilizzato un ricevitore GPS, che può essere o a bordo del modulo Stima o quello integrato in uno smartphone o in un tablet basato su Android.

Differenti tipologie di rete

La configurazione della rete può essere differente a seconda delle esigenze; oltre alla classica configurazione a stella (moduli master e base) con un broker al centro, è disponibile la configurazione ad albero sia via cavo (modulo master + base) che via radio: con la possibilità di utilizzare moduli radio di maggiore potenza (portata di circa 1 km in aria libera) è possibile prevedere coperture di un territorio di ampia superficie.

Salvataggio locale dei dati

I dati possono essere pubblicati in real time e/o salvati localmente. È previsto un meccanismo di salvataggio dei dati su SD-Card

formattata FAT; i file vengono frammentati con una dimensione prefissata per farne circa uno al giorno e numerati da 000 a 999; i dati salvati prevedono un flag che indica se sono stati già pubblicati correttamente su MQTT. I file che devono essere controllati per possibili reinvi hanno estensione **.que**, mentre l'estensione di quelli che hanno tutti i dati già inviati è **.don**. Con tali premesse si ottengono queste funzionalità:

- salvataggio dati su SD almeno per due anni con campionamenti ogni 5 s (un parametro);
- reinvio automatico al server dei dati salvati ma non pubblicati correttamente sul server;
- ottimizzazione dei tempi, in quanto solo i file che contengono dati da inviare vengono letti per selezionare i dati da reinviare ;
- i dati possono essere rilette su un normale PC estraendo la SD-Card.

Messaggistica di diagnostica

C'è la possibilità di ottenere un'ampia messaggistica di diagnostica per la soluzione dei problemi; attiva per impostazione predefinita, può essere disabilitata.

Configurazione

Le versioni delle configurazioni vengono verificate e quando il firmware non è retrocompatibile il modulo resta in attesa di una nuova configurazione. Si può forzare la configurazione tramite un apposito ponticello sulla board Stima-I2C. Le configurazioni vengono subito verificate: non è possibile configurare un modulo con dei sensori non corretti o non funzionanti.

Operazioni periodiche di manutenzione

Il software effettua periodicamente tutte le funzioni di manutenzione necessarie al corretto

funzionamento, quali quelle relative al DHCP o alla sincronizzazione dell'orologio interno con una sorgente esterna, la gestione dei pacchetti per la manutenzione dei protocolli su TCP/IP o via radio. Tutti i firmware hanno attivo un watchdog hardware che evita blocchi permanenti dovuti a malfunzionamenti su eventi improbabili; ogni 8 secondi, quindi, il watchdog deve essere reinizializzato per evitare un reset del microcontrollore.

Orologio di riferimento

Quando è necessario salvare i dati localmente (su SD-Card) e la connessione utilizzata per pubblicare i dati sul server (broker) non è considerata stabile, bisogna disporre di una base dei tempi precisa. Se invece la connessione (trasporto) viene considerata stabile (o non sia necessario recuperare i dati in caso di guasto) un preciso orologio di riferimento non è necessario e il tempo di riferimento verrà aggiunto automaticamente dal server in tempo reale alla pubblicazione del dato. Ci sono diversi sistemi per avere un orologio di riferimento preciso sui moduli Stima.

Crittografia

Qualora il trasporto non sia considerato sicuro (via radio) viene utilizzata la crittografia per garantire riservatezza e autenticità.

Risparmio energetico

Nello sviluppo del firmware, particolare attenzione è stata posta alla limitazione dei consumi: quando possibile, i microcontrollori e i sensori vengono messi in sleep e sono alcuni interrupt a risvegliare il sistema. Questo agevola l'utilizzo con batterie dei sistemi a basso consumo quali il modulo Stima-satellite che funziona con un modulo radio.

Integrazione con la domotica

Per quello che è stato possibile, nello sviluppo di STIMA si è cercato di integrarsi con gli standard della domotica (MQTT). Tutti i moduli possono essere utilizzati anche da attuatori on/off (fino a quattro relé) ma è molto semplice aggiungere altre funzionalità tramite remote procedure in formato json su tutti i trasporti o tramite MQTT.

LIBRERIE

Analizzate le caratteristiche del firmware, vediamo adesso le librerie che esso utilizza e che in parte abbiamo visto nella puntata precedente.

aJson

Questo è un tentativo di portare una completa implementazione di JSON nell'ambiente Arduino; la libreria è basata su cJSON, ridotta di dimensione. Di JSON e JSON-RPC abbiamo parlato nella scorsa puntata in quanto tutte le comunicazioni implementate dal nostro firmware si basano su tale standard. La libreria è stata adattata per ridurre l'uso della memoria e implementare alcuni tipi di dato non supportati.

JsonRPC

JsonRPC è la libreria che implementa un sottoinsieme del protocollo JSON-RPC.

PubSubClient

Fornisce un client per semplici publish/subscribe, scambiando messaggi con un server che supporta MQTT (broker); è adattata per funzionare, oltre che con l'ethernet, anche con GSM/GPRS.

Arduino_uip

Implementa le stesse API della Arduino Ethernet library, ma utilizzando l'ENC28J60 Microchip come chip per la comunicazione

ethernet. Oltre a creare un'implementazione completamente open, infatti, questa libreria permette di utilizzare l'ENC28J60, che è molto più economico dei chip con stack IP incluso. Offre supporto completo per le connessioni TCP e UDP persistenti (streaming) (Client e Server), ARP, ICMP, DHCP e DNS. La libreria è stata sviluppata da Norbert Truchsess derivandola dallo stack uIP di Adam Dunkels.

RF24/RF24Network

OSI Network Layer ottimizzato per moduli radio nRF24L01(+) 2,4GHz ISM. Abbiamo scelto l'ultima versione elaborata da TMRh20, che dalle nostre prove è risultata la più stabile e capace di offrire più opzioni, oltre che di funzionare con Arduino e Raspberry Pi. Supporta una topologia di rete a stella o ad albero; con il modulo da noi utilizzato si possono coprire distanze di circa 50 metri in aria libera. Queste le caratteristiche principali di funzionamento: *Host Addressing* ogni nodo ha un indirizzo logico nella rete local; *Message Forwarding* i messaggi possono essere mandati da un nodo a qualsiasi altro nodo senza limite al numero di "salti" che il messaggio deve fare; *Ad-hoc Joining* un nodo può entrare a far parte della rete senza nessun cambiamento alla configurazione dei nodi già esistenti. Per quanto riguarda *RF24Network indirizzamento e topology*, a ogni nodo deve essere assegnato dall'amministratore un indirizzo a 15-bit. Questo indirizzo descrive esattamente la posizione del nodo all'interno della rete ad albero. L'indirizzo è un numero in formato ottale dove ogni cifra rappresenta una posizione all'interno dell'albero, partendo dalla base:

- Nodo 00 è il nodo base;
- Nodi 01-05 sono nodi il cui genitore è la base;

- Nodo 021 è il secondo figlio del nodo 01;
- Nodo 0321 è il terzo figlio del nodo 021, e così via.

L'indirizzo di nodo più grande è 05555, così in teoria sono possibili fino a 3,125 nodi in singolo canale.

sim800

Questa libreria è stata sviluppata ex novo in quanto le funzionalità di cui c'era bisogno non sono disponibili in nessun'altra libreria di gestione dei moduli GSM SIM800/SIM900 della SIMCom.

La libreria implementa:

- protocollo TCP/IP transparent mode con le API Etherlib;
- http in modalità nativa sim800;
- utilizzo dell'RTC interno alla sim800.

Relativamente a, TCP/IP, una volta che la connessione è stabilita nella modalità trasparente, tutti i dati ricevuti dalla porta seriale saranno trattati per formare pacchetti da essere poi trasmessi, similmente tutti i dati ricevuti dal lato remoto saranno inviati direttamente alla porta seriale.

SensorDriver

Libreria di "driver" per la gestione dei sensori; di essa esistono attualmente due versioni: una in C++ e una in python. La libreria porta la gestione della sensoristica ad un livello di astrazione più alto. In essa, aggiungere un nuovo tipo di sensore consiste nell'estendere una classe con quattro metodi per effettuare la lettura di quello specifico sensore; il comando:

```
int setup (int address);
```

effettua eventuali settaggi necessari al funzionamento del sensore. Ad esempio, per la temperatura la sintassi è: numero di bit di

risoluzione, operazione di misura one-shot. Il comando:

```
int prepare (unsigned long* waittime);
```

impartisce al sensore il comando per effettuare una singola misurazione e ritorna il tempo in millisecondi di attesa necessario.

Invece il comando:

```
int get (int* value);
```

ritorna i valori della misurazione. Il comando:

```
JsonObject* getJson();
```

ritorna i valori misurati in formato compatibile json.

Time/TimeAlarms

Time fornisce un orologio software che può essere sincronizzato con sorgenti esterne per mantenerlo sempre preciso. Nel nostro caso utilizziamo NTP se è disponibile la rete ethernet, l'RTC della scheda sim800 o l'RTC del DS1307 tramite I²C. *TimeAlarms* unitamente a *Time* esegue funzioni a istanti di tempo specificati, una-tantum o periodicamente.

SdFat

SdFat è una libreria per Arduino che supporta FAT16 and FAT32 file systems su SD-Card standard o ad alta capacità (SD/SDHC Flash card). SdFat supporta la creazione di file, cancellazione, read, write, oltre al troncamento. SdFat supporta anche l'accesso a subdirectories, nonché la creazione e cancellazione di subdirectories. Supporta Long File Names e usa la libreria Arduino SPI.

AESLib

Questa libreria implementa la crittografia tramite AES per Arduino; è un metodo per rendere un messaggio "offuscato" in

modo da non essere comprensibile/intelligibile a persone non autorizzate a leggerlo. Questo è particolarmente importante quando si utilizza un trasporto non sicuro quale quello via radio. La crittografia, nel nostro firmware è utilizzata per le comunicazioni con nrf24, ma il livello di sicurezza non è ancora elevato e l'implementazione è incompleta.

YwrobotLiquidCrystal_I2C

La libreria per la gestione del display LCD tramite I²C-Bus possiede tutte le funzioni presenti nella libreria di sistema "LiquidCrystal". Inoltre supporta gran parte delle funzioni listate nella specifica ufficiale per i display LCD.

IL FILE DI CONFIGURAZIONE A TEMPO DI COMPILAZIONE

Il firmware STIMA fa un uso intensivo delle direttive del preprocessore C; definendo delle variabili è possibile ottenere da un unico codice diversi firmware per i vari moduli. Il file *rmap_config.h* situato in *sketchbook/rmap/rmap* comanda quasi tutte le opzioni e le configurazioni hardware; contiene i template da utilizzare per ottenere i firmware per i moduli qui presentati. Non riuscendo, in queste pagine, ad analizzare in modo completo tutto il firmware, ci limitiamo a evidenziare le funzioni principali contenute in *sketchbook/rmap/rmap/rmap.ino*. Nel main loop troviamo:

- *mgrserialjsonrpc*; provvede alla gestione della porta seriale per jsonrpc;
- *mgrnrf24jsonrpc*; provvede alla gestione della comunicazione radio e jsonrpc;
- *mgrmqtt*; esegue la gestione del protocollo MQTT e jsonrpc;
- *mgrethserver*; gestisce il server tcp/ip per jsonrpc.

La funzione *mgrjsonrpc* è la funzione utilizzata per gestire il protocollo json-RPC. Se il modulo è attivo periodicamente viene eseguita la funzione *Repeats*, che svolge le operazioni di interrogazione dei sensori e pubblicazione dei dati.

L'HARDWARE DEL PROGETTO

È giunto il momento di descrivere le tre configurazioni proposte in modo da spiegare che tipo di misura possono eseguire e quale connettività possono supportare. Ogni modulo consta di una scheda Microduino e di alcuni shield per essa.

MODULO STIMA MASTER

Questo modulo è ideale per gli appassionati di meteorologia che sognano una stazione meteo fissa, con sensori locali e remoti collegati sia con cavo che via radio. Il Modulo Stima Master può essere dotato di display LCD per il controllo dello stato del sistema e trasmette i dati tramite ethernet al server RMAP. Vediamo da quali board è composto.

Board microduino core+ 1284

La board core+ 1284 è molto simile al modello core+ 644 che abbiamo visto nello scorso numero, solo che è più prestante; l'abbiamo scelta perché il Modulo STIMA Master ha bisogno di un microcontrollore dotato di molta memoria Flash e SRAM per le diverse funzioni che svolge. Sulla board, che rappresenta il modello più evoluto fra le Microduino core, il componente principale è l'Atmega1284P funzionante a 16 MHz, dotato di 128 K di memoria Flash e 16 K di SRAM. Il firmware che vi carichiamo è molto articolato

perché gestisce comunicazioni via LAN e via radio, nonché tutti i protocolli R-map.

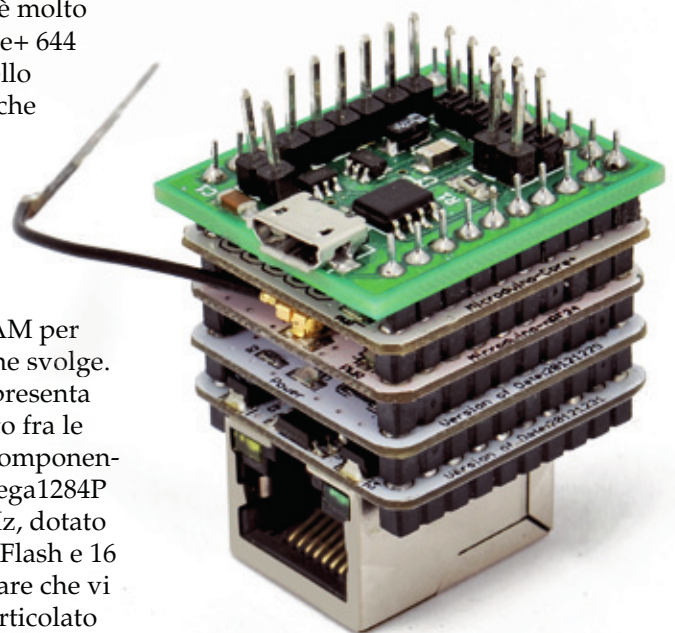
Board microduino ENC

Permette di collegare alla rete cablata il modulo Stima Master; è ancora uno shield Microduino, o meglio, una coppia: una scheda che alloggia il connettore RJ45 e lo stabilizzatore switching per il PoE, ed una seconda sulla quale si trova l'integrato ENC28J60, compatibile con lo standard IEEE 802.3. Questo integrato prodotto da Microchip integra entrambe le funzioni MAC e 10BASE-T PHY e permette una gestione semplificata per il microcontrollore di tutto lo stack di comunicazione Ethernet; anche questa board dialoga tramite la porta SPI.

Microduino ENC ha anche una caratteristica molto interessante, permette l'alimentazione della board e delle altre schede in modalità PoE.

Utilizzando un apposito adattatore (ad esempio quello

STIMA Master al completo.



commercializzato dalla Futura Elettronica con il codice 7300-SETCAVIPOE) ed un alimentatore da 12 a 24 volt (a seconda della distanza del collegamento) possiamo alimentare il modulo tramite un solo cavo, usato sia per la comunicazione Ethernet che per l'alimentazione del modulo. Ciò facilita notevolmente i cablaggi.

Board STIMA-I2C

Questa board l'abbiamo vista nella prima puntata ed è utilizzata da tutti i moduli; ha diverse funzioni: genera 3,3 volt per l'alimentazione, protegge i componenti da scariche elettrostatiche ed espone il bus I2C per il collegamento dei sensori con un comodo connettore.

Board microduino nRF24 (opzionale)

Questa board, che è opzionale, è prodotta da Microduino e progettata per comunicare via radio in banda ISM alla frequenza di 2,4 GHz. Componente principale della scheda è il diffusissimo integrato nrf2401+ prodotto da Nordic Semiconductor, che permette comunicazioni radio fino a qualche decina di metri di distanza in campo aperto. La board permette la ricezione dei dati da moduli STIMA Satellite, oltre che da sensori remoti che possiamo collocare sul terrazzo o in punti non collegabili con cavo.

Sulla scheda è previsto un connettore SMA cui collegare un'antenna fornita in dotazione al modulo o una a nostra scelta, essendovi un attacco di tipo standard.

MODULO STIMA-SATELLITE

Si tratta di un'unità, composta al solito da più schede, che è di tipo "passivo" in quanto rimane in attesa di richieste via radio da un modulo STIMA Master o da uno STIMA-GSM/GPRS; all'arrivo di una richiesta, il modulo interroga i sensori collegati via I2C-Bus e ne trasmette le misurazioni al modulo Master. Il modulo STIMA Satellite si può utilizzare nel sistema solo se sul modulo STIMA Master o GSM/GPRS avete montato la board Microduino nRF24.

Lo STIMA Satellite è composto, oltre che da Board STIMA-I2C (per la quale vale quanto già detto nel modulo Master) e da una board microduino nRF24, da una Microduino core+ 644 che ne costituisce il cervello.

La board nRF24 gestisce i sensori e le comunicazioni radio. È possibile utilizzare più moduli Stima-satellite: fino a cinque che comunicano direttamente con lo STIMA Master, più eventuali

altri disposti in una configurazione ad albero. È necessaria un'opportuna configurazione, ma in questo caso la possibilità di far fare ai dati "salti multipli" rende possibile l'uso di questi sensori in campo aperto su medie distanze, alimentando il modulo con batterie.

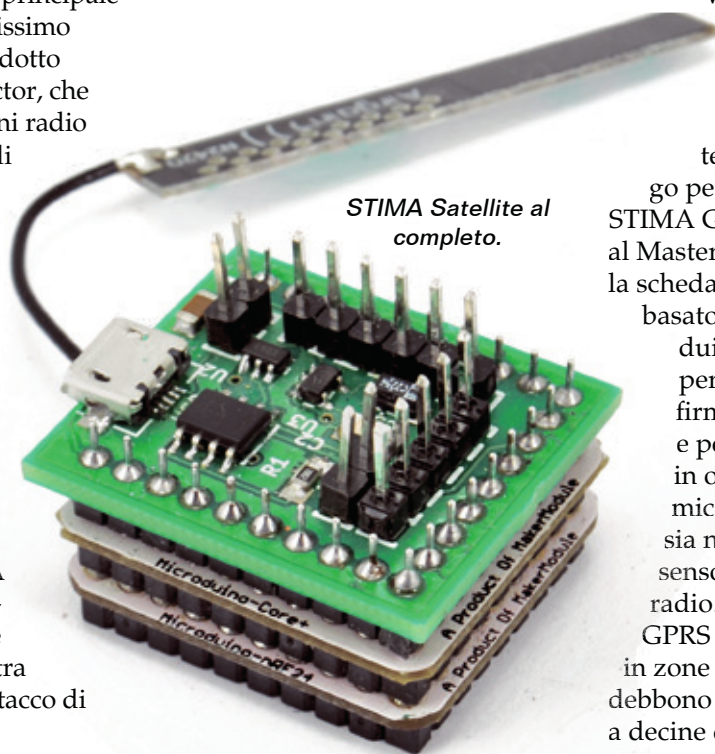
Board microduino core+ 644

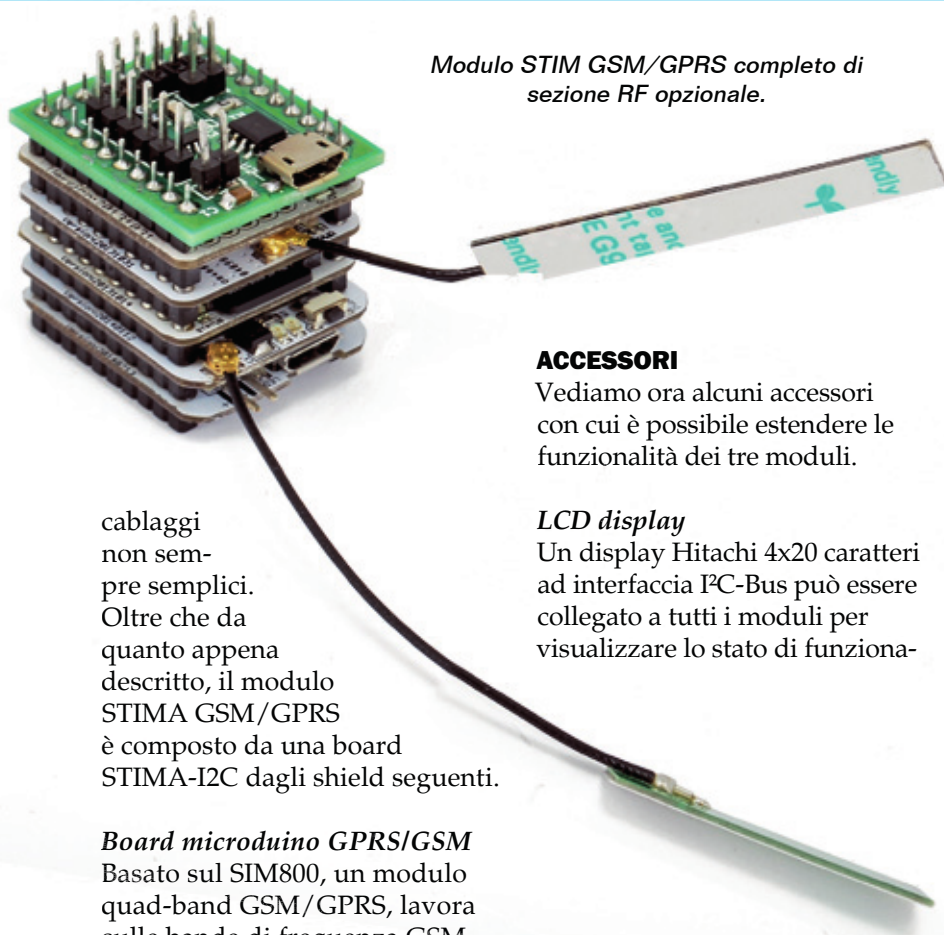
Trattandosi di un modulo con funzionalità meno articolate del modulo Master, la board core+ 644 è sufficiente per eseguire le funzioni richieste, perché ha un firmware di dimensioni contenute se non si utilizza la crittografia o altre estensioni software.

MODULO STIMA GSM/GPRS

Abbiamo analizzato il modulo Stima-Master, che comunica con il server RMAP attraverso rete cablata, ma per utilizzare il sistema STIMA in zone senza connessione ethernet è possibile utilizzare questo modulo con 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-Card. STIMA GSM/GPRS è simile al Master per quanto riguarda la scheda Arduino (è infatti basato su una board Microduino core+1284, scelto per eseguire ed ospitare il firmware, che è complesso e pesante) e può disporre, in opzione, di una board microduino nRF24 laddove sia necessario aggiungere sensori remoti collegati via radio. Infatti lo STIMA GSM/GPRS nasce per essere usato in zone isolate, dove i sensori debbono essere montati su pali o a decine di metri di distanza con





Modulo STIM GSM/GPRS completo di sezione RF opzionale.

cablaggi non sempre semplici. Oltre che da quanto appena descritto, il modulo STIMA GSM/GPRS è composto da una board STIMA-I2C dagli shield seguenti.

Board microduino GPRS/GSM

Basato sul SIM800, un modulo quad-band GSM/GPRS, lavora sulle bande di frequenza GSM 850 MHz, EGSM 900 MHz, DCS 1.800 MHz e PCS 1.900 MHz. Il SIM800 è un GPRS multi-slot class 12/class 10 e supporta gli schemi di codifica GPRS CS-1, CS-2, CS-3 e CS-4. Per gestirlo si utilizzano comandi AT impartiti via seriale con un data-rate che può variare tra 1.200 e 115.200 bps. Il transfer-rate della comunicazione su GPRS è di 85,6 kbps sia in uplink che in downlink.

Board microduino SD

Per ridurre i costi di connessione o evitare perdite di dati in caso di mancata connessione al server, il modulo memorizza i dati su SD-Card, che nello specifico è nel formato microSD. Quando la connessione al server viene ristabilita, i dati vengono inviati alla destinazione prevista. La memorizzazione dati su SD consente di salvare dati campionati ogni 5 secondi per un periodo massimo di due anni.

ACCESSORI

Vediamo ora alcuni accessori con cui è possibile estendere le funzionalità dei tre moduli.

LCD display

Un display Hitachi 4x20 caratteri ad interfaccia I²C-Bus può essere collegato a tutti i moduli per visualizzare lo stato di funziona-

mento, la connessione al server, l'ora e i dati rilevati, permettendo una diagnostica immediata.

Relé

Siccome il sistema STIMA è predisposto per la domotica, sulla board Stima-I2C si trova un connettore pronto per collegare moduli da uno a quattro relé pilotabili con JsonRPC. Dato che la scheda equipaggia tutti e tre i moduli, ogni modulo può pilotare degli attuatori.

Sensori

L'ampia possibilità di collegamento dei sensori sul bus I²C verrà esaminata nella prossima puntata; per ora utilizziamo per i test il sensore di temperatura incluso sulla board STIMA-I2C.

COMPILAZIONE E INSTALLAZIONE DEL FIRMWARE

Passiamo adesso al firmware, installando per prima cosa l'IDE

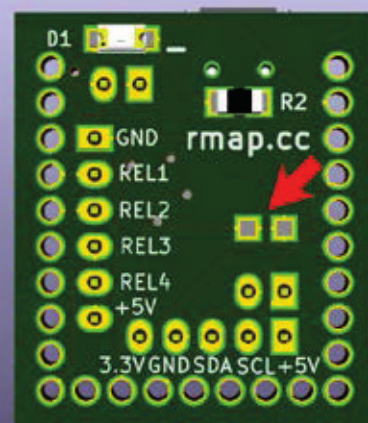


Fig. 1 - Le piazzole SET sono dal lato saldature della board STIMA-I2C.

Arduino 1.6.5 da <https://www.arduino.cc/en/Main/Software>.

Scaricate l'ultima versione del software stima (file *stima-sketchbook*) da <https://github.com/r-map/rmap/releases> e scompattate il file zip stando attenti a ignorare i file "Source code". Aprite l'IDE Arduino e in *File->Impostazioni* disabilitate "Controlla aggiornamenti all'avvio" poi in *File->Impostazioni->Percorso della cartella degli sketch* selezionate la cartella *sketchbook* appena scompattata dal file scaricato e chiudete e riaprite l'IDE. In questo modo, quando compilerete il firmware utilizzerete alcune opzioni specializzate contenute nel file *sketchbook/hardware/Microduino/avr/platform.local.txt*, necessarie per ottenere un firmware funzionante.

Per quanto riguarda il modulo Stima-master, se in esso avete montato anche la board microduino nRF24 e quindi volete utilizzare anche il modulo Stima-Satellite, con un editor modificate il file *sketchbook/libraries/SensorDriver/SensorDriver_config.h* decommentando l'opzione "#define RADIORF24"; decommentando anche l'opzione "#define AES" abiliterete anche la crittografia AES, tuttavia consigliamo l'utilizzo di quest'ultima opzione solo ai più esperti. In *sketchbook/rmap/rmap* dovete copiare il file *stima_master.h* in *rmap_config.h*.

Listato 1

```
rmapctrl --syncdb
rmap-configure --station_slug=ETH_fixed --board_slug=master_eth_fi-
xed --height=<altezza stazione in metri> --stationname="<descrizione
della stazione>" --username=<il vostro utente> --password=<la vostra
password> --server=rmap.cc --samplerate=5 --lat=<latitudine stazione>
--lon=<longitudine stazione> --addensors_by_template=<sensortemplate>
--wizard --config_station --upload_to_server
```

Poi andate nel menu *Strumenti* e nella voce *Scheda* selezionate *Microduino Core+ (1284pa)*; nella voce *Processore* scegliete *ATmega1284pa@16M5V* e in *Porta* impostate la COM disponibile. Adesso, impartite il comando *Strumenti->Cartella degli sketch->rmap->rmap*; infine, impartite il comando *Sketch->Carica*.

Passiamo adesso alla configurazione per il modulo STIMA Satellite: in *sketchbook/rmap/rmap* copiate il file *stima_satellite.h* in *rmap_config.h*. A questo punto dal menu *Strumenti* scegliete:

- *Scheda: Microduino Core+ (644pa)*;
- *Processore: ATmega644pa@16M5V*;
- *Porta: quella disponibile*.

Impartite quindi il comando *Strumenti->Cartella degli sketch->rmap->rmap* e poi *Sketch->Carica*. Concludiamo con la configurazione per il modulo STIMA GSM/GPRS: in *sketchbook/rmap/rmap* copiate il file *stima_gsm.h* in *rmap_config.h*; il file è predisposto per l'utilizzo di una SIM-Card della TIM. Laddove ne utilizzate una di diverso operatore, inserite in fondo al file *rmap_config.h*:

```
#define GSMAPN ""
#define GSMUSER ""
#define GSPASSWORD ""
```

scrivendo in GSMAPN l'APN fornito dal vostro operatore, in GSMUSER lo username con cui accedete a Internet e in GSPASSWORD la password per l'accesso. Salvate quindi il file. Ora aprite il file accessibile da

sketchbook/libraries/PubSubClient/PubSubClient.h e modificate:

```
//if use sim800 client
#include "sim800Client.h"
#define TCPCLIENT sim800Client
```

con questa porzione di codice:

```
//if use arduino uip or etherclient
//#include "Client.h"
//#include "Stream.h"
//#define TCPCLIENT Client
```

Nel menu *Strumenti* scegliete:

- *Scheda: Microduino Core+ (1284pa)*;
- *Processore: ATmega1284pa@16M5V*;
- *Porta: quella disponibile*.

Sempre in *Strumenti*, impartite il comando *Cartella degli sketch->rmap->rmap*. Infine, impartite il comando *Sketch->Carica*.

CONFIGURAZIONE

Per configurare le board dovete utilizzare l'applicazione grafica o l'utilità a linea di comando. Come avere il software installato sul proprio computer lo abbiamo spiegato nella scorsa puntata, ma le istruzioni si possono anche recuperare sul web da http://www.raspibo.org/wiki/index.php/Gruppo_Meteo/HowTo.

Poi per pubblicare i dati sul server RMAP.cc bisogna registrarsi al sito; il pulsante "Registrazione" dell'app dovrebbe aprire un browser alla url della registrazione (<http://rmap.cc/registrazione/register/>). Una volta fatta la registrazione sarete in possesso di uno user e di una password.

A questo punto dovrete trasferire

la vostra configurazione sulla EEPROM del microcontrollore e allo scopo:

- ponticellate sulla board Stima-I2C le piazzole "SET" (Fig. 1).
- collegate il modulo con la board Microduino FT232RL alla USB del vostro PC.

Configurazione tramite l'applicazione grafica

Aprite l'applicazione grafica e dal menu *Impostazioni* che si aprirà automaticamente al primo avvio, nella sezione "Rmap" dovrete inserire "RMAP user" e "RMAP password" ottenuti durante la registrazione a rmap.cc.

Adesso, dal menu *Impostazioni* selezionate la sezione "Sensors" e impostate:

- per il modulo Stima-Bluetooth:
 - Name: HC-05;
 - Station: BT_fixed;
 - Board:
 - su android: BT_fixed;
 - su linux: BT_fixed_LINUX;
 - su windows: BT_fixed_WINDOWS;
 - su OSX: BT_fixed_OSX.
 - Template: test_indirect;

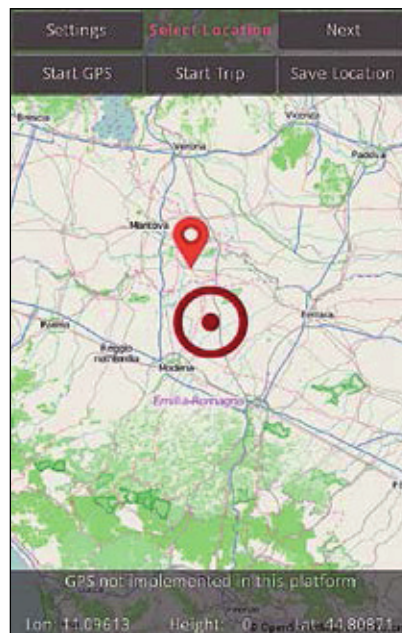


Fig. 2

- Remote Board: stima_bt;
- Remote Template: test ;
- per il modulo Stima-Master o Stima-GSM/GPRS:
 - Station: ETH_fixed;
 - Board:
 - su linux: rmapgui_LINUX;
 - su windows: rmapgui_WINDOWS;
 - su OSX: rmapgui_OSX.
 - Template: test_indirect;
 - Remote Board: master_eth_fixed;
 - Remote Template: test (test_master se avete la board nRF24).

Per il modulo Stima-Satellite fate come per lo Stima-Master, ma in Remote Board scegliete satellite_eth_fixed. Nella sezione "Location" potete inserire manualmente le vostre coordinate e selezionare "Close" attivando così la stazione. Se non conoscete le vostre coordinate, dalla pagina "Posizione" selezionate accuratamente la vostra posizione con il mouse e salvatela con il pulsante "Salva posizione" (Fig. 2). La prima pagina dell'App "Avvia" presenta un manuale che potrà aiutarvi. Dalla pagina "Dati automatici" fate clic sul pulsante "configura" e verificate che tutto vada a buon fine.

Configurazione a linea di comando

È possibile eseguire tutte le configurazioni con due comandi da terminale: da Windows utilizzate la finestra che rimane aperta dopo aver eseguito `rmap-configure.bat`; su Linux o OSX attivate il Virtualenv di python come per eseguire l'App grafica. Il primo comando inizializza il DB. Il secondo, con l'opzione `--wizard` inserisce tutti i metadati nel DB, `--config_station` trasferisce la configurazione sulla EEPROM del microcontrollore

e `--upload_to_server` comunica i metadati al server (Listato 1). Ovviamente sostituite il contenuto tra <> con i vostri dati; cercate di avere latitudine e longitudine definite fino alla quinta cifra decimale; <sensortemplate> dovrà essere "test_master" se il vostro modulo Stima-master o Stima-GSM/GPRS comprende una board microduino nRF24, "test" in tutti gli altri casi.

Operazioni finali

Scollegate il modulo dalla USB, rimuovete la board Microduino FT232RL e attivate l'alimentazione a 3,3V chiudendo il ponticello JP2 della board Stima-I2C.

MESSA IN OPERA

Sul modulo Stima-GSM/GPRS inserite una micro SD formattata FAT; dovete inserire anche una SIM-Card. Alimentate i moduli tramite il connettore micro-USB sulla board Stima-I2C; il modulo Stima Master potete alimentarlo con l'apposito Injector e un alimentatore da 12 a 24V (connettore con positivo al centro). Sul modulo Stima-GSM/GPRS il SIM800 va acceso manualmente tenendo premuto l'apposito switch. Sulla rete ethernet dovrete avere un server DHCP in quanto STIMA Master lo richiede. Se tutto funziona regolarmente, ogni 5 secondi i dati della temperatura di test del modulo verranno inviati a `rmap.cc`. A questo punto se sul server `rmap.cc` fate login con il vostro nome utente, accederete alla vostra pagina personale con l'elenco delle stazioni di misura e la possibilità di visualizzare i grafici (Fig. 3) dei vostri dati. Per ora potete inviare solo i dati di test (temperatura del modulo); nella prossima puntata vedrete come aggiungere sensori per la visualizzazione dei dati.

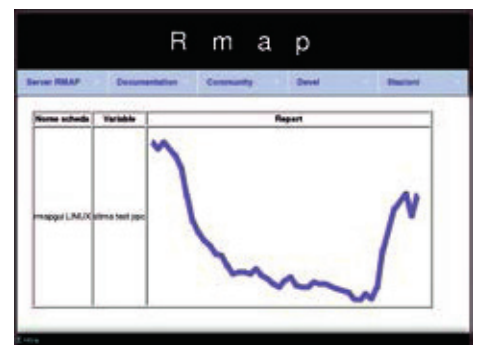


Fig. 3



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. I master dei circuiti stampati possono essere scaricati dal sito della rivista. I componenti elencati di seguito si possono acquistare presso Futura Elettronica:

- Microduino Core ATMEGA1284 (cod. MDUINO1284) a Euro 22,00.
- Microduino Core+ ATMEGA644PA (cod. MDUINOCORE+) a Euro 15,50.
- Microduino Shield USB/TTL (cod. MDUINOUSBTTL) a Euro 19,00.
- Microduino Shield SDcard (cod. MDUINOSD) a Euro 12,50.
- Microduino Shield NRF24 (cod. MDUINONRF24) a Euro 16,00.
- SET CAVI POE (cod. SETCAVIPOE) a Euro 6,00.
- Display LCD 20x4 con interfaccia I²C (cod. LCD20X4AI2C) a Euro 26,00.
- MODULO 4 RELÈ 5VDC 10A (cod. RELAY4CH) a Euro 13,00.
- MODULO 2 RELÈ 5 VDC 10A (cod. RELAY2CH) a Euro 8,00.
- Modulo 1 relè 5 Vdc 10A (cod. RELAY1CH) a Euro 5,00.

Tutti i prezzi si intendono IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775 - Fax: 0331-792287
<http://www.futurashop.it>