

Il mondo dell'



dell'Ing. MIRCO SEGATELLO



**Sperimentiamo l'IOT con
l'impiego della rete a basso
consumo SigFox, alternativa
alle reti wifi e GSM.
Quarta e ultima puntata.**

Durante questo corso, dopo aver spiegato in cosa consiste l'IoT e quali siano i suoi utilizzi, ci siamo occupati di alcune implementazioni facenti uso di schede Arduino MKR appositamente progettate per lo scopo. Nella seconda lezione abbiamo implementato un dispositivo IOT su rete WiFi, mentre nella terza lezione è stata usata una scheda MRX 1300 su rete LoraWAN. Quando si parla di IoT sembra imprescindibile parlare di reti LPWAN (Low Power Wide Area Network) ovvero di quella tecnologia che permette trasmissioni utilizzando moduli radio a bassissima potenza ma con lungo raggio di copertura, anche se a discapito

della quantità e velocità dei dati trasmessi. Questo è valido in quanto a livello europeo la banda di frequenze comprese tra 865MHz e 870MHz è di libero utilizzo a patto di non utilizzare trasmettitori con potenza superiore ai 25mW e con un tempo di utilizzo inferiore all'1% ovvero 36 secondi ogni ora. In questa ultima lezione ci soffermiamo sulla soluzione fornita dalla compagnia francese SigFox che dispone di una rete di telecomunicazioni specificatamente progettata per l'IOT, attualmente presente in Europa occidentale, San Francisco, ed in fase test in Sud America e Asia. La rete SigFox utilizza la tecnologia LPWAN ed è stata progettata per l'invio

di messaggi di ridotte dimensioni in cui la comunicazione viene sempre avviata dal dispositivo finale, il quale non dovrà rimanere costantemente attivo in attesa di una ricezione. Questa modalità assicura un'elevata efficienza energetica che permette ad un dispositivo SigFox di durare anni con una batteria standard, come in apparecchiature per il rilievo di parametri fisici o ambientali (lettura dei contatori di acqua e gas) che non richiedono in invio continuo dei dati.

La **Fig. 1** schematizza la rete SigFox.

La comunicazione SigFox utilizza una modulazione Ultra Narrow Band (UNB) senza alcuna negoziazione dei parametri della connessione tra il dispositivo e la base station ricevente, permettendo di ridurre la quantità di dati scambiati senza rinunciare all'immunità alle interferenze. Il dispositivo semplicemente emette nella banda di frequenza disponibile ed il segnale viene rilevato dalle base station più vicine, decodificato e trasmesso al back-end della rete, dove i messaggi vengono poi inoltrati all'applicazione dell'utente e resi accessibili tramite le API di SigFox. Ogni messaggio viene autenticato utilizzando un meccanismo hash, e una chiave privata specifica del dispositivo garantendo l'impossibilità di hackerare i dati trasmessi; vediamo le caratteristiche più salienti.

È possibile inviare fino a 12 byte (96 bit) per messaggio ed il protocollo include già un timestamp (data e ora) della trasmissione e l'ID unico del dispositivo nei suoi metadata.

Il numero massimo di messaggi che possono essere inviati ogni giorno è di 140, limitazione dovuta al rispetto delle normative del settore in quanto, la normativa europea che disciplina la banda 868 MHz consente un duty cycle massimo di trasmissione del 1%. Un singolo dispositivo non potrà pertanto trasmettere per più dell' 1% del tempo in un'ora.

Giusto per completezza vediamo quali sono le differenze tra le due tecnologie trattate in questo corso, LoRaWAN e SigFox. SigFox usa uno protocollo di trasmissione radio

chiamato BPSK (Binary Phase Shift Keying) che utilizza una porzione molto piccola dello spettro radio permettendo di minimizzare l'effetto del rumore sulla comunicazione ed utilizzare apparati (endpoint) più semplici ed economici, ma richiede l'utilizzo di gateway/basestation molto più sofisticati e costosi. LoRa invece utilizza una tecnologia chiamata Spread Spectrum con un maggiore utilizzo di banda (oltre 125 kHz) che introduce molto più rumore, pertanto l'hardware è tendenzialmente più costoso ma può essere utilizzato indifferentemente su tutti i dispositivi (non vi è differenza tra endpoint e basestation) permettendo di avere gateway più economici e semplici da realizzare rispetto SigFox.

Le grosse differenze tra le due reti riguardano invece la gestione commerciale. LoRa è open source e di libero utilizzo ed i punti di accesso possono essere creati da tutti, anche privati, non ci sono costi per accedere alla rete, pertanto è considerata la rete delle persone ed è in continua espansione proprio grazie agli utenti stessi che desiderano installare un punto di accesso per la loro zona. L'unico aspetto negativo è che i moduli radio Lora sono prodotti da un'unica azienda (solo recentemente sono stati stipulati accordi commerciali per la produzione su licenza) con l'effetto di mantenere elevati i prezzi di vendita.

SigFox invece si comporta come un operatore telefonico noleggiando l'accesso alla rete tramite i suoi gateway, oppure gateway di aziende partner con le quali ha stipulato contratti commerciali. Il costo è di circa 1 euro al mese per singolo device. Di contro la produzione dell'hardware è demandata a molti produttori in regime di libero mercato permettendo prezzi di vendita decisamente contenuti.

Nettrotter è l'unico operatore Italiano in grado di implementare e gestire la rete SigFox sul territorio nazionale Italiano, avendo il ruolo di SNO (SigFox Network Operator). Questa iniziativa fa parte di un più ampio progetto, coordinato da



Fig. 1 - Struttura della rete SigFox.

Sigfox service maps

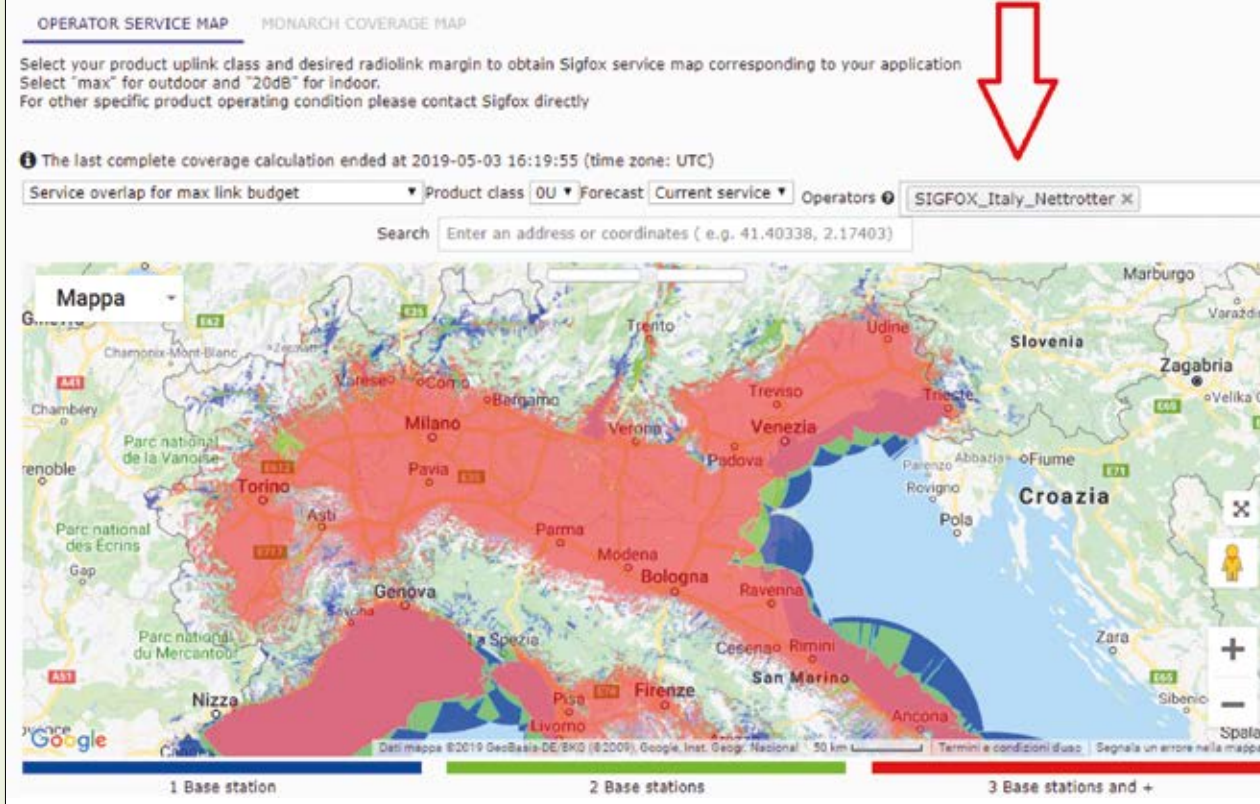


Fig. 2 Copertura della rete SigFox.

SigFox, che ha l'obiettivo di sviluppare un network globale per l'IoT a supporto di una ampia gamma di applicazioni. La rete offerta da Nettetrotter sfrutta le esistenti torri TLC e televisive e prevede di raggiungere una copertura nazionale, con quasi 1.000 Base Station SigFox già installate.

Già coperte oltre 40 tra le principali città italiane, fra cui Roma, Milano, Torino, Bologna, Firenze, Napoli, Bari, Reggio Calabria, Palermo e già servita l'80% della popolazione. La Nettetrotter fornisce il sistema Telenet Water, prima soluzione Italiana su rete SigFox per lo smart metering idrico, caratterizzata da bassi consumi ed alta sensibilità in ricezione del segnale trasmesso dall'apparato. La soluzione consente, previa applicazione del prodotto sul contatore dell'acqua, la trasmissione in real time dei dati caratterizzanti la distribuzione dell'acqua potabile alle utenze domestiche ed industriali con una durata della batteria fino a 15 anni. Per conoscere la copertura nella vostra zona dovete accedere al portale (<https://backend.sigfox.com/auth/login>), cliccare sul logo in alto a sinistra, selezionare service maps e selezionare come provider SIGFOX_Italy_Nettetrotter, come visibile nella Fig. 2.

Vediamo ora come implementare praticamente un dispositivo che sfrutti le potenzialità della rete SigFox e lo facciamo

utilizzando la scheda Arduino MKR FOX 1200 (Fig. 3), appositamente progettata allo scopo. Come già detto nelle precedenti puntate le MKR sono schede basate su MCU SAM D21E (Microchip ARM Cortex M0) operanti alla frequenza di 48MHz che includono 256K di memoria Flash e 32K di memoria SRAM, quindi decisamente più preformanti di una scheda Arduino UNO. La parte di comunicazione SigFox è demandata ad un IC ATA8520 della Microchip, il quale comunica con la



Fig. 3 - La scheda Arduino MKR FOX 1200 con antenna

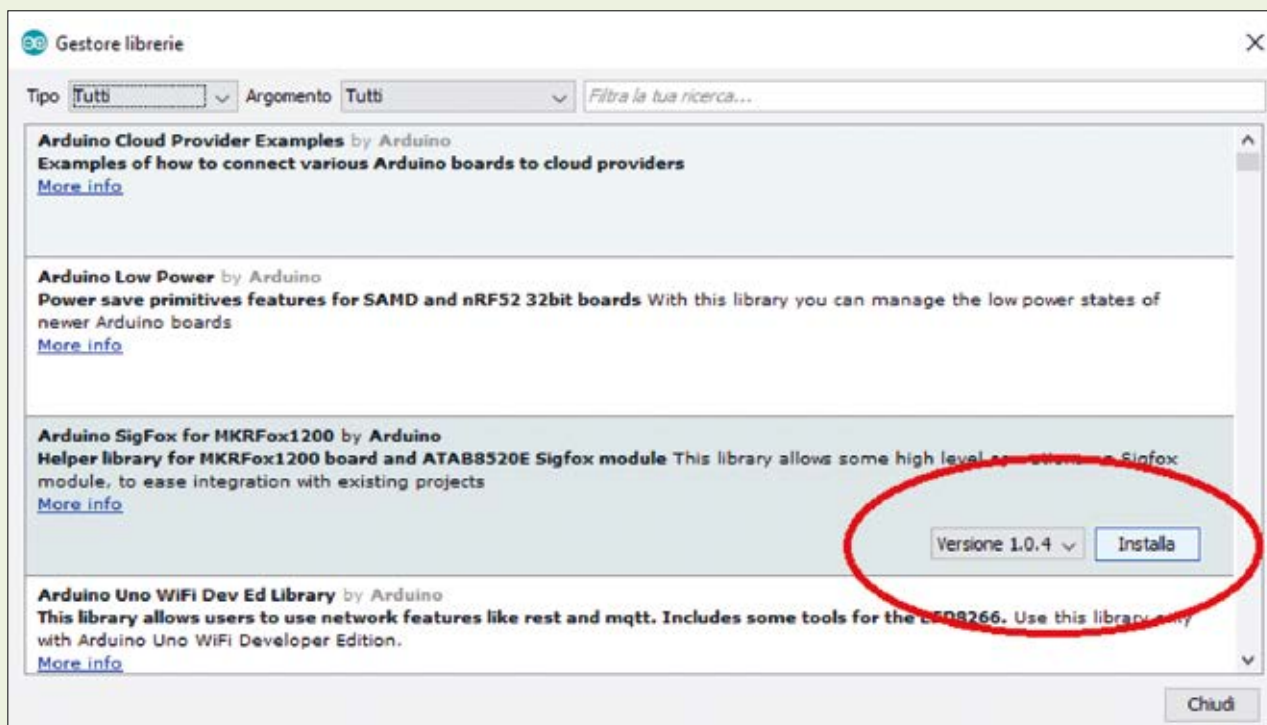


Fig. 4 - Installazione della libreria per l'applicazione SigFox.

CPU principale tramite bus SPI. Il prezzo di acquisto comprende un abbonamento per due anni ai servizi SigFox. L'utilizzo del modulo radio e l'accesso alla rete SigFox è notevolmente semplificato grazie a librerie scritte appositamente per lo scopo. Sarà quindi necessario installare le librerie: Arduino SAMD Boards, Arduino RTCZero, Arduino Low-Power, Arduino SigFox for MKRFox1200.

Aprire e caricare l'esempio di nome *FirstConfiguration.ino* contenuto in File-Esempi-Arduino SigFox for MKRFox 1200, su Serial Monitor potete leggere la versione del firmware del modulo radio e i due parametri univoci che identificano l'hardware ID e PAC (Fig. 5). La Fig. 4 propone l'installazione della libreria per l'applicazione SigFox. Il prezzo della scheda MKRFox1200 include un abbonamento

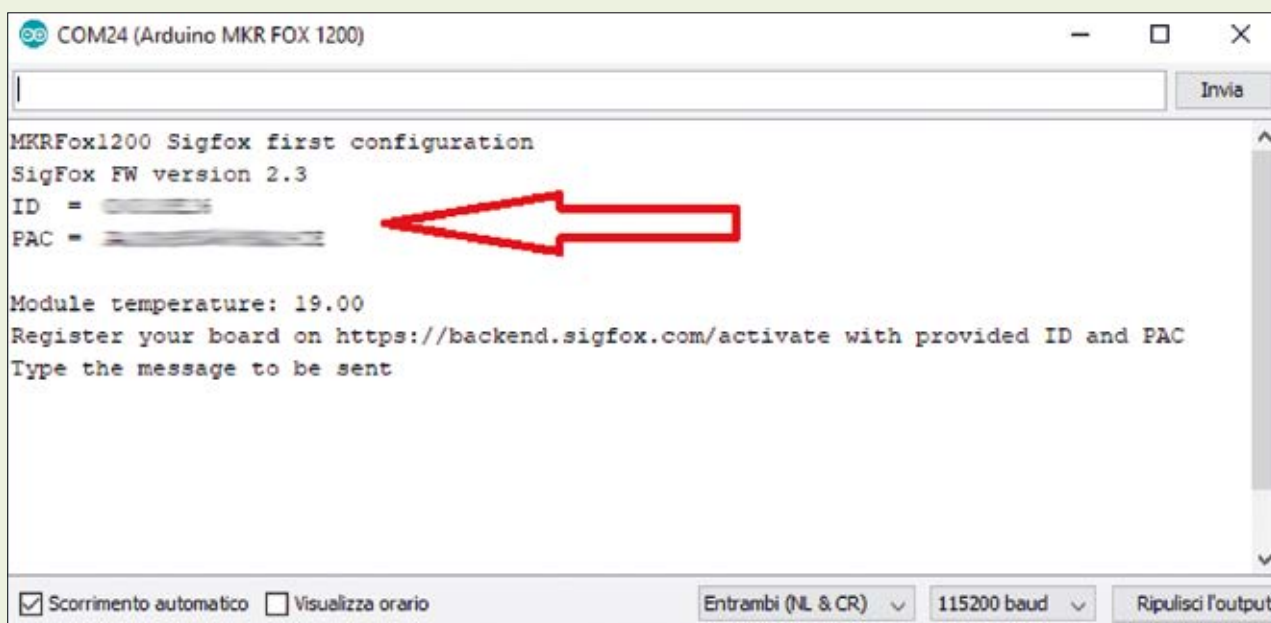


Fig. 5 Schermata di Serial monitor relativa all'esempio FirstConfiguration.ino

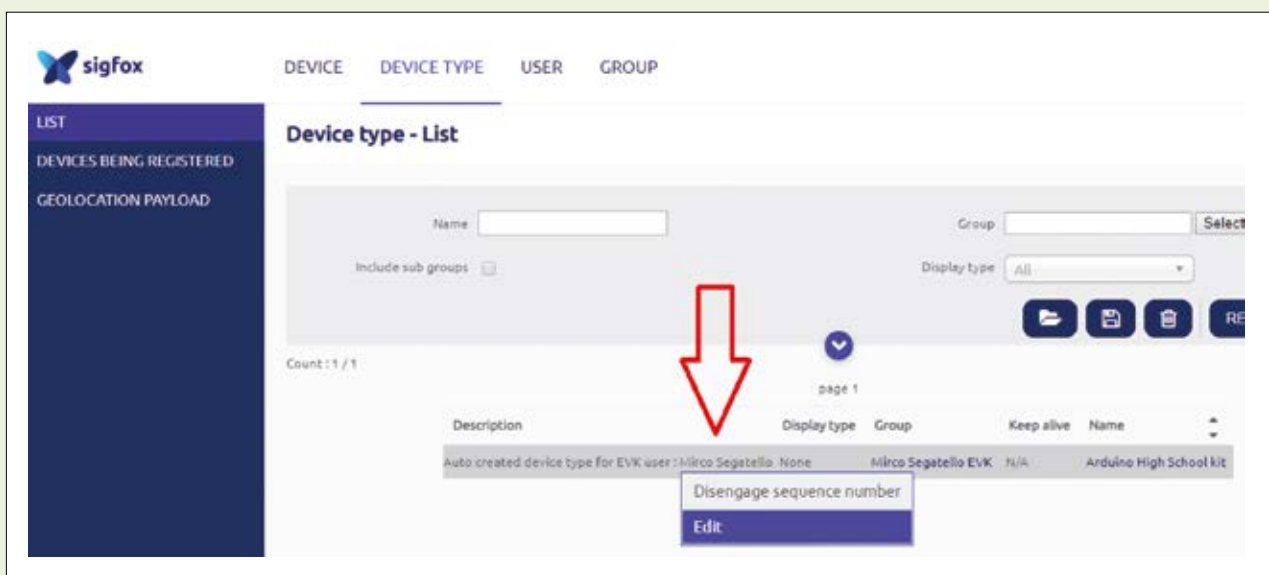


Fig. 6 Apertura del menu per la configurazione della scheda.

alla rete SigFox per un anno. Il piano verrà attivato automaticamente dopo l'invio del quarto messaggio, con il limite di 140 messaggi al giorno. Affinché la scheda MKR1200 possa accedere alla rete SigFox è necessario attivarla tramite il portale

<https://buy.sigfox.com/activate>. Il primo passo è selezionare il paese (Italia), il secondo è selezionare il tipo di hardware (Arduino), successivamente la procedura chiede di specificare a quale provider volete connettervi (Nettrotter). Nei passi successivi dovrete inserire i dati della scheda (ID e PAC) e i

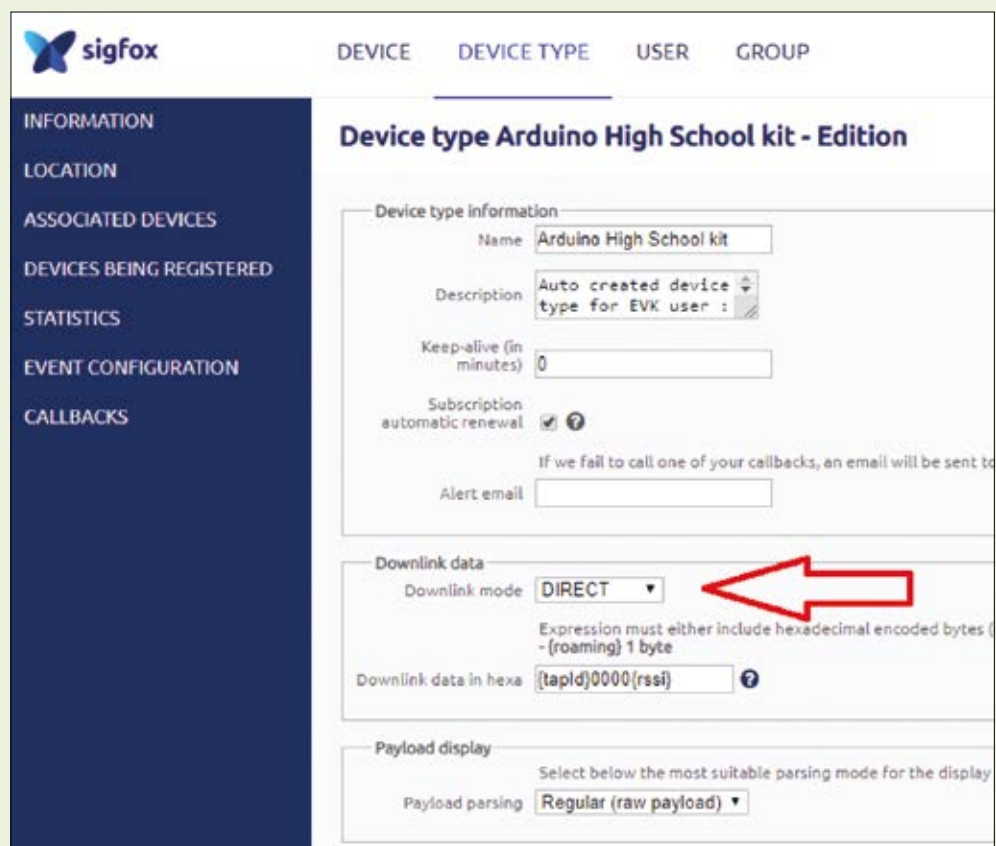


Fig. 7 Configurazione della scheda.

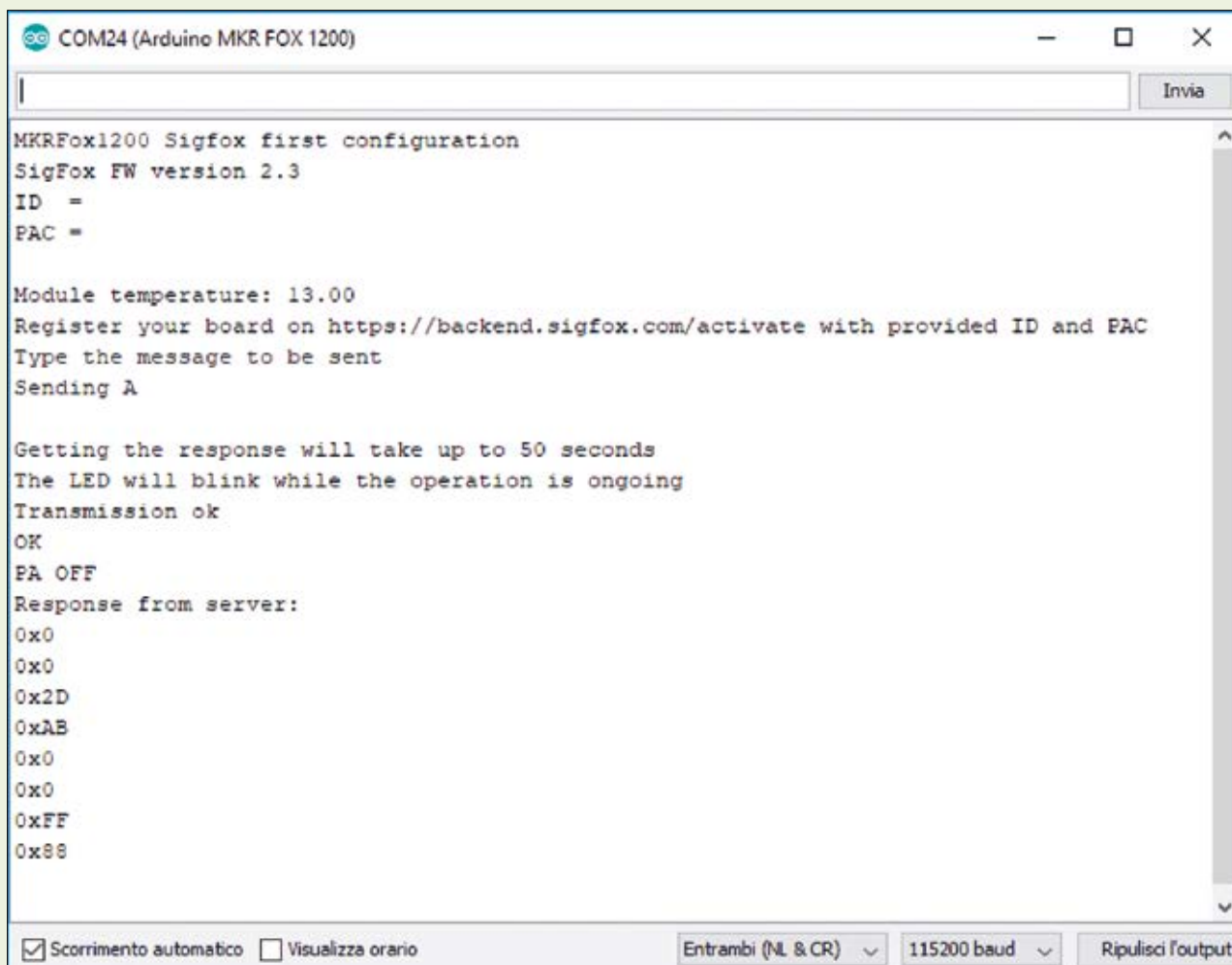


Fig. 8 - Schermata di Serial monitor con invio andato a buon fine

vostri dati personali, a procedura conclusa riceverete una mail di conferma che contiene un link che vi condurrà all'in-

serimento di una password per l'accesso al portale; conclusa la registrazione vi ritroverete una schermata come quella

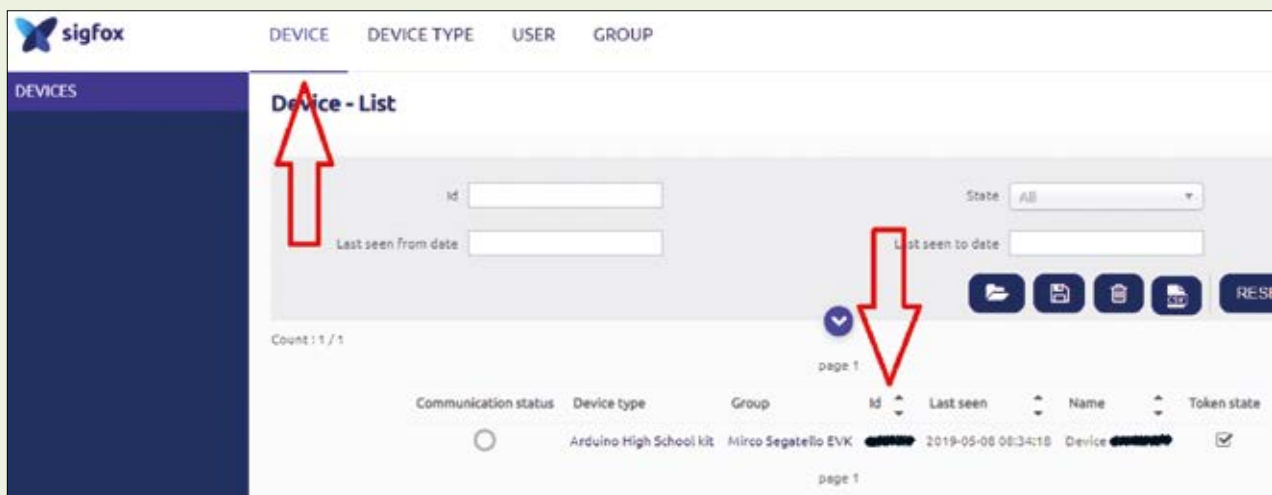


Fig. 9 - Selezione dei dispositivi dal portale SigFox.

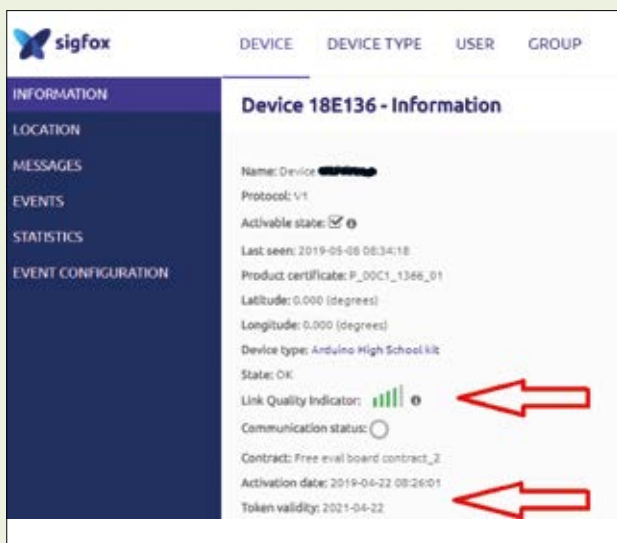


Fig. 10 Schermata con le informazioni della scheda

visibile in **Fig. 6**.

Il menu in alto permette di accedere alle informazioni delle vostre schede associate e ai dati del vostro account: se cliccate su Device type e successivamente sulla vostra scheda (sotto la voce description), si aprirà un menu a tendina e selezionando il comando Edit aprirete il menu di configurazione della scheda (**Fig. 7**). Da questa schermata potete modificare il nome associato alla scheda, l'indirizzo email per la funzionalità Call Back (esportazione dei dati verso altri portali) ma soprattutto potremmo abilitare la risposta automatica ai messaggi ricevuti, permettendo di avere conferma di ricezione; alla voce "download link mode" selezioniamo DIRECT. Sempre da questo menu, è possibile avere la statistica dei messaggi cliccando sulla voce STATISTICS.

Adesso è tutto pronto per la nostra prima comunicazione, quindi avviamo nuovamente lo sketch *FirstConfiguration.ino* e questa volta inseriamo il messaggio da inviare, nel nostro caso il semplice carattere **A** maiuscolo. Ricordiamo che SigFox limita fortemente il transito dei dati per cui è sempre necessario ridurre al massimo i bytes inviati.

La schermata di risposta fornita da Serial Monitor di Arduino è visibile nella **Fig 8**; nel caso la risposta dal server tardasse ad arrivare Serial Monitor ne darà notifica consigliando di trovare una posizione della scheda con più segnale. I dati di risposta contengono la data, l'ora e l'intensità del segnale

page 1

Time	Data / Decoding	LQI	Callbacks	Location
2019-05-08 08:33:41	41			
2019-05-08 07:55:45	7177			

Fig. 11 - Elenco dei messaggi ricevuto dal portatore SigFox.

Listato 1

```
int sensorValue;
#include <SigFox.h> // inclusione libreria sigfox
#define INTERVAL 10 //intervallo di tempo dei

void setup()
{
  Serial.begin(115200); // abilita la seriale per il debug
  while (!Serial) {}; // aspetta che la seriale sia avviata

  if (!SigFox.begin()) // verifica la presenza del modem sigfox
  {
    Serial.println("Shield error or not present!");
    return;
  }

  SigFox.debug(); // Abilita il debug del LED
  SigFox.reset(); //resetta il modem e la libreria
  delay(100);
  SigFox.end(); // pone il modem in modalità di basso consumo
}

void loop()
{
  SigFox.begin(); // avvia il modulo
  delay(100); // attende che il modulo sia inizializzato

  // legge la temperatura interna del modem
  int8_t value = (int8_t)SigFox.internalTemperature();
  Serial.print("Temp=");
  Serial.println(value); // visualizzo il valore acquisito

  SigFox.beginPacket(); // prepara il modem per l'invio
  SigFox.write(value); // invia il valore

  // chiude la comunicazione e verifica la corretta trasmissione
  int ret = SigFox.endPacket(false);
  if (ret > 0) {
    Serial.println("No transmission");
  } else {
    Serial.println("Transmission ok");
  }
  delay(100);
  SigFox.end(); // pone il modem in modalità di basso consumo
  delay(100);
  Serial.println("Attendo...\n");
  delay(INTERVAL*60*1000);
}
```

(codificati in esadecimale), così come specificato nella voce download data. Noterete che la temperatura del modulo è di 13°C, questo perché è stato posizionato all'esterno dell'abitazione al fine di garantire una buona intensità del segnale, non sempre garantita in ambienti interni.

Dal portale SigFox cliccate sul menu DEVICES e, nella lista dei dispositivi che vi viene presentata, cliccate sull'ID della vostra scheda (**Fig. 9**): a questo punto si aprirà un nuovo menu visibile in **Fig. 10** con alcuni dati (compresa la data di scadenza del contratto); a sinistra selezionate MESSAGE per visualizzare i dati ricevuti.

Come potete vedere dalla **Fig. 11**, nella schermata corrispondente vengono riportati data e ora del messaggio, il nostro carattere inviato in formato ASCII codificato in esadecimale e l'intensità del segnale radio.

Fig. 12 - Impostazione dei parametri in DEVICE TYPE.

COSTRUIAMO LA NOSTRA APPLICAZIONE CON SIGFOX

Vediamo ora come possiamo realizzare un'applicazione sfruttando la rete SigFox, tenendo anche in considerazione il fatto che l'invio di dati verso la nostra scheda è limitata a circa un messaggio ogni dieci minuti (almeno con il contratto base). Per prima cosa reimpostiamo la comunicazione in modalità unidirezionale in modo da evitare di dover lasciare attiva la scheda per ricevere il messaggio di ritorno (che può impiegare anche 50 secondi) e successivamente impostiamo il dato di arrivo di tipo intero, visualizzabile come temperatura. Dalla schermata principale clicchiamo su DEVICE TYPE e poi sopra il nome della scheda, nel menu che si aprirà clicchiamo su edit ed impostiamo i parametri come visibile nella **Fig. 12**. Non ci resta che caricare lo sketch identificato dal nome *MRKrfx1200_TempMonitor.ino* (**Listato 1**) per iniziare l'invio dei dati. Vediamo brevemente, esaminando il **Listato 1**,

quali sono i punti salienti di questo programma: la prima riga riguarda l'inclusione della libreria SigFox mentre la seconda specifica, tramite il parametro INTERVAL, ogni quanti minuti sarà inviato il dato. Nella sezione setup viene semplicemente configurata la comunicazione con il PC e verificato il corretto funzionamento del modem.

La parte più importante, quella che gestisce l'invio del dato, è tutta contenuta nel loop, semplificato per garantirne la massima chiarezza. In questa parte si abilita il modem e si legge la temperatura fornita dello stesso (ha un sensore interno) che successivamente viene inviata a Serial Monitor per la visualizzazione. Ovviamente in questa sezione potrete inserire la lettura di qualsiasi altro valore, fermo restando che sarà un valore ad otto bit. Le righe successive si occupano di inviare il pacchetto di dati e chiudere la comunicazione, viene anche verificata se la trasmissione è andata a buon fine ma non viene controllata la corretta ricezione dei dati trasmessi, in quanto abbiamo volutamente disabilitato dal portale l'invio della ricevuta di ritorno. Esiste la possibilità di implementare la funzionalità di deep sleep della scheda e del modulo al fine di ridurre drasticamente il consumo di corrente, funzione che però in alcuni PC, interrompe definitivamente la funzionalità di invio dati a Serial Monitor. In questa prima fase ci sembra utile mantenere attivi i messaggi di debug al fine di verificare che tutto funzioni correttamente.

Cliccando su MESSAGE nel portale SigFox potrete vedere i dati ricevuti (**Fig. 13**) e cliccando sul simbolo a goccia rovesciata di colore blu potete visualizzare una mappa con la posizione della vostra scheda. Siccome le coordinate sono determinate per triangolazione con le varie base station conoscendo solo il segnale RSSI, l'area a volte è troppo vasta per poter essere usata come localizzatore.

CONCLUSIONI

Con la descrizione del sistema SigFox concludiamo questo corso dedicato all'IoT. Prossimamente pubblicheremo progetti applicativi sia con la scheda MKR1200 per la rete SigFox, sia con hardware per LoRa. A presto!

Time	Data / Decoding	LQI	Callbacks	Location
2019-05-09 15:49:57	16 Temperatura: 22			
2019-05-09 15:39:46	13 Temperatura: 19			
2019-05-09 15:29:34	12 Temperatura: 18			

Fig. 13
Letture dei messaggi ricevuti con l'esempio del listato 1