



UNIVERSITÀ DEGLI STUDI DI UDINE

Corso di Laurea in Ingegneria elettronica

Dipartimento di (DPIA) Dipartimento Politecnico di
Ingegneria e Architettura

Tesi di Laurea

LORA AND IOT

Relatore:

Prof. Antonio Abramo

Laureando:

Enrico Tolotto

Correlatore:

Prof.

Anno Accademico 2016/2017.

Sommario

La richiesta di device dotati di una connessione wireless è in continuo aumento, seguendo il trend del *Internet delle cose*(IoT). Questa continua crescita ha portato alla creazione di nuove tecnologie, le quali sono in competizione per aggiudicarsi la maggioranza del mercato. In questo campo le cosiddette (LPWAN) *Low Power Wide Area Networks* sono in forte aumento, grazie alla loro connettività a lungo raggio sfruttando bande d frequenza libere. Questa tesi si focalizzerà sulla tecnologia LoRaTM, implementata attraverso l'utilizzo del framework open-source KuraTM sviluppato da EurotechTM.

Note

■ Scrivi qualcosa di decente qui	1
■ Completare e riscrivere	6
■ Trovare termine per multipath	7
■ Modificare e riscrivere	7
■ Riscrivere	7
■ Rivedere i vari punti e cambiare il linguaggio	9
■ Aggiungere immagine e finire la spiegazione	10
■ Riguardare ultimo punto della documentazione pagina 14	11
■ Chiedere se il termine protocollo è appropriato	11
■ Aggiungere citazione	16
■ Aggiunger link	21

Indice

Indice	v
Elenco delle figure	vii
0.1 IoT	1
1 LPWAN e Lora	5
1.1 LPWAN	5
1.2 LoRa	6
1.3 CSS	6
1.4 LoRaWAN	11
1.5 Frequenze	13
1.6 LPWAN	15
1.6.1 NB-IoT	16
1.6.2 NB-IoT e LTE-M	16
1.6.3 Sigfox	17
1.7 Implementazione	19
1.7.1 SX1301	19
1.7.2 ReliaGATE	20
1.7.3 Architettura del software	20
1.7.4 Semtech packet forwarder	21
1.7.5 LoRa Gateway Bridge	23
1.7.6 ESF	23
1.7.7 Realizzazione	24
1.7.8 Mqtt Bridge Config	25
1.7.9 Misurazioni	26
Bibliografia	31

Elenco delle figure

1	Numero di dispositivi per anno	1
2	Numero di dispositivi per anno	3
1.1	Comparazione tra UNB e SSP	6
1.3	Comparazione simbolica dei vari SF	9
1.4	Struttura pacchetto Lora	10
1.5	Pacchetto codificato dal layer fisico	10
1.6	Struttura interna ricevitore SX1301	11
1.7	Struttura rete a stella LPWAN	12
1.8	Stack del protocollo della rete LoRaWAN	13
1.9	Comparazione tipologia di reti	15
1.10	Modalità di funzionamento NB-IoT	16
1.11	Comparazione reti LPWAN	17
1.12	Caratteristiche elettriche SX1301	20
1.13	Architettura del software	21
1.14	Architettura del software	23
1.15	Architettura del software	26
1.16	Architettura del software	27
1.17	Architettura del software	28
1.18	Architettura del software	29

0.1 IoT

Sempre più spesso si parla di Internet delle cose, o IOT, con questo termine si intende un'evoluzione delle applicazioni legate al settore mobile, al settore della home automation e al settore embedded. In questo scenario ogni oggetto il quale contiene un sensore sarà connesso ad Internet. Avvalendosi di questa connessione, i vari dati raccolti potranno essere inviati nel cloud, dove verranno elaborati e resi disponibili alle varie applicazioni. Per fare in modo che questo update avvenga, è necessario riuscire a creare una rete di devices, correlati nelle loro funzionalità, i quali riescano a *parlare un linguaggio comune*. Il punto principale di questo upgrade sta nel riuscire a creare una rete di devices connessi ad internet. Per supportare e utilizzare una potenza di calcolo maggiore andando a combinare tecniche di data analytics per estrarre le informazioni più significative.

Scrivi qualcosa di decente qui

In questa visione, milioni di devices saranno connessi a Internet e molto presto milioni di milioni di devices.

Il mercato di questi *smart devices* è in rapida crescita con una stima di 8,3 miliardi di dispositivi connessi nel anno 2017, e di circa 20 miliardi per l'anno 2020 [1]. Andando ad creare un impatto economico compreso tra i 2.7 e i 14 trilioni di dollari. I mercati principali saranno quelli del health care con un introito compreso tra i 1.1 e i 2.5 trilioni di dollari e il settore industriale con 2.3 a 11.6 trilioni di dollari.

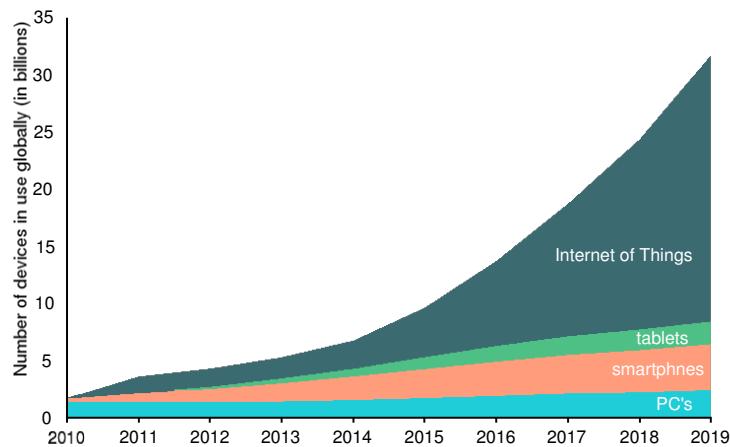


Figura 1: Numero di dispositivi per anno

Questa rapida crescita ha portato alla ricerca e sviluppo di nuove solu-

zioni tecnologiche per supportare il carico di dispositivi simultaneamente connessi alla rete, senza avere un degrado evidente delle performance. Per non alterare il *QoS* Quality of Service della rete , hardware e software dovranno essere rivisti insieme alla topologia di rete utilizzata. I principali punti chiave sono

- **Scalabilità:** Dato l'elevato numero di devices connessi, scenari urbani ed industriali, la network tecnologi alla base dovrà essere estremamente adattabile, in maniera dinamica, al carico di dispositivi connessi.
- **Costo unitario:** Il costo del singolo modulo, dovrà essere basso per garantire la più ampia fetta di mercato.
- **Durata della batteria:** La maggior parte dei dispositivi sarà alimentata tramite batteria, e la durata media è stimata di anni.
- **Costo computazionale:** La modulazione alla base di queste nuove tipologie di rete, dovrà essere concepita in modo da non avere un costo computazionale elevato.
- **Distanza:** Un altro punto fondamentale è la possibilità di avere comunicazioni a lunga distanza.
- **Sicurezza:** Ogni devices dovrà essere difficilmente penetrabile da attacchi esterni.
- **Management:** I vari dispositivi dovranno essere facilmente controllabili da remoto.
- **Fail-safe:** Il non funzionamento di devices non dovrà compromettere l'intera infrastruttura a lui connessa.

Vari tipi di architetture di rete sono stati proposti per realizzare questa nuova infrastruttura. Per quanto le varie proposte si basano su tecnologie differenti, è possibile individuare tre layer comuni

- **Device layer** formato da tutti i dispositivi che collezionano dati e sono connessi alla rete.
- **Network layer** La struttura della rete, la quale permette di connettere i vari devices in modo che possano scambiare i dati tra di loro o inviarli ad un data-center.

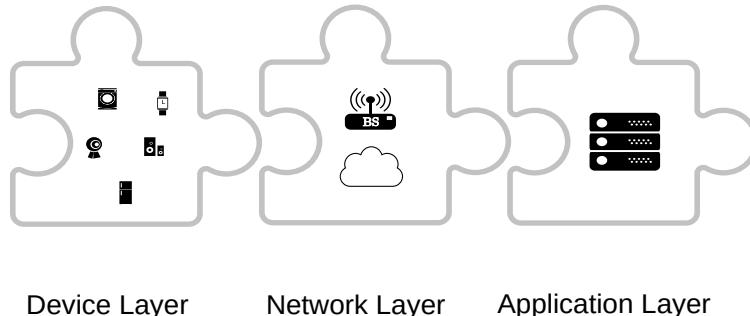


Figura 2: Numero di dispositivi per anno

- **Application layer** il quale interpreta e utilizza i dati ricevuti.

Attualmente sono diversi i concorrenti che provano ad affermarsi nel settore del IoT proponendo soluzioni diverse. Nei seguenti capitoli ci sarà un analisi generale delle varie topologie proposte, in particolare verrà analizzata la tecnologia Lora ed il protocollo LoraWAN.

Capitolo 1

LPWAN e Lora

Nel seguente capitolo si approfondirà il concetto di reti *LPWAN* e la loro struttura. Si analizzeranno i principali vantaggi che queste reti portano e i vari use-case, focalizzando l'attenzione sulla tecnologia Lora sviluppata da Semtech.

1.1 LPWAN

Tra le varie tipologie di rete emergenti per l'IoT, LPWAN sta riscuotendo sempre più interesse. Questo tipo di reti si basano su una topologia a stella, la quale permette di avere un elevato numero di devices connessi ad una sola stazione base. Ideate per comunicazioni a lungo raggio , risultano ideali per i vari use-case del internet of things quali smart-city, agricoltura, assistenza sanitaria ecc... I principali concorrenti che implementano queste tecnologie sono NWave,NB-IoT, SigfoxTM e SemtechTM possessore di LoraTM. Tra i principali competitors, Sigfox e Lora sono le tecnologie maggiormente utilizzate. L'implementazione proposta da Sigfox utilizza un tipo di comunicazione basata sulle frequenze ISM e modulazione Ultra Narrow Band. Questa tecnologia è capace di inviare messaggi con payload lungo 12 byte in 6 secondi usando una frequenza di 100[Hz]. Per via della modulazione scelta e le varie regolazioni delle frequenze ISM, utilizzando la tecnologia Sigfox si ha un numero limitato di 140 messaggi per giorno. Al contrario la tecnologia Lora, implementa una modulazione basata su uno spettro diffuso il quale permette una maggiore ricezione andando ad influire sul data-rate possibile

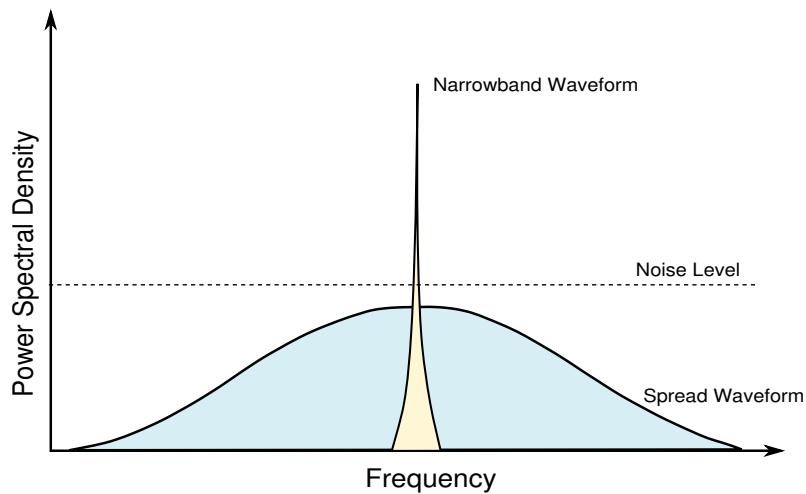


Figura 1.1: Comparazione tra UNB e SSP

1.2 LoRa

Lora è una tecnologia di modulazione wireless semi-proprietaria sviluppata da Semtech. Lora è composta da un layer fisico ,proprietario, che prende il nome di *Lora*[2] , e una parte libera chiamata LoRaWAN[3], la quale definisce un protocollo di comunicazione, il quale usa LoRa come layer fisico. I punti chiave dei questa tecnologia, sono il grande raggio di copertura e il basso consumo energetico per effettuare uno scambio dati. Per le comunicazioni , vengone utilizzata la banda ISM, grazie alle quale è possibili costrurire una rete ,che implementa questa tecnologia, senza possedere alcuna licenza.

Completare e riscrivere

1.3 CSS

Alla base del layer fisico troviamo la modulazione Chirp Spread Spectrum (CSS). Questo tipo di modulazione, ottenuta facendo variare in modo lineare, la frequenza di un segnale sinusoidale di durata finita, chiamato chirp. In questo modo è possibili distribuire l'informazione su di un spettro maggiore rispetto a quello normalmente utilizzato. In questo modo è possibile ottenere numerosi vantaggi quali

- Uno spettro idealmente rettangolare, il quale utilizza tutta la capacità del canale e fornisce un ottima densità spettrale di potenza rispetto agli altri tipi di trasmissione.
- **Segnali di tipo Chirp** possono essere sovrapposti in modo tale da poter variare il data-rate e l'energia per bit in modo adattativo per aumentare l'efficienza complessiva.
- **Hanno guadagno programmabile**, il quale permette di raggiungere distanze considerevoli mantenendo un buon SNR .
- **Ottima risoluzione nel asse del tempo**, quindi ottimi per coprire lunghe distanze.
- **Immuni al effetto Doppler**
- **Immuni alle degenerazioni per effetto di multipath**

Trovare termine per multipath

Un segnale di tipo *Chirp* assume valori compresi nella banda di frequenza $B = [f_0, f_1]$, il suo andamento è di tipo monotono , crescente o decrescente compreso tra le due frequenze f_0 e f_1 . Quindi possiamo parlare di *Up-chirp* o di *down-chirp*.

Uno degli aspetti principali del layer fisico, è la possibilità di adattare il numero di bit codificati in un simbolo in base alle varie esigenze. Questo parametro che prende il nome di *Spread Factor*, permette a parità di potenza impiegata nella trasmissione, di ottenere distanze maggiori andando ad aumentare il tempo totale di trasmissione secondo la seguente formula: In questo modo è possibile, inviare segnali a distanze maggiori, utilizzando la stessa potenza trasmisiva.

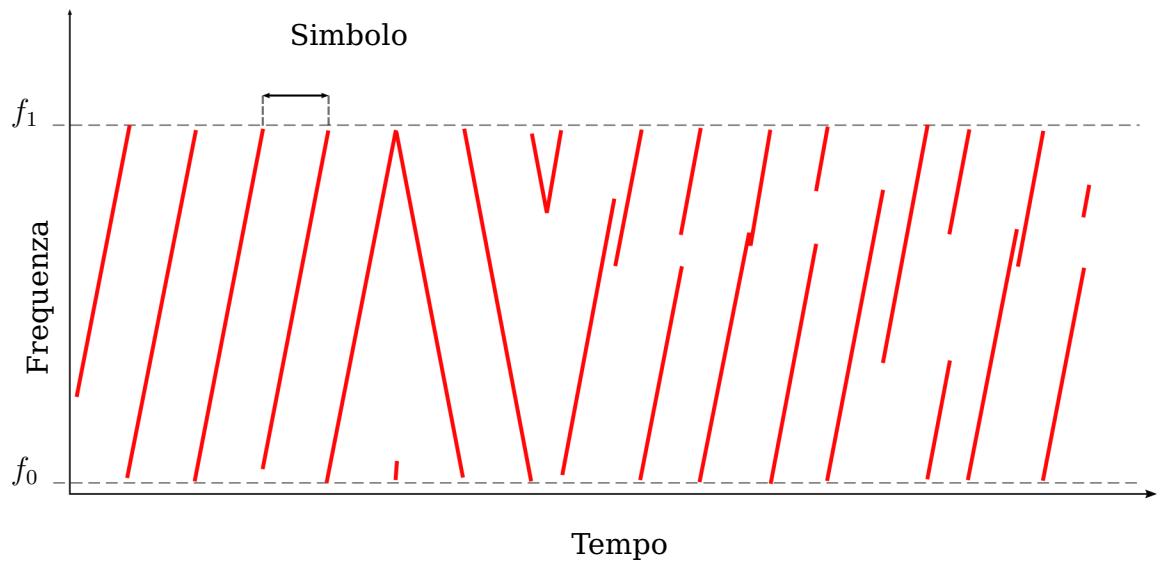
Modificare e riscrivere

— Questa possibilità di adattamento permette a parità di potenza di riuscire a raggiungere distanze maggiori andando a variare, quello che nella documentazione ufficiale è chiamato *Spread Factor*. Tutto ciò significa che (SF) rappresenta 2^{SF} bits in un simbolo. Un differente SF implica anche un differente tempo di comunicazione secondo la formula

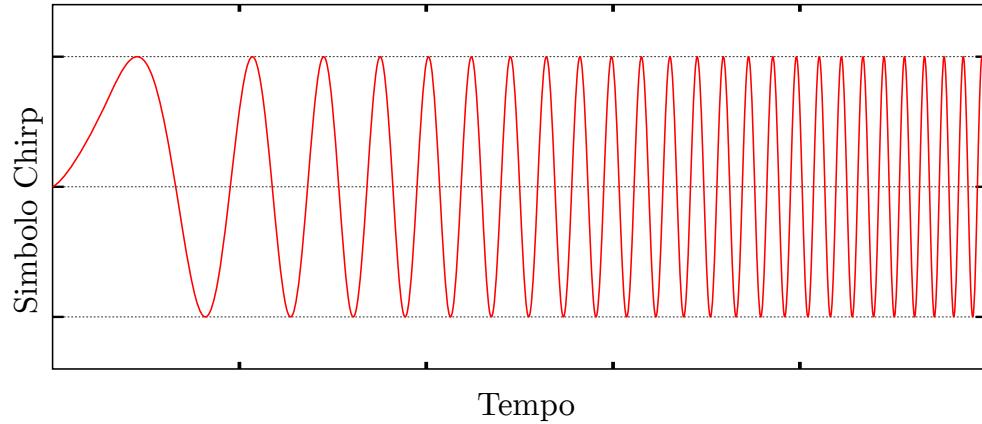
$$T_s = \frac{2^{SF}}{B}. \quad (1.1)$$

Riscrivere

Dalla quale si evince che andando ad aumentare lo spread factor di una unità, mantenendo una lunghezza di banda fissa B , otteniamo un raddoppio nel tempo di trasmissione. Il fatto di avere messaggi più



(a) Segnale Chirp nel dominio della frequenza



(b) Simbolo codificato col metodo Chirp nel dominio del tempo

lunghi, conferisce un robustezza superiore alle interferenze e al rumore. In discapito a tutto ciò, il fatto di dover codificare il messaggio con un maggiore numero di simboli, aumenta la possibilità di errore alla ricezione.

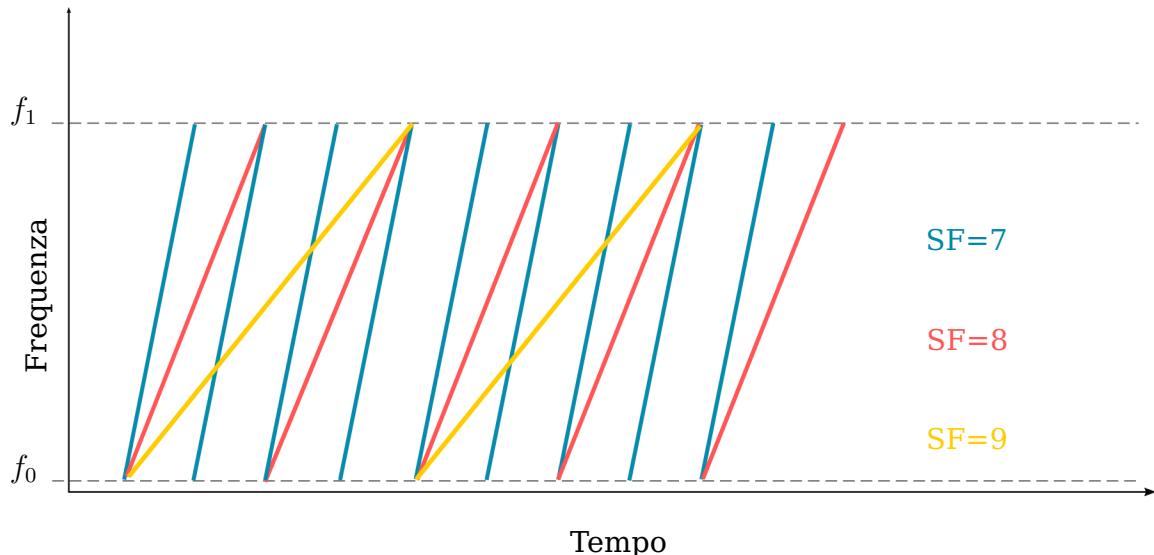


Figura 1.3: Comparazione simbolica dei vari SF

Questa nuovo modo di trasmettere i vari dati porta con se molti vantaggi.

- La modulazione Lora è semplice da implementare nei dispositivi, quindi i moduli radio al loro interno saranno economici.
- Resistente alle interferenze in banda e fuori banda.
- Resistente all'effetto Doppler, in questo modo è possibili utilizzare cristalli non molto accurati al interno dei devices, in modo tale da abbattere i costi di produzione.
- Il modulo di ricezione è altrettanto semplice da costruire, quindi non molto costoso.

Analizzando lo spettrogramma di una comunicazione Lora è possibile distinguere in modo semplice le varie parti che compongono il pachetto trasmesso.

Rivedere i vari punti e cambiare il linguaggio

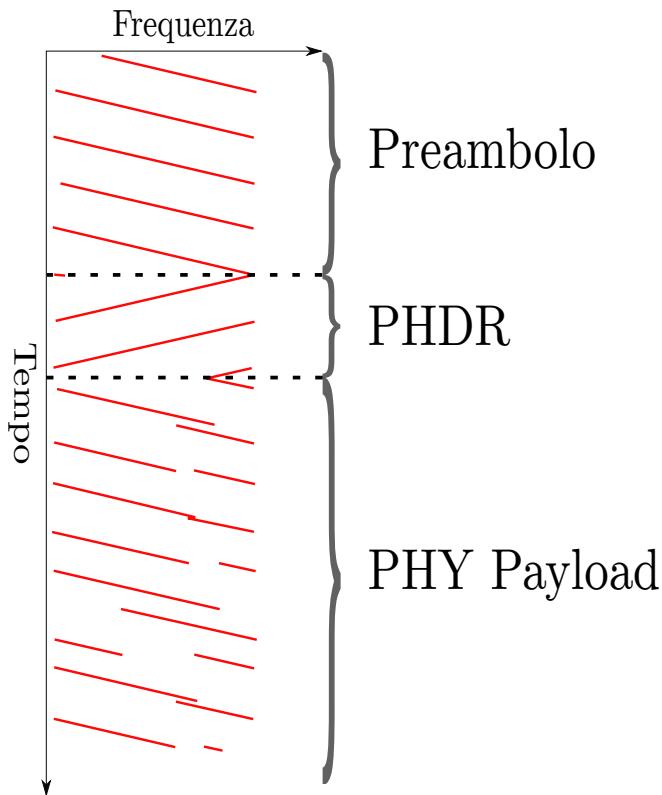


Figura 1.4: Struttura pacchetto Lora

Osservando l'immagine precedente è facile notare come è strutturato un pacchetto. La prima parte della trasmissione, nonché il preambolo stesso è codificato con una serie di *up-chirp*. In questo modo il gateway riesce a sintonizzarsi sulla stessa frequenza del dispositivo trasmittente. Successivamente vengono inviati una serie di *down-chirp* i quali rappresentano l'header codificato dal layer fisico, dove sono presenti dei bit di controllo e correzione degli errori. L'ultima parte rappresenta il payload codificato dal PHY, in questa parte, formata solo da *up-chirp* sono presenti dei salti, i quali sono un chiaro segnale della presenza di dati codificati.



Figura 1.5: Pacchetto codificato dal layer fisico

Aggiungere immagine e finire la spiegazione

Per quanto riguarda la struttura interna dei moduli radio, non si hanno molte informazioni dato che la tecnologia è proprietaria di Semtech. Nella documentazione ufficiale è presente una rappresentazione grafica dei vari blocchi presenti al interno dei moduli radio.

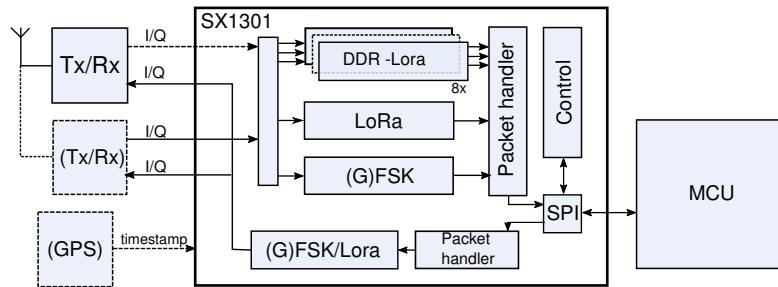


Figura 1.6: Struttura interna ricevitore SX1301

Dalla figura 1.6 è evidente che il gateway rimane in ascolto su 8 frequenze diverse, le quali permettono di coprire tutti i vari SF. Tutto ciò è possibile anche grazie al fatto che i vari SF sono quasi ortogonali fra di loro, perciò il ricevitore è in grado di ricevere pacchetti da SF diversi contemporaneamente. Questo tipo di ricevitore può demodulare fino ad un massimo di 8 pacchetti contemporaneamente. Inoltre questa topologia permette di avere vari vantaggi

- I vari nodi della rete, possono cambiare frequenza in ogni trasmissione in modo casuale, andando a migliorare di molto la robustezza del sistema alle varie interferenze.
- Non è necessario avere tabelle contenenti informazioni riguardanti il data-rate dei vari nodi. Ogni data-rate viene demodulato contemporaneamente.
- È possibile utilizzare più antenne nel gateway per realizzare il cosiddetto true antenna diversity, per aumentare la robustezza al multi-path

Riguardare ultimo punto della documentazione pagina 14

1.4 LoRaWAN

Chiedere se il termine protocollo è appropriato

La parte non proprietaria del protocollo è chiamata *LoRaWAN*, in essa viene descritta la topologia di rete, la struttura dei pacchetti e le varie classi di device possibili.

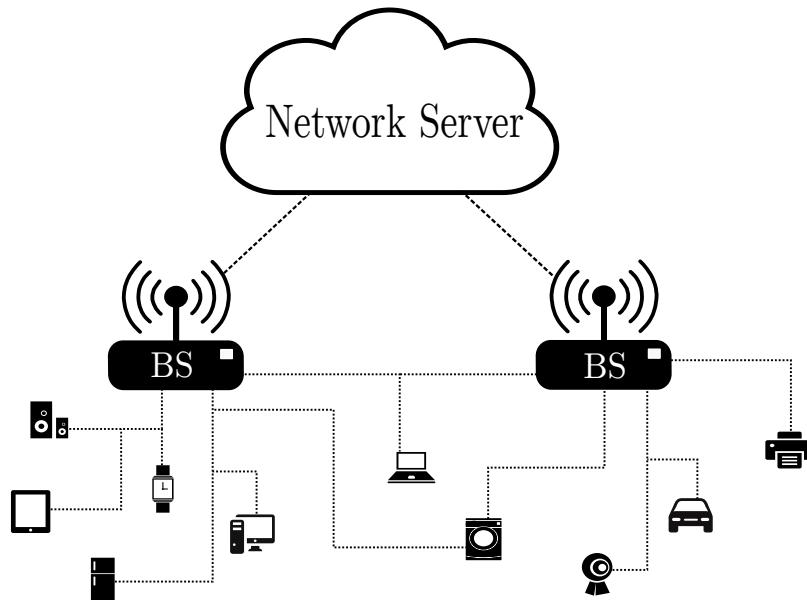


Figura 1.7: Struttura rete a stella LPWAN

La topologia di rete utilizzata, è una topologia a stella, nella quale molti dispositivi sono connessi e comunicano con uno o più base station. Le BS non sono altro che dei ponti per poter trasmettere i messaggi ricevuti dai vari devices all Network Server, tramite una connessione ethernet, 3G o 2G. Per come è strutturata la rete, un messaggio inviato da un singolo device, può essere ricevuto e inoltrato da più BS all Network Server.

Il NS ha il compito di interpretare e scartare i vari messaggi duplicati che arrivano, selezionare la BS più adatta per inviare il messaggio di downlink creando un database di tutti i vari devices presenti nella rete. Nella figura 1.8 è rappresentato lo stack del protocollo degli end-devices, gateway e network-server.

Esistono tre classi di devices, le quali specificano i vari use-case per possibili.

- **Class A** è la modalità di funzionamento predefinita. In questa modalità il device si occupa solo di trasmettere i vari messaggi in maniera completamente asincrona. Eseguita la trasmissione, due finestre di

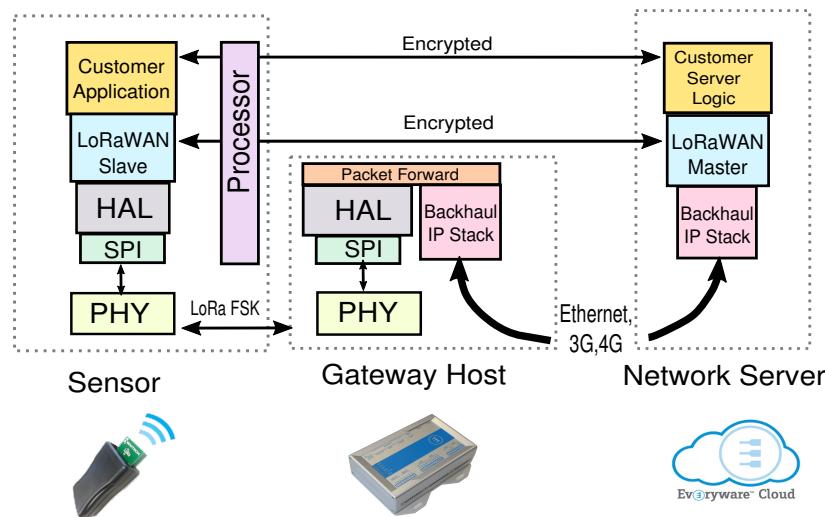


Figura 1.8: Stack del protocollo della rete LoRaWAN

ascolto vengono aperte nel end-device. La prima finestra rimane in ascolto nella stessa frequenza in cui il dato è stato comunicato, mentre la seconda rimane in ascolto su una frequenza nota a priori e comunicata tramite il MAC.

- **Class B** sono devices i quali sono sincronizzati con il NS tramite beacon packets. In questo modo hanno la possibilità di ricevere dati in un determinata finestra di tempo. In questa classe rientrano interruttori , attuatori ecc..
- **Class C** è riservata ai devices che hanno la possibilità di essere alimentati direttamente dalla rete elettrica, quindi possono mantenere il ricevitore costantemente in ascolto.

1.5 Frequenze

A seconda della regione in cui operano i devices, lo standard LoRaWAN definisce quali frequenze vengono occupate in base alle regolamentazione vigente in Europa Cina e Stati Uniti. Per ognuna di queste regioni, vengono definite la struttura del payload, la frequenza, lo spreading factor, e la massima lunghezza possibile del payload. Inoltre tutte le fre-

Stato	Frequenza [MHz]
Europa	868-870
US	902-928
China	779-787

Tabella 1.1: Bande di frequenza per le varie regioni

quenze sono comprese nelle bade ISM, e non sono particolarmente elevate, rendendole preferibili per le lunghe distanze rispetto a frequenze quali 2.4 [GHz] e 5[GHz]

1.6 LPWAN

Il principale problema che si presenta nella ottica del IoT è avere una infrastruttura di rete capace di gestire il traffico di milioni di dispositivi contemporaneamente connessi. Fino ad oggi la principale tecnologia wireless usata nelle comunicazioni M2M è stata la rete cellulare, la quale copre la quasi totalità di tutte le aree geografiche. La scelta della rete 2G, 3G, 4G è svantaggiosa poiché offre data-rate molto maggiore rispetto a quello normalmente utilizzato in applicazioni IoT, comportando l'uso di moduli sovradimensionati e andando ad aumentare di molto il prezzo per unità. Inoltre l'elevato consumo energetico e il prezzo svantaggioso degli abbonamenti offerti dagli operatori telefonici, ha portato alla ricerca di nuovi standard. Per colmare il gap tra tecnologie esistenti e la necessità di connettere milioni di devices diversi sono nate le LPWAN *Low power wide area network*. Con questo termine, si identificano tutte quelle reti ideate appositamente per l'IoT, le quali garantiscono una ampia raggio di azione, andando a sacrificare il bit rate e la complessità dei moduli radio.

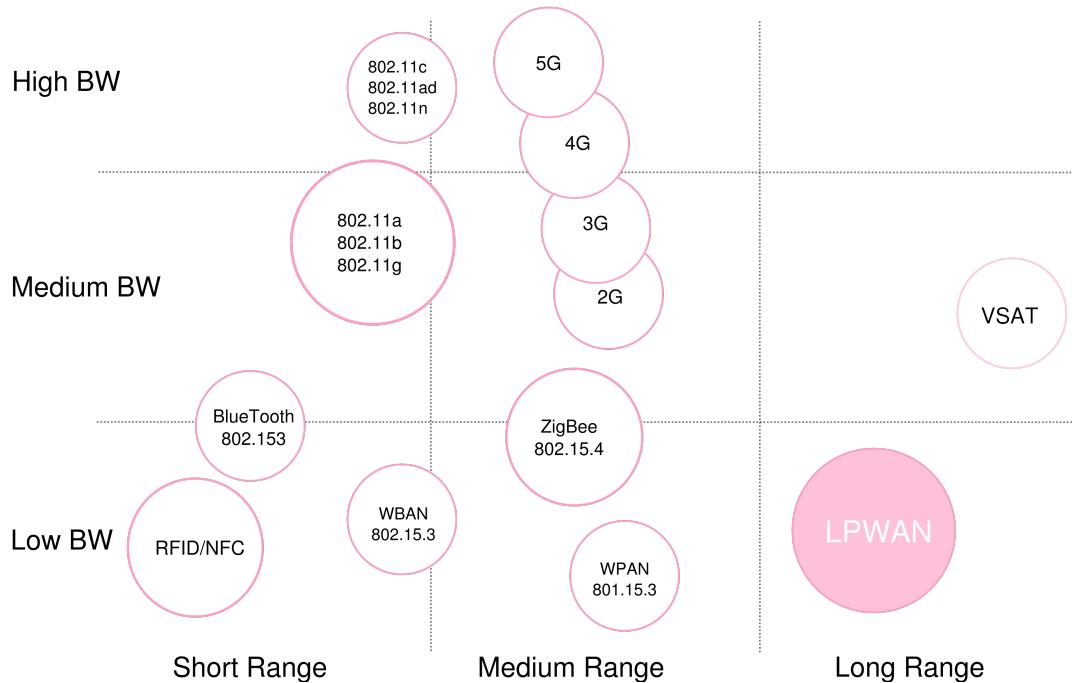


Figura 1.9: Comparazione tipologia di reti

In questo contesto, i maggiori competitor sono Lora, Sigfox, NB-IoT e LTE-M. Anche se ognuna di queste soluzioni punta a ottenere la leader-

ship del mercato, esse implementano soluzioni tecniche molto diverse tra di loro, ognuna delle quali ha vantaggi e svantaggi.

1.6.1 NB-IoT

NB-IoT, o LTE Cat NB1 è un nuovo standard proposto da Huawei, Ericsson, Qualcomm, e Vodafone. Utilizzando frequenze licenziate, NB-IoT è in grado di offrire tre diversi scenari di sviluppo, *standalone*, *guard band*, *in band*

Aggiungere citazione

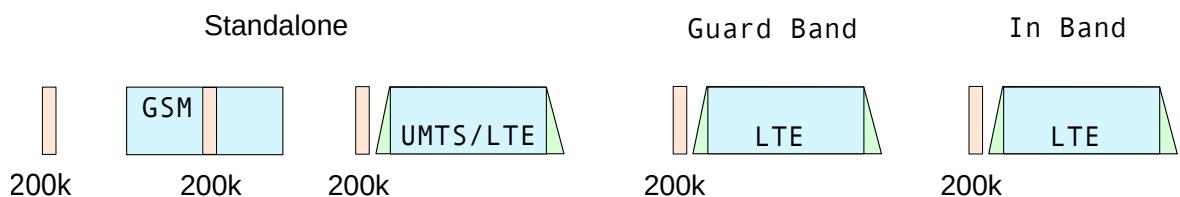


Figura 1.10: Modalità di funzionamento NB-IoT

Così facendo operatori di rete i quali possiedono ancora l'infrastruttura gsm, possono andare a riutilizzarla , diversamente è possibile andare ad adattare le cellule 4G per renderle compatibili con questa tecnologia. Quello che si prefigge NB-IoT, è riuscire a garantire la sicurezza delle comunicazioni cellulari, offrendo un basso costo di sviluppo dal lato hardware, una lunga durata della batteria, limitando il thruput massimo dei devices.

1.6.2 NB-IoT e LTE-M

Simile ad NB-IoT, LTE-M o LTE Cat-M1 è uno standard che opera su frequenze licenziate , della rete 4G. Il vero punto di forza di questa tecnologia è il poter utilizzare le celle 4G senza dover apportare modifiche, andando ad offrire una maggiore thruput, a discapito dei consumi energetici. Per questo motivo LTE-M è in rapida espansione.

1.11, queste due tecnologie offrono prestazioni diverse, e si impongono due use-case diversi. Quindi funzionano in uno spettro proprietario, e forniscono prestazioni diverse per i diversi use-case. In particolare devices che utilizzano NB-IoT hanno una durata della batteria molto più prolungata di quelli che utilizzano la connessione LTE-M, la quale permette un maggiore band-with a discapito delle durata. Il vantaggio di

questi standard e che non servono grandi cambiamenti alla rete telefonica, quindi sono facilmente implementabili dalle compagnie telefoniche. Inoltre, ogni device sarà dotato di una sim-card per poter connettersi, in questo modo è possibile ottenere una sicurezza simile a quella implementata nelle connessioni LTE. Altro punto a favore è il fatto che non è necessario utilizzare gateway, ogni device è connesso direttamente alla rete. La maggior parte delle compagnie telefoniche ha già dei piani per rendere disponibile queste due tecnologie.

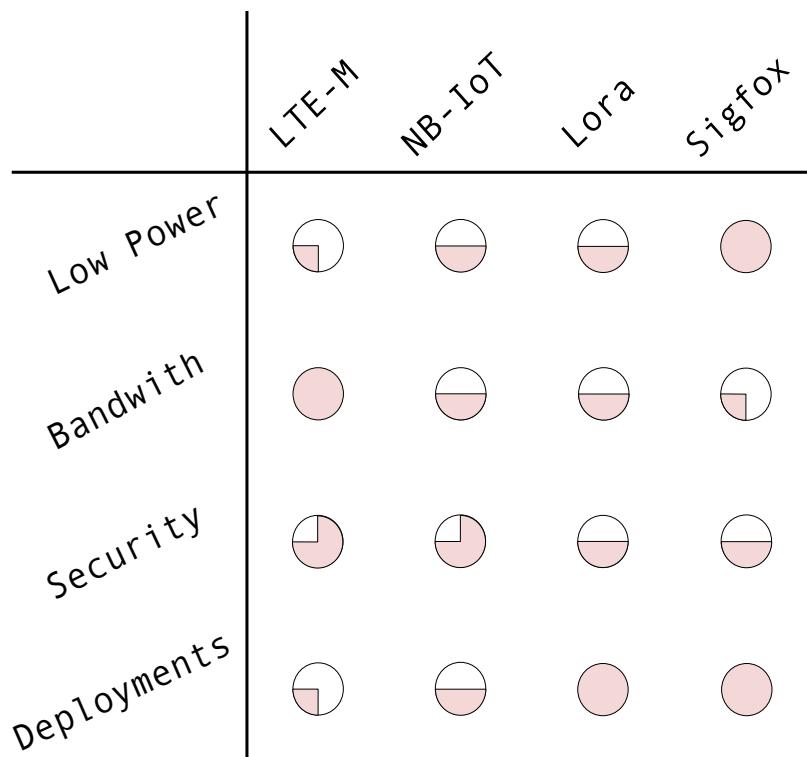


Figura 1.11: Comparazione reti LPWAN

1.6.3 Sigfox

Sigfox, azienda francese la quale dal 2015 ha iniziato a sviluppare una rete privata in Francia Spagna e Regno unito, promettendo di coprire più di 60 stati entro il 2020. Sigfox opera sulle frequenze ISM, quindi i costi delle infrastrutture e i costi che l'utente finale è tenuto a supportare sono molto inferiori rispetto alle soluzioni che si basano sulla rete cellulare. Il principale, vantaggio di avere una rete unica mondiale, è la possibilità di

un abbonamento unico e la continuità di funzionamento dei diversi devices in tutte le aree coperte. Utilizzando una modulazione *Ultra narrow band*, Sigfox, permette di coprire distanze pari a un paio di chilometri nelle zone urbane, fino a alcune decine di chilometri in zone rurali. Essendo Sigfox una tecnologia open hardware, è semplice per i produttori di moduli radio, adattare i loro componenti per renderli compatibili alla rete Sigfox. In questo modo il costo per unità diventa molto basso. Appoggiandosi a frequenze ISM, questa tecnologia è limitata la dimensione massima del payload per messaggio pari a 12 byte, con 140 messaggi inviabili al giorno per devices.

Per riuscire a gestire in maniera adeguata il sempre maggior numero di device connessi, e le diverse problematiche che essi comportano, è necessario andare a ridisegnare la topologia di rete per fare in modo che rispecchi i punti necessari per garantire il corretto funzionamento. Finora le principali tecnologie utilizzate erano la rete cellulare 2G, 3G, 4G, la quali hanno un'ampia copertura su tutto il territorio, ma devices che alimentati tramite batteria hanno durate massime di mesi. Oltre alla rete cellulare abbiamo gli standard IEEE 802.11, il comunemente chiamato wifi, il quale garantisce una copertura di poche decine di metri e favorisce una connessione veloce a discapito della durata della batteria. Oppure, Bluetooth e ZigBee i quali hanno offerto una connessione wireless con copertura di poche decine di metri. Oltre al fatto di una scarsa copertura e di un elevato consumo di energia per comunicazione, tutte queste tipologie di rete non sono scalabili cioè non sono ideate per supportare un carico di milioni di devices connessi contemporaneamente.

1.7 Implementazione

Utilizzando il modulo SX1301 in combinazione con il gateway prodotto da Eurotech ReliaGATE 10-11, si è andato a sviluppare un software in grado di ricevere i pacchetti inviati da dispositivi LoRa e inviarli ad un Broker Mqtt.

1.7.1 SX1301

Nella tabella 1.2 sono riportate le caratteristiche elettriche massime del chip SX1301. Il Chip, supporta tensioni di alimentazione fino a 4V e come è possibile osservare il range di temperatura in cui il chip può operare è molto ampio, rendendolo ideali per applicazioni esterne ed interne.

Parameter	Symbol	Conditions	Value
IO power supply to VSS	$V_{DDIO,ABSMAX}$		-0.5 V to 4.0 V
Core power supply to VSS	$V_{DDCORE,ABSMAX}$		-0.5 V to 2.0 V
Storage temperature	$T_{J,STORE}$		-50 °C to 150 °C
Junction temperature	$T_{J,ABSMAX}$		-40 °C to 125 °C
Pin voltage on IO and Clock pins	$V_{DPIN,ABSMAX}$		-0.3 V to $V_{DDIO} + 0.3$ V
Peak reflow temperature	T_{PKG}		260 °C
Latchup	I_{LUP}	JESD78D, class I	+/-100 mA
Humidity	H_R		0 – 95 %
ESD	HBM	Human Body Model JESD22-A114 CLASS 2	2 kV
	CDM	Charged Device Model JESD22-C101 CLASS III	300 V

Tabella 1.2: Caratteristiche elettriche SX1301

Utilizzando i valori nominali riportati nella tabella 1.3, si hanno valori di corrente pari a 1[uA] in idle e di 5[mA] in pieno funzionamento. Il chip

Parameter	Symbol	Conditions	Min	Typ	Max	Unit
Digital IO supply	V_{DDIO}	Operating Conditions for Electrical Specification	3.0		3.6	V
Digital core supply	V_{DDCORE}	Operating Conditions for Electrical Specification	1.75		1.85	V
Ambient operating temperature	T_A	With chip paddle soldered to PCB ground plan with minimum 100 cm ² air exposed area and heat sink	-40		85	°C

Tabella 1.3: Caratteristiche elettriche SX1301

è equipaggiato con un connettore sma al quale è collegata una antenna

omindirezionale, ideata per la frequenza 868[MHz] con un guadagno pari a 3dB.

1.7.2 ReliaGATE

Il gateway a cui è collegato il modulo SX1301 è il ReliaGATE 10-11 prodotto da Eurotech. Al suo interno troviamo un processore Texas Instruments TI AM335X Cortex-A8 equipaggiato con 512MB di RAM e 4GB di storage eMMC. Il ReliaGATE offre una vasta gamma di porte tra cui 232/485, 2CAN bus, 2 porte USB e 2 porte Ethernet, inoltre ha connettività Bluetooth, WiFi e GPS. Al suo interno è installato Everyware™ Software Framework (ESF), la versione commerciale del software Kura.



Figura 1.12: Caratteristiche elettriche SX1301

1.7.3 Architettura del software

Lo scopo di questa tesi era riuscire ad implementare dei moduli osgi, installabili al interno del framework ESF, in grado di controllare il comportamento del devices SX1301. In particolare utilizzando l'utility messa

a disposizione da Semtech per l'interfacciamento con il dispositivo denominata packet forwarder, e l'utility LoRa Gateway Bridge , sotto licenza MIT, sviluppata da brocaar .

[Aggiunger link](#)

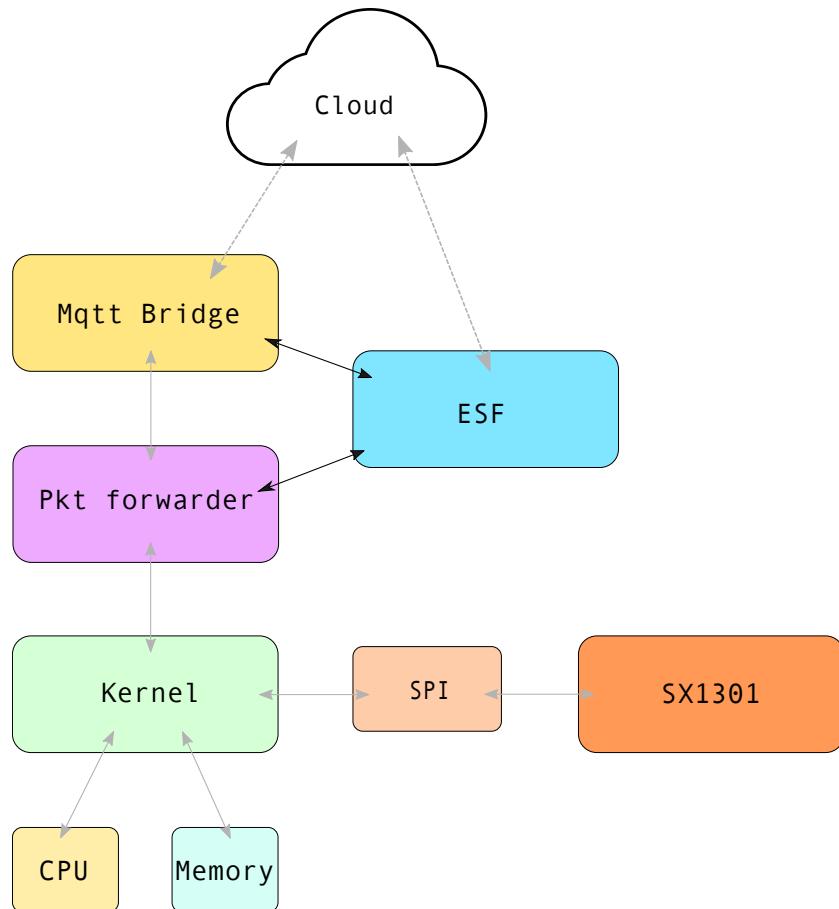


Figura 1.13: Architettura del software

1.7.4 Semtech packet forwarder

Il packet forwarder, è un software il quale riceve o invia pacchetti radio Lora , tramite una connessione SPI con il device SX1301. Nel caso di ricezione di un pacchetto, l'applicativo incapsula i dati ricevuti in un formato IP/UDP, e li ritrasmette nella rete internet/intranet. Per la sua configurazione viene utilizzato un file Json nel quale trviamo tutte le varie opzioni di configurazione per i moduli radio presenti al interno del chip.

```
1 "SX1301_conf": {
2     "lorawan_public": true,
3     "clksrc": 1,
4     "radio_0": {
5         "enable": true,
6         "type": "SX1257",
7         "freq": 867500000,
8         "rss_i_offset": -166.0,
9         "tx_enable": true
10    },
11    "radio_1": {
12        "enable": true,
13        "type": "SX1257",
14        "freq": 868500000,
15        "rss_i_offset": -166.0,
16        "tx_enable": false
17    },
18    "chan_multiSF_0": {
19        "enable": true,
20        "radio": 1,
21        "if": -400000
22    },
23 }
```

```
173 "gateway_conf": {
174     "gateway_ID": "AA555A0000000000",
175     "server_address": "0.0.0.0",
176     "serv_port_up": 1703,
177     "serv_port_down": 1702,
178     "keepalive_interval": 10,
179     "stat_interval": 30,
180     "push_timeout_ms": 100,
181     "forward_crc_disabled": false,
182     "forward_crc_valid": true,
183     "forward_crc_error": false
184 }
```

1.7.5 LoRa Gateway Bridge

LoRa Gateway Bridge, è l'applicativo il quale riceve i vari pacchetti UDP inviati dal packet forwarder e li trasmette ad un Broker MQTT. Il software è scritto nel linguaggio GO. Per renderlo interfacciabile con ESF, sono state apportate delle modifiche al codice, in particolare è stata aggiunta la possibilità di specificare il publish e il subscribed topic, i quali erano hard-coded.

1.7.6 ESF

Everyware software framework, è la versione commerciale del software Kura preinstallata nel ReliaGATE 10-11. ESF si pone l'obiettivo di offrire un container Java/OSGi per le applicazioni M2M. Gli applicativi installati in ESF sono sviluppati come OSGi Declarative Service, dando la possibilità di configurare i vari applicativi del gateway tramite una semplice interfaccia web.

Ev@ryware Gateway

ClockService

Apply

enabled*
Whether or not to enable the ClockService
 true false

clock.set.hwclock*
Whether or not to sync the system hardware clock after the system time gets set
 true false

clock.provider*
Source for setting the system clock

clock.ntp.host*
The hostname that provides the system time via NTP

clock.ntp.port*
The port number that provides the system time via NTP

clock.ntp.timeout*
The NTP timeout in milliseconds

clock.ntp.max-retry*
The maximum number of retries for the initial synchronization (with interval `clock.ntp.retry.interval`). If set to 0 the service will retry forever.

clock.ntp.retry.interval*

Figura 1.14: Architettura del software

1.7.7 Realizzazione

Per gestire i due software preinstallati si è optato per la creazione di due applicativi osgi distinti. Il primo applicativo chiamato Lora Config si pone il compito di leggere ed interpretare il file di configurazione utilizzato dal programma *Packet Forwarder*, per poi andare ad esporre i punti principali al utente finale. La libreria utilizzata per manipolare i file di tipo Json è

```
import com.eclipsesource.json
```

Tramite la quale vengono riempiti i campi della classe LoraSettings. Il file Json, come detto precedentemente è composto da due parti. Nella prima parte troviamo tutte le impostazioni per la configurazione del SX1301, nella seconda parte sono presenti le impostazioni per il forward dei pacchetti. Per questa ragione due classi diverse sono state create, SX1301Configuration e GatewayConfiguration, accessibili tramite la classe LoraSettings.

```
1 public class LoraSettings {
2     public static final String KEY_GATEWAY_CONFIG = "gateway_conf";
3     public static final String KEY_SX1301 = "SX1301_conf";
4     private SX1301Configuration sx1301Conf;
5     private GatewayConfiguration gatewayConf;
6 }
```

Per applicare le modifiche eseguite alla configurazione, è necessario che l'applicativo sia riavviato. Per eseguire questa operazione, si è scelto di utilizzare la libreria

```
import com.apache.commons.exec
```

la quale fornisce delle API per chiamare processi esterni, in particolare si è scelto di utilizzare l'utility di sistema *pkill* per terminare il processo del pkt forwarder.

```
1 public void startPktForwarder() {
2     DefaultExecutor pktExecutor = new DefaultExecutor();
3     CommandLine pktCmdLine = new CommandLine(KEY_PKT_BIN);
```

```

4         pktCmdLine.addArgument("start");
5         pktExecutor.setExitValue(0);
6         try {
7
8             pktExecutor.execute(pktCmdLine);
9             s_logger.info("start PKT");
10        } catch (Exception e) {
11            s_logger.warn("Couldn't start pkt forwarder");
12        }
13    }
14
15    public void stopPktForwarder() {
16        DefaultExecutor pktExecutor = new DefaultExecutor();
17        CommandLine pktCmdLine = new CommandLine("pkill");
18        pktCmdLine.addArgument("basic_pkt_fwd");
19        pktExecutor.setExitValue(0);
20        try {
21
22            pktExecutor.execute(pktCmdLine);
23            s_logger.info("Stop PKT");
24        } catch (Exception e) {
25            s_logger.warn("Couldn't stop pkt forwarder");
26        }
27    }

```

1.7.8 Mqtt Bridge Config

Il secondo bundle creato ha lo scopo di esporre all'utente un facile modo per configurare *Lora Gateway Bridge*. Anche in questo caso è necessario il riavvio del applicativo per fare in modo che le modifiche abbiano effetto, anche in questo caso è stata usata la libreria

```
import com.apache.commons.exec
```

Tramite questo bundle l'utente finale ha la possibilità di decidere

- il topic sul quale pubblicare i messaggi ricevuti in formato UDP dal *pkt forwarder*.
- il topic al quale rimanere in ascolto .

Lora Config

Apply Reset Delete

gateway.ID *
Lora MAC gateway.
AA555A0000000000

server.address *
Udp server address.
0.0.0.0

server.port.forward *
Port of the forwarded UDP packet.
1780

server.port.receiving *
Port of the receiving UDP packet.
1782

keepalive.interval *
Keep alive interval.
10

stat.interval *
Interval between statistics notifications.
30

push.timeout.ms *
Push notification timeout in ms.
100

forward.crc.valid *
Enable forward of crc.
 true false

forward.crc.error *
Enable forward of package with incorrect crc.
 true false

forward.crc.disabled *
Enable forward package crc.
 true false

Figura 1.15: Architettura del software

- su quale indirizzo e porta mandare/ricevere i pacchetti UDP.
- a quale broker Mqtt iscriversi.
- l'username e password per connettersi al broker.

1.7.9 Misurazioni

In via sperimentale, è stato scelto di installare il gateway ReliaGate 10-11 ad una altezza di 11m. Configurato il gateway per l'utilizzo del broker open source mosquitto.org. La prova di ricezione è stata condotta per tentativi, cercando di testare la massima distanza di trasmissione del dispositivo LoRa Mote .

Mqtt Bridge Config

Apply Reset Delete

mqtt.settings.publish.topic*
Name of the topic for the forwarded packets.

mqtt.settings.subscribed.topic*
Name of the subscribed topic.

mqtt.settings.udp.server.and.port*
Udp server and port.

mqtt.settings.server.and.port*
Address and port of the mqtt broker.

mqtt.settings.username
Username.

mqtt.settings.password
Password.

Figura 1.16: Architettura del software

Per constatare l'avvenuta ricezione del messaggio è stata utilizzata l'applicazione Android gratuita MyMQTT, tramite la quale è stato possibile iscriversi allo stesso topic in cui il gatway era configurato.



Figura 1.17: Architettura del software

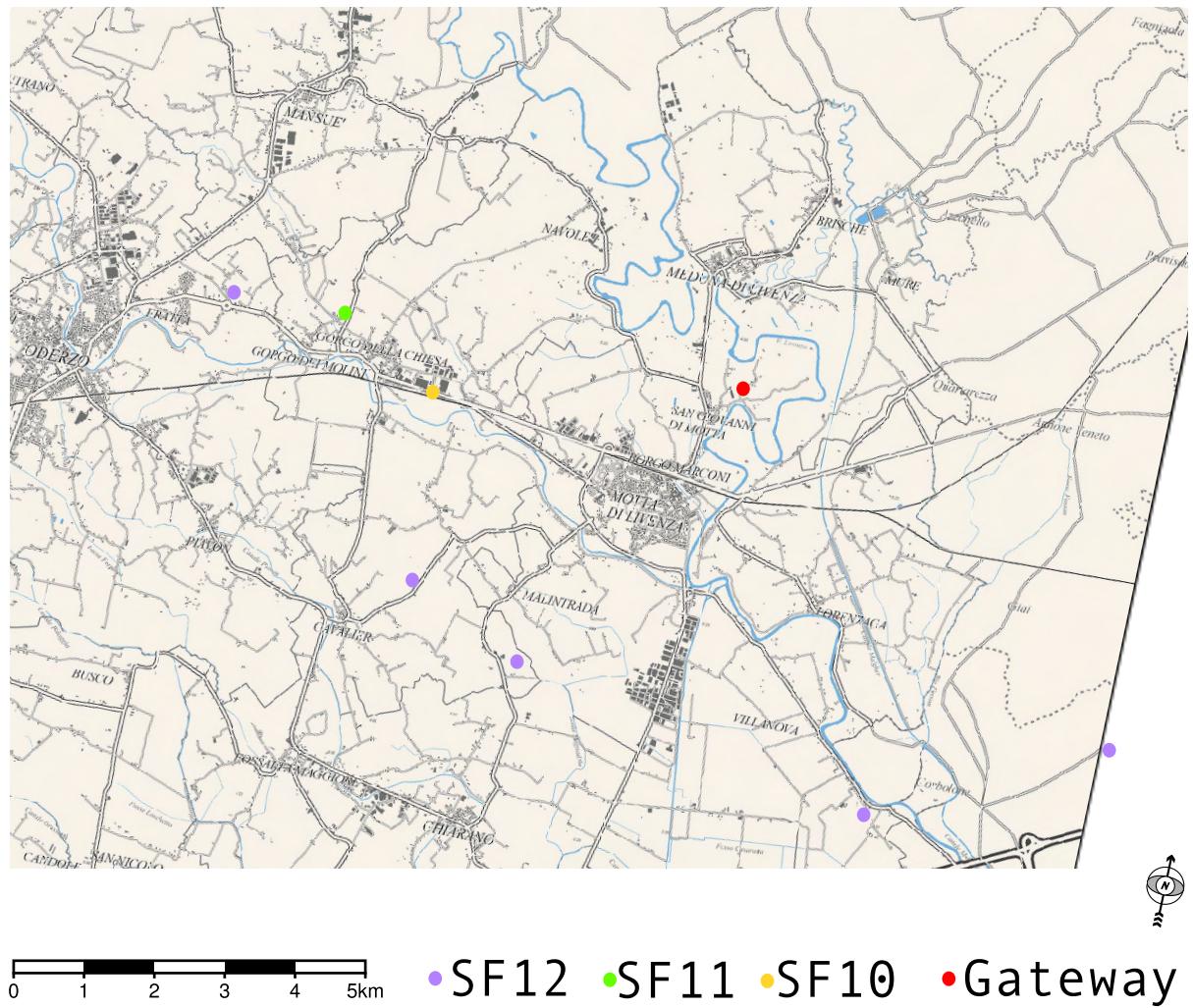


Figura 1.18: Architettura del software

Bibliografia

- [1] Gartner. Gartner says 8.4 billion connected 'things' will be in use in 2017, up 31 percent from 2016. *[Online]*, 2016. <http://www.gartner.com/newsroom/id/3598917>.
- [2] Semtech. An1200.22 lora™ modulation basics. *[Online]*, 2015. <http://www.semtech.com/images/datasheet/an1200.22.pdf>.
- [3] Semtech. Lorawan 101 a technical introduction. *[Online]*, 2017. https://docs.wixstatic.com/ugd/ecc1a_20fe760334f84a9788c5b11820281bd0.pdf.