

Relazione IoT Systems

Daniele Tognetti

10 marzo 2023

1 Scopo del progetto

La seguente relazione si pone l'obiettivo di andare ad analizzare due delle tecnologie di comunicazione wireless implementabili all'interno di dispositivi IoT: "Long Range (LoRa)" e "Wi-Fi low energy". Per poter confrontare i due metodi di comunicazione sopra citati sono stati presi in analisi fattori fondamentali in ambito IoT quali il consumo di batteria, l'affidabilità, la latenza, la velocità di trasmissione e la distanza massima che il segnale riesce a percorrere. Le analisi sono state effettuate sia in luoghi in cui erano presenti ostacoli e in altri in cui non erano presenti.

2 Dispositivi utilizzati

I dispositivi IoT scelti per il seguente progetto sono elencati di seguito

2.1 Feather M0 RFM95

Feather M0 è un microcontrollore con un ricetrasmittitore radio dotato di tecnologia "Long Range (LoRa)" in cui è integrata una porta USB e una di ricarica di un'eventuale batteria collegata al dispositivo.

Il dispositivo utilizzato in questo progetto è dotato di un modulo radio da **868 MHz** già integrato all'interno, il RFM9X, basato sul modulo SX1276 LoRa con cui comunica utilizzando l'interfaccia SPI. Il Feather M0 utilizza un processore ARM Cortex M0 ATSAMD21G18, con clock a 48 MHz, 256 KB di memoria flash e 32KB di RAM.

2.2 ESP32 devkit v1

Esp32 DevKit v1 è una delle schede di sviluppo basate sul modulo ESP-WROOM-32.

Il processore offre diverse tecnologie di comunicazione tra le quali: Wi-Fi, Bluetooth e Bluetooth Low Energy(BLE).

Il microprocessore utilizza una CPU Xtensa dual-core a 32-bit LX6, che può operare con un clock a 160 o a 240 MHz, 320 KB di memoria RAM, 448 KB di flash e con un modulo Wi-Fi in grado di arrivare ad un throughput pari a 150 Mbps.

3 Librerie e programmi utilizzati

3.1 ESP-NOW

La libreria e di conseguenza il protocollo ESP32-NOW è stato sviluppato da Espressif Systems per consentire trasferimenti di dati peer-to-peer wireless tra i dispositivi a basso costo e basso consumo della famiglia di microcontrollori ESP32.

Il protocollo ESP-NOW consente la trasmissione dati nella stessa banda a **2,4 GHz** del protocollo Wi-Fi, ma non viene stabilita alcuna connessione al WAP(Wireless Access Point) Wi-Fi e non interferisce con le normali operazioni Wi-Fi.

Il minor consumo energetico non si ottiene riducendo la potenza del trasmettitore, bensì semplificando il protocollo di connessione, riducendo così il tempo necessario affinché un'unità si connetta ad un'altra e quindi trasferisca un pacchetto di dati.

Più breve è il tempo necessario per il trasferimento completo, minore sarà il consumo di energia.

Il protocollo ESP-NOW è concepito per la trasmissione di piccoli pacchetti di dati fino a un massimo di **250 byte**.

Espressif definisce il protocollo ESP-NOW come un protocollo di comunicazione senza connessione tra i dispositivi Master e Slave.

Il protocollo ESP-NOW consente inoltre la comunicazione bidirezionale tra dispositivi ESP.

In questa modalità, un dispositivo può funzionare contemporaneamente sia da trasmettitore (master) che da ricevitore (slave).

Il numero massimo di dispositivi che si possono connettere contemporaneamente sono 20 mentre se viene utilizzata una comunicazione cifrata questo numero scende a 17.

La velocità massima dichiarata da Espressif per tale protocollo è di **1 Mbps**

La struttura dei frame inviati è mostrata in figura 1.

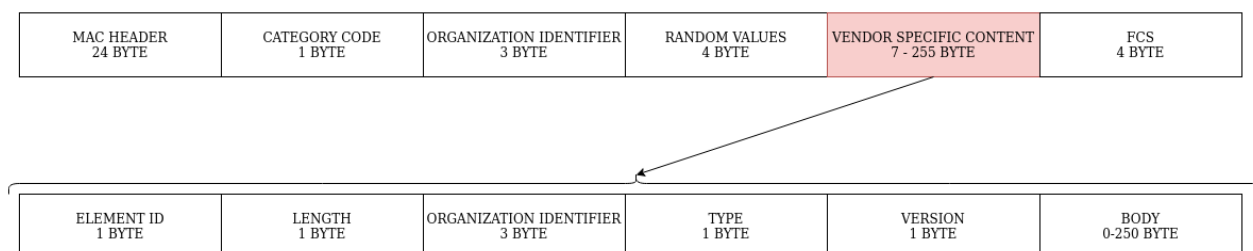


Figura 1: Struttura di un frame generato dalla libreria ESP-NOW

3.2 RadioHead

RadioHead Packet Radio è una libreria orientata agli oggetti che fornisce dei metodi per l'invio e la ricezione di pacchetti tramite una moltitudine di trasmettitori radio differenti.

La libreria è composta da 2 gruppi principali di classi: Driver e Manager.

- Driver: Forniscono un accesso di basso livello ad una vasta gamma di trasmettenti radio e sono la base su cui vengono utilizzati i manager.
- Manager: Forniscono servizi per invio e ricezione di messaggi ad alto livello per diversi utilizzi.

Solitamente un programma che utilizza la libreria RadioHead crea un'istanza di un driver per fornire l'accesso al trasmettitore radio e, normalmente, un manager che utilizza quel driver per inviare e ricevere messaggi per una certa applicazione.

È anche possibile utilizzare solamente il driver, utilizzando i metodi messi a disposizione dallo stesso, senza un manager, sebbene ciò comporti un protocollo di comunicazione inaffidabile e senza possibilità di indirizzamento.

Nel progetto realizzato è stato utilizzato solamente il driver RH RF95 che permette di interfacciarsi con diversi moduli Lora tra cui quello utilizzato dal microcontrollore Feather M0 Lora RFM95.

Tale scelta è dovuta ad un maggiore controllo sui pacchetti inviati e ricevuti che hanno permesso di ottenere delle metriche più precise inerenti al tempo impiegato dal pacchetto per essere spedito e successivamente ricevuto.

La dimensione massima che può avere un pacchetto è di **250 byte** e il modulo LoRa ha una velocità massima teorica di **300kbps**. La struttura dei frame inviati è mostrata in figura 2.

Preamble 8 byte	Explicit Header	Header 4 byte	Data 250 byte	CRC 2 byte
--------------------	--------------------	------------------	------------------	---------------

Figura 2: Struttura di un frame generato dalla libreria RadioHead

3.3 Eclipse Mosquitto

Eclipse Mosquitto è un broker di messaggi open source che implementa diverse versioni del protocollo MQTT.

Mosquitto è leggero ed è adatto per l'uso su tutti i dispositivi, da quelli più economici e a bassa potenza ai server più performanti.

Il protocollo MQTT fornisce un protocollo di trasporto messaggi leggero utilizzando il modello publish/subscribe.

Questo lo rende adatto ad usi in ambito IoT, come ad esempio con sensori a basso consumo o a microcontrollori.

Il progetto Mosquitto fornisce anche una libreria C per l'implementazione di client MQTT e l'utilizzo di comandi come publish e subscribe ad un determinato topic che si è rivelato utile in fase di configurazione del sistema.

3.4 Node Red

Node-RED è uno strumento di programmazione open source, per connettere dispositivi hardware, API e servizi online in modo intuitivo e semplice.

In prima istanza è un programma dotato di interfaccia grafica progettato per l'Internet of Things, ma il suo utilizzo può anche essere sfruttato da altre applicazioni.

L'impiego dell'applicazione prevede l'utilizzo di blocchi di codice software basati su JavaScript, chiamati nodi.

Questi nodi predefiniti riducono il rischio inerente allo sviluppo del software e accelerano la realizzazione del programma stesso.

I nodi vengono trascinati e rilasciati all'interno dell'area di lavoro per rendere lo sviluppo di applicazioni IoT più semplice, ripetibile e dotato di maggiore scalabilità.

4 Funzionamento del progetto

L'interfaccia utente è realizzata mediante l'utilizzo della dashboard di Node Red.

Qui l'utente attraverso il menù presente in alto a sinistra, può scegliere quale modulo utilizzare tra LoRa e il "Wi-Fi low power".

In entrambi i casi viene chiesto all'utente di inserire, all'interno di un form, il mac address di destinazione del pacchetto e il numero di pacchetti che si desidera inviare.

Quando l'utente preme il bottone di submit, vengono controllati i campi del form per verificare se questi siano stati effettivamente inseriti e nel caso questo non fosse avvenuto viene generato un avviso. Il mac address viene controllato a livello sintattico e nel caso mostrato il warning relativo.

In entrambi i casi precedenti non vengono eseguite ulteriori azioni.

Se invece i campi sono corretti, Node Red pubblica sul broker MQTT il comando relativo per far partire il modulo richiesto.

All'interno del progetto i due MQTT client, appartenenti ai due moduli, sono iscritti ai seguenti due topic:

- Per LoRa: /command/esp32
- Per il Wi-Fi: /command/esp32.2

Il messaggio viene quindi pubblicato da Node Red con un topic corrispondente al modulo che l'utente ha precedentemente scelto e contenente le due informazioni inserite dall'user nel form precedente.

Il client MQTT del modulo scelto, successivamente, manda le informazioni ricevute, utilizzando la porta seriale, al dispositivo a lui direttamente connesso e si mette in attesa di ricevere i risultati del test.

Il dispositivo ricevente, una volta ottenuti i dati necessari, procede con l'invio dei pacchetti al dispositivo destinatario.

Durante questa fase vengono calcolate le diverse metriche che poi verranno mostrate all'utente, ovvero il numero di pacchetti totali, quelli ricevuti correttamente in risposta a quelli inviati ed anche tempo e throughput medio.

Al termine della fase di invio/ricezione dei messaggi, ovvero quando i valori dei parametri sopra riportati sono definitivi, il dispositivo procede ad inviarli attraverso il collegamento seriale al client MQTT. Il client MQTT, che era in attesa di ricevere dati dalla porta seriale, una volta acquisiti i dati, gli pubblica sul broker utilizzando per ciascuno di essi un topic diverso, che risultano essere i seguenti:

- LoRa:
 - /result/esp32/num_pack_rcv: numero di pacchetti ricevuti
 - /result/esp32/num_pack_tot: numero di pacchetti inviati
 - /result/esp32/tempo: tempo medio di invio e risposta di un pacchetto
 - /result/esp32/throughput: throughput medio di un pacchetto
- Wi-Fi:
 - /result/esp32_2/num_pack_rcv: numero di pacchetti ricevuti
 - /result/esp32_2/num_pack_tot: numero di pacchetti inviati
 - /result/esp32_2/tempo: tempo medio di invio e risposta di un pacchetto
 - /result/esp32_2/throughput: throughput medio di un pacchetto

Node Red, che attraverso i nodi MQTT, è sottoscritto ai topic sopra citati, prende i dati e li mostra all'utente attraverso la dashboard mediante anche l'uso di grafici.

Il funzionamento dell'intero progetto è mostrato in maniera sintetica in figura 3

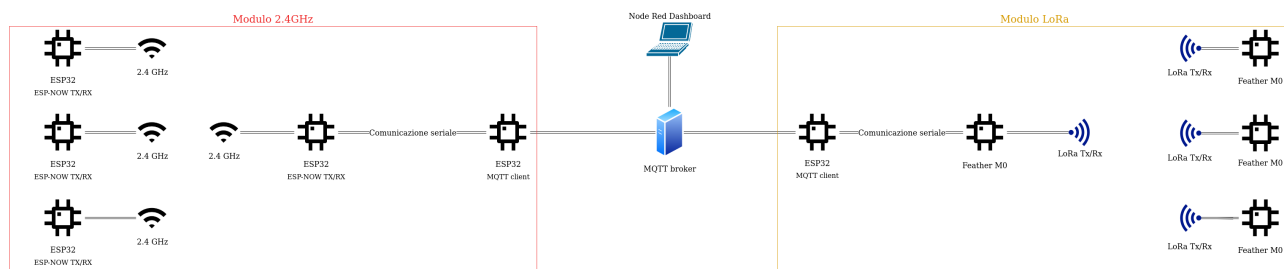


Figura 3: Overview del progetto

5 Risultati ottenuti

5.1 Considerazioni preliminari

I test con la tecnologia LoRa sono stati condotti utilizzando un'antenna di lunghezza pari a 8,2 cm (dimensione coerente con la frequenza di 868 MHz utilizzata) realizzata attraverso un semplice filo conduttore collegato al microcontrollore e con potenza di trasmissione massima (23 dBm).

Utilizzando un'antenna specificamente realizzata per lo scopo, le prestazioni ottenute sarebbero probabilmente migliori.

Per ogni test condotto sono stati utilizzati 10 pacchetti, un numero che garantisce una discreta stabilità nei valori ottenuti.

Ciascun pacchetto è stato impostato con una dimensione pari a 250 byte.

5.2 LoRa

Di seguito i risultati ottenuti utilizzando la tecnologia LoRa

5.2.1 LoRa in ambiente domestico

I risultati qui esposti sono stati eseguiti in ambiente domestico all'interno di un appartamento.

Di conseguenza il segnale era soggetto a diversi ostacoli durante il suo percorso.

Sono stati testate tre diverse possibili configurazioni relative al segnale trasmesso.

Bw125Cr45Sf128 I parametri utilizzati sono stati i seguenti:

- Band width: 125kHz
- Code rate: 4/5
- Spreading factor: 7
- CRC: on

Tale configurazione dovrebbe permettere un bilanciamento tra distanza percorsa dal segnale e velocità di trasmissione.

I risultati ottenuti sono i seguenti:

Distanza	Pacchetti persi	Tempo medio	Throughput
1m senza ostacoli	0	62 ms	32258 bit/s
8 metri con ostacoli (stesso piano)	0	62 ms	32258 bit/s
13 metri con ostacoli (piano diverso)	0	62 ms	32258 bit/s
15 metri con ostacoli (piano diverso)	0	62 ms	32258 bit/s

Bw500Cr45Sf128 I parametri utilizzati sono stati i seguenti:

- Band width: 500kHz
- Code rate: 4/5
- Spreading factor: 7
- CRC: on

Tale configurazione permette una velocità di trasferimento dati massima a discapito della distanza massima.

I risultati ottenuti sono i seguenti:

Distanza	Pacchetti persi	Tempo medio	Throughput
1m senza ostacoli	0	18 ms	111111 bit/s
8 metri con ostacoli (stesso piano)	0	18 ms	111111 bit/s
13 metri con ostacoli (piano diverso)	0	18 ms	111111 bit/s
15 metri con ostacoli (piano diverso)	0	18 ms	111111 bit/s

Bw125Cr48Sf4096 I parametri utilizzati sono stati i seguenti:

- Band width: 125kHz
- Code rate: 4/8
- Spreading factor: 12
- CRC: on

Tale configurazione permette al segnale di percorrere la massima distanza a discapito della velocità di trasferimento.

I risultati ottenuti sono i seguenti:

Distanza	Pacchetti persi	Tempo medio	Throughput
1m senza ostacoli	0	2036 ms	982 bit/s
8 metri con ostacoli (stesso piano)	0	2037 ms	981 bit/s
13 metri con ostacoli (piano diverso)	0	2037 ms	981 bit/s
15 metri con ostacoli (piano diverso)	0	2037 ms	981 bit/s

Considerazioni sull'uso di LoRa in ambiente domestico In tutte e tre le configurazioni provate il segnale è riuscito a coprire interamente l'appartamento di circa 100 metri quadri inclusi anche il solaio e il garage.

Nell'eventualità di utilizzo del dispositivo utilizzando una batteria è possibile ridurre i consumi andando a settare una potenza di trasmissione adatta alla distanza che si vuole coprire.

Da notare come la velocità, nei diversi test effettuati, non sia cambiata ma sia rimasta costante nonostante la presenza di numerosi ostacoli lungo il tragitto.

Per un utilizzo indoor, alla luce dei risultati ottenuti, risulta più efficiente la configurazione Bw500Cr45Sf128 che permette una velocità superiore di trasferimento dei dati e una latenza inferiore.

5.2.2 LoRa in ambiente outdoor

I risultati sotto esposti sono stati eseguiti in ambiente outdoor all'interno di un parco cittadino in un contesto urbano, in cui quindi possono verificarsi situazioni di disturbo del segnale radio.

Il segnale, in alcuni dei test, è stato soggetto alla presenza di ostacoli, principalmente abitazioni, mentre in altri non vi erano presenti impedimenti.

Il trasmettitore e il ricevitore si trovavano entrambi a circa un metro e cinquanta centimetri di altezza dal suolo.

Di seguito i luoghi in cui sono stati effettuati i test:

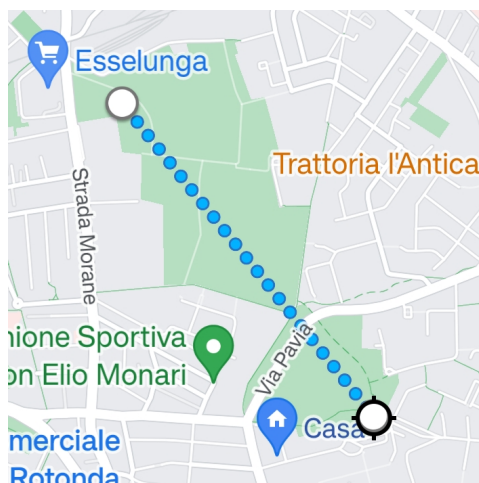


Figura 4: Test 1 - 1.1 km senza ostacoli

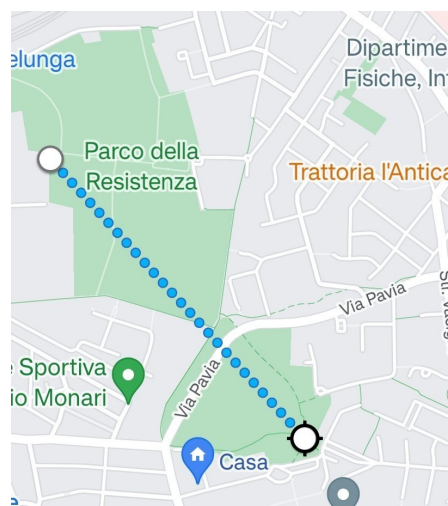


Figura 5: Test 2 - 900 m senza ostacoli

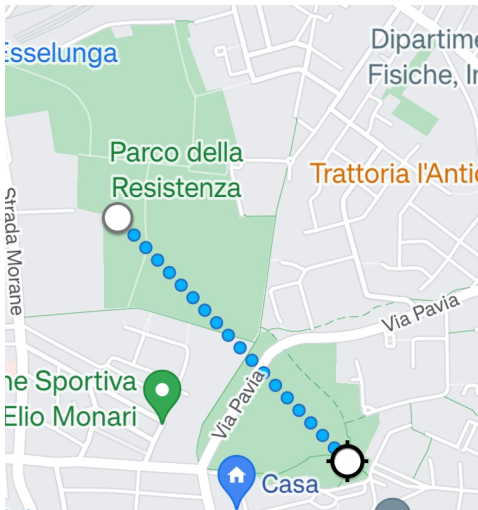


Figura 6: Test 3 - 750 m senza ostacoli

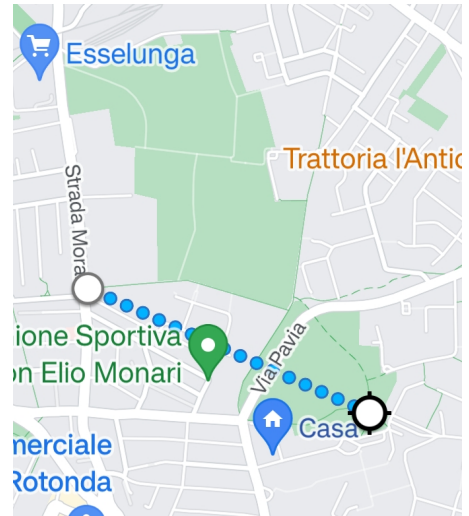


Figura 7: Test 4 - 780 m con ostacoli

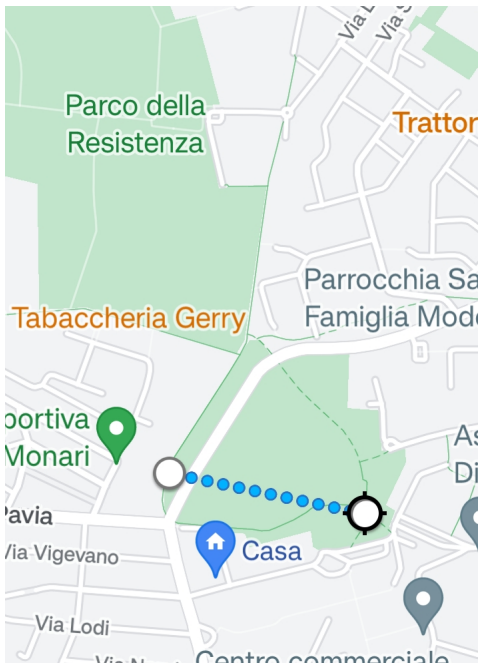


Figura 8: Test 5 - 350 m senza ostacoli

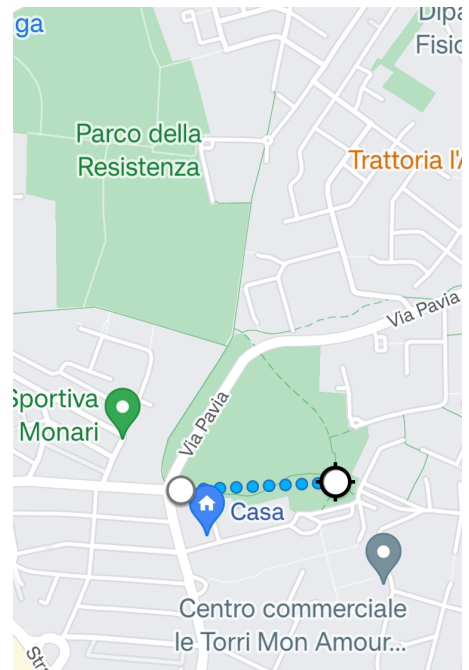


Figura 9: Test 6 - 300 m con ostacoli

Sono stati testate anche in questo caso tre diverse possibili configurazioni del modem LoRa.

Bw125Cr48Sf4096 I parametri di questa configurazione sono stati già esposti precedentemente.

	Pacchetti persi	Tempo medio	Throughput
Test 1 - 1.1 km senza ostacoli	9	2034 ms	983 bit/s
Test 2 - 900 m senza ostacoli	2	2034 ms	983 bit/s
Test 3 - 750 m senza ostacoli	0	2034 ms	983 bit/s
Test 4 - 780 m con ostacoli	2	2033 ms	983 bit/s
Test 5 - 350 m senza ostacoli	0	2034 ms	983 bit/s
Test 6 - 300 m con ostacoli	0	2035 ms	982 bit/s

Da tali risultati è emerso che, in assenza di ostacoli nel percorso, è stato possibile raggiungere i 1100 metri di distanza utilizzando questa configurazione del modem.

Tale distanza però non garantisce un'affidabilità tale da poter essere utilizzata, per aver tale garanzia bisogna scendere appena sotto i 900 metri.

La presenza di alcune abitazioni ha interferito in modo lieve senza una significativa compromissione del segnale.

Bw500Cr45Sf128 I parametri di questa configurazione sono stati già esposti precedentemente.

	Pacchetti persi	Tempo medio	Throughput
Test 1 - 1.1 km senza ostacoli	10	0 ms	0 bit/s
Test 2 - 900 m senza ostacoli	10	0 ms	0 bit/s
Test 3 - 750 m senza ostacoli	5	18 ms	111111 bit/s
Test 4 - 780 m con ostacoli	10	0 ms	0 bit/s
Test 5 - 350 m senza ostacoli	1	18 ms	111111 bit /s
Test 6 - 300 m con ostacoli	2	18 ms	111111 bit/s

I risultati mostrano come questa configurazione riduca il raggio di azione a circa 750 metri e venga influenzato in maniera significativa da ostacoli come le abitazioni.

Sulla lunga distanza tale setup permette comunque di mantenere costante la velocità di trasmissione dei dati che risulta superiore alle altre due configurazioni.

Il compromesso per raggiungere un throughput più elevato viene pagato in un'affidabilità che risulta essere la peggiore delle tre configurazioni provate.

Bw125Cr45Sf128 I parametri di questa configurazione sono stati già esposti precedentemente.

Distanza	Pacchetti persi	Tempo medio	Throughput
Test 1 - 1.1 km senza ostacoli	10	0 ms	0 bit/s
Test 2 - 900 m senza ostacoli	5	62 ms	32258 bit/s
Test 3 - 750 m senza ostacoli	0	62 ms	32258 bit/s
Test 4 - 780 m con ostacoli	0	0 ms	0 bit/s
Test 5 - 350 m senza ostacoli	0	62 ms	32258 bit/s
Test 6 - 300 m con ostacoli	0	62 ms	32258 bit/s

Da tali dati possiamo affermare che per ottenere una copertura stabile, la distanza massima che può separare trasmettente e ricevente è intorno agli 800 metri. La presenza di case in questa configurazione non ha diminuito la portata massima della trasmettente.

5.2.3 Consumo elettrico

Bw125Cr45Sf128 La tecnologia LoRa utilizzando la configurazione del modem Bw125Cr45Sf128, attraverso il misuratore di corrente, ha fatto registrare un consumo di 23,4 mA in stato di ricezione mentre in fase di trasmissione arriva ad un consumo massimo di 60 mA. Il consumo di potenza, considerando l'alimentazione a 5V, risulta quindi essere circa 0,117 Watt in ricezione e 0,300 Watt di picco in trasmissione.

Bw500Cr45Sf128 La tecnologia LoRa utilizzando la configurazione del modem Bw500Cr45Sf128, attraverso il misuratore di corrente, ha fatto registrare un consumo di 25,3 mA in stato di ricezione mentre in fase di trasmissione arriva ad un consumo massimo di 36 mA. Il consumo di potenza, considerando l'alimentazione a 5V, risulta quindi essere circa 0,126 Watt in ricezione e 0,180 Watt di picco in trasmissione.

Bw125Cr48Sf4096 La tecnologia LoRa utilizzando la configurazione del modem Bw500Cr45Sf128, attraverso il misuratore di corrente, ha fatto registrare un consumo di 23,4 mA in stato di ricezione mentre in fase di trasmissione arriva ad un consumo massimo di 133 mA. Il consumo di potenza, considerando l'alimentazione a 5V, risulta quindi essere circa 0,117 Watt in ricezione e 0,665 Watt di picco in trasmissione.

Considerazioni sui consumi Appare evidente dai dati che l'utilizzo di una configurazione del modem per inviare il segnale a lungo raggio fa registrare consumi più elevati.

All'interno della libreria RadioHead utilizzata è possibile diminuire la potenza di trasmissione riducendo i consumi ma diminuendo la portata del segnale.

In fase di configurazione infatti è possibile cambiare la potenza da un minimo +5 ad un massimo di +23 dBm, scelta che deve tener in considerazione la distanza che deve percorrere l'utente ma anche il consumo energetico nel caso il microprocessore sia collegato ad una batteria.

Inoltre è possibile abbassare leggermente il consumo energetico non accendendo il led all'arrivo di un pacchetto, con un risparmio di circa 0,0035 Watt.

5.3 Wi-Fi

I test condotti utilizzando la tecnologia Wi-Fi low energy sono stati condotti in prima battuta all'interno dell'ambiente domestico e successivamente anche all'esterno in un parco cittadino.

Negli spazi indoor erano naturalmente presenti degli ostacoli principalmente legati alle pareti dell'abitazione.

5.3.1 Wi-Fi in ambiente domestico

Distanza	Pacchetti persi	Tempo medio	Throughput
1 m	0	6.62 ms	276133 bit/s
12 m	0	7.89 ms	208862 bit/s
11 m a fianco del ripetitore Wi-Fi	10	0 ms	0 bit/s
10 m e ripetitore Wi-Fi a 1 m	2	17.78 ms	83665 bit/s
12 m	0	15.50 ms	143113 bit/s
16 m	10	0 ms	0 bit/s
4 m con attraversamento di un piano	0	6.33 ms	296469 bit/s
12 m con attraversamento di un piano	10	0 ms	0 bit/s
10 m con attraversamento di due piani	10	0 ms	0 bit/s

I risultati mostrano come in un appartamento di 100 metri quadri il segnale venga propagato in quasi tutto l'alloggio.

Da evidenziare come il posizionamento vicino ad un ripetitore Wi-Fi degradi molto il segnale, fino ad arrivare a "coprirlo" completamente nel caso venga posizionato a fianco ad esso.

Anche cambiando il canale in uso alla rete Wi-Fi i risultati sono stati i medesimi.

In caso di ricevente posizionata ad un piano abitativo diverso da quello della trasmittente è stato possibile comunicare solo quando la loro posizione era perpendicolare l'una all'altra e con un solo piano a separarle.

5.3.2 Wi-Fi in ambiente outdoor

Questa batteria di test è stata realizzata in assenza di ostacoli andando ad allontanarsi progressivamente dalla sorgente.

Trasmittitrice e ricevente sono stati posti ad un metro e cinquanta centimetri di altezza.

Di seguito i risultati.

Distanza	Pacchetti persi	Tempo medio	Throughput
20 m	0	6.30 ms	305201 bit/s
30 m	0	6.75 ms	295762 bit/s
40 m	0	6.00 ms	330666 bit/s
50 m	0	6.49 ms	280271 bit/s
60 m	0	6.01 ms	329067 bit/s
100 m	1	9.55 ms	182938 bit/s
105 m	4	21.33 ms	92605 bit/s
115 m	10	0 ms	0 bit/s

Analizzando i risultati si può notare come fino a circa cento metri di distanza il segnale viene ricevuto in modo pressoché perfetto senza nessuna perdita di pacchetti.

Il throughput, a differenza di LoRa, risulta essere abbastanza variabile da test ad un altro e si può notare come questo diminuisca all'aumentare della distanza.

Nei test LoRa invece la velocità era rimasta costante anche a distanze considerevoli.

La copertura effettiva è risultata essere di circa centocinque metri prima di perdere completamente il segnale wireless.

5.3.3 Consumo elettrico

La tecnologia Wi-Fi low energy, attraverso il misuratore di corrente, ha fatto registrare un consumo di 149 mA in stato di ricezione mentre in fase di trasmissione arriva ad un consumo di 150 mA. Il consumo di potenza, considerando l'alimentazione a 5V, risulta quindi essere circa 0,75 Watt.

6 Conclusioni

La scelta di una tecnologia wireless rispetto ad un'altra dipende fondamentalmente dal tipo di progetto che stiamo realizzando.

In linea generale LoRa soddisfa quelle realtà che necessitano di far percorrere al segnale lunghe distanze e/o che utilizzano una batteria come fonte di alimentazione ma che non devono trasferire grosse quantità di dati.

Dall'altro lato abbiamo la tecnologia Wi-Fi che riesce a trasferire in modo più rapido i dati ma ha una portata limitata e un consumo maggiore di batteria.