

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

---

SCUOLA DI SCIENZE  
Corso di Laurea in Informatica

**ESPERIMENTI SUL CONTROLLO REMOTO  
DI STAZIONI PER IL MONITORAGGIO DI  
PROCESSI GEOFISICI  
PER LA VALUTAZIONE DI GEOHAZARD**

**Relatore:**  
Chiar.mo Prof.  
VITTORIO GHINI

**Presentata da:**  
STEFANO SGARLATA

**Correlatore:**  
Dott.  
STEFANO  
CACCIAGUERRA

**Sessione II**  
**Anno Accademico 2014-2015**



# Introduzione

In collaborazione con la sezione di Bologna dell'Istituto Nazionale di Geofisica e Vulcanologia (INGV) si è dato il via ad un progetto mirato alla costruzione di nuove stazioni monitoranti.

Stazioni che dovranno avere un basso impatto in termini di costi e consumi energetici, dando contemporaneamente la possibilità di essere pilotate a distanza.

Quest'ultimo è un parametro fondamentale perché permetterebbe di interagire con la stazione anche durante il verificarsi di calamità naturali.

Per giungere ad una soluzione si è quindi proceduto analizzando i possibili scenari e le soluzioni attualmente impiegate. Valutandone gli aspetti negativi e positivi.

Successivamente, terminata la fase di documentazione, si è passati a vagliare le soluzioni tecnologiche attualmente disponibili sul mercato e ad assemblare un prototipo sperimentale.

Non appena in possesso di una macchina sulla quale poter effettuare tutte le prove necessarie si è passati alla fase successiva.

L'implementazione di test che potessero rappresentare al meglio una situazione reale.

Da qui si è giunti ad una conclusione, tenendo in considerazione diversi fattori: le condizioni imposte dall'ambiente durante le operazioni di test ed i risultati ottenuti.



# Indice

<b>Introduzione</b>	<b>i</b>
<b>1 Geohazard</b>	<b>1</b>
1.1 Tettonici . . . . .	1
1.2 Vulcanici . . . . .	2
1.3 Metereologici e causati dall'uomo . . . . .	2
1.4 Studio dei processi geofisici . . . . .	3
1.4.1 Stazioni fisse . . . . .	6
1.4.2 Stazioni mobili . . . . .	8
1.4.3 Stazioni a confronto . . . . .	11
1.4.4 Limiti delle stazioni e soluzione proposta . . . . .	12
<b>2 Stazioni usa e getta</b>	<b>15</b>
2.1 Hardware . . . . .	16
2.1.1 Single-board computer . . . . .	17
2.1.2 Moduli radio . . . . .	19
2.2 Sistema operativo . . . . .	19
2.3 Connettività . . . . .	21
2.4 Consumo energetico . . . . .	23
2.5 Analisi della soluzioni proposta . . . . .	24
2.5.1 Autonomia . . . . .	24
2.5.2 Costi . . . . .	26
2.5.3 Controllo remoto . . . . .	26
2.5.4 Portabilità . . . . .	27

---

2.6	Possibili scenari di funzionamento . . . . .	28
<b>3</b>	<b>Stato dell'arte</b>	<b>29</b>
3.1	Installazione e configurazione delle librerie software . . . . .	29
3.1.1	RF24Network_config.h . . . . .	31
3.1.2	RF24Gateway . . . . .	31
3.2	Strumenti adottati . . . . .	34
3.3	Script . . . . .	36
<b>4</b>	<b>Test e analisi dei risultati</b>	<b>39</b>
4.1	Scenario . . . . .	39
4.2	Test . . . . .	40
4.2.1	Considerazioni . . . . .	43
	<b>Conclusioni</b>	<b>45</b>
	<b>Bibliografia</b>	<b>47</b>

# Elenco delle figure

1.1	Stazione permanente di monitoraggio sismico . . . . .	6
1.2	Stazione mobile di monitoraggio sismico . . . . .	8
2.1	Accelerometro/giroscopio digitale . . . . .	16
2.2	Modulo radio nrf24 . . . . .	19
2.3	Rete stazioni monitoranti in collegamento diretto . . . . .	21
2.4	Rete stazioni monitoranti con ripetitori . . . . .	22
2.5	Modem umts Conel ur5 . . . . .	25
2.6	Acquisitore reftek 130-1 . . . . .	25
2.7	Power bank 30000mAh con pannello solare . . . . .	26
3.1	Schema connessioni test . . . . .	34
4.1	Trasferimenti a 100m e 2Mbit/s . . . . .	42
4.2	Trasferimenti a 200m e 250kbit/s . . . . .	42
4.3	Trasferimenti a 300m e 250kbit/s . . . . .	43



# Elenco delle tabelle

2.1	Raspberry Pi . . . . .	18
3.1	Stack di rete TCP/IP over rf24[31] . . . . .	30
3.2	Diff RF24Gateway_ncurses.cpp . . . . .	33
3.3	pseudo codice del diagramma di flusso . . . . .	36
3.4	Script per l'esecuzione dei test . . . . .	38
3.5	Automatizzazione dello script 3.4 . . . . .	38



# Capitolo 1

## Geohazard

Un geohazard è uno stato geologico con potenzialità distruttive che può portare a danni estesi sul territorio con conseguente rischio per l'incolumità della popolazione.

I geohazards sono caratterizzati da processi geologici di breve o lunga durata. Possono manifestarsi con intensità variabili, dal più lieve non comportando alcun rischio per l'uomo o per l'ambiente circostante, fino ad investire grosse parti di territorio con una potenza tale da modificarne definitivamente la morfologia.

Le origini di questi processi possono essere di natura tettonica, vulcanica, causati da fenomeni meteorologici o dall'intervento umano.

### 1.1 Tettonici

I fenomeni di natura tettonica, come i terremoti terrestri e sottomarini, si verificano in zone marginali interessate da uno scorrimento laterale di due placche o da un loro accavallamento (subduzione).

L'enorme quantità di energia accumulata nel tempo viene rilasciata improvvisamente generando un terremoto.

Gli eventi sismici più catastrofici fanno parte di questa categoria e sono localizzati in zone ben precise della crosta terrestre: cintura di fuoco del

Pacifico, costa ovest del nord America, mediterraneo nord-orientale, Iran, Nuova Zelanda, Indonesia, Giappone e Cina.

## 1.2 Vulcanici

Sempre riconducibili alla teoria della tettonica a placche, gli eventi vulcanici generano fenomeni come terremoti, eruzioni effusive ed esplosive e colate piroclastiche.

A differenza dei terremoti di origine tettonica, un sisma vulcanico (tremore armonico) è un fenomeno solitamente di durata maggiore ed è associato alla risalita del magma ed al rilascio di gas che tipicamente precedono un'eruzione.

La composizione del magma, che una volta fuoriuscito prende il nome di lava, ed in particolare la sua viscosità è il fattore principale che permette di distinguere tra eruzione effusiva ed esplosiva.

Se il magma non è eccessivamente denso allora l'eruzione sarà di tipo effusivo e la lava colerà lungo le pendici del vulcano, al contrario se il magma ha una densità tale da non permettere lo sfogo dei gas, una volta raggiunta una pressione limite questi daranno luogo ad un'esplosione distruggendo parzialmente o totalmente l'edificio vulcanico.

Tutto il materiale espulso da un'eruzione esplosiva, ricadendo genererà delle nubi ad altissime temperature (500°C - 1200°C) contenenti gas misti a ceneri, bombe vulcaniche e lapilli che discendono le pendici del vulcano a velocità variabili tra i 50Km/h e i 300Km/h.

Queste nubi, devastanti per le regioni adiacenti all'eruzione, prendono il nome di colate piroclastiche o nubi ardenti.

## 1.3 Metereologici e causati dall'uomo

Con caratteristiche ben differenti, i processi geologici causati da eventi climatici o dall'intervento umano si possono identificare con frane o crolli, clas-

sificati in base al tipo di materiale movimentato.

Possono interessare piccole aree o interi costoni di una montagna o una collina.

Le loro cause sono molteplici e spesso sono la combinazione di più fattori, come la composizione del suolo, il suo contenuto d'acqua, l'erosione, la deforestazione e la costruzione incontrollata di edifici e infrastrutture che alterano gli equilibri già precari del territorio.

Più in generale, quando si verifica una rottura allora si è in presenza di uno squilibrio tra forze agenti che tendono a far muovere la massa e forze reagenti che tendono a mantenerla compatta.

I fattori che tendono a far aumentare le forze agenti sono suddivisi in:

- fattori che asportano il materiale (erosione e scavi)
- fattori che aumentano il carico (piogge eccessive)
- fattori che riducono il supporto sotterraneo (disboscamento, attività minerarie)
- fattori che esercitano sforzi transitori (sismi ed esplosioni)

Quelli che invece tendono a far diminuire le forze reagenti sono:

- fattori intrinseci (natura del terreno, pendenza)
- fattori esterni (sollecitazioni transitorie, variazione del contenuto d'acqua)

## 1.4 Studio dei processi geofisici

Da questa breve descrizione è facile comprendere come questo tipo di fenomeni sia in grado di minacciare la sicurezza di persone e infrastrutture, siano esse di tipo civile o industriale.

Uno strumento indispensabile per tentare di prevedere, quando possibile, il verificarsi o meno di un determinato evento o quantomeno la sua entità è

costituito dallo studio dei fattori scatenanti, dagli sviluppi del fenomeno e dagli effetti che questo può avere sull'area colpita.

Uno studio di questo genere dà ai progettisti la possibilità di scegliere la località e le modalità migliori per la costruzione di un edificio o di una strada ed alla protezione civile informazioni sufficienti per impostare un piano efficace d'evacuazione o di recupero.

Col fine di fornire il maggior numero di informazioni possibili, sia per scopi civili che di ricerca, l'istituto nazionale di geofisica e vulcanologia (INGV) ha installato su tutto il territorio nazionale numerose stazioni di rilevamento. Stazioni che hanno il compito di monitorare zone ad alto rischio, rilevando e registrando tutti i mutamenti di quei parametri che caratterizzano un determinato fenomeno.

Periodicamente, i dati registrati vengono prelevati dagli operatori o inviati tramite reti apposite direttamente alle centrali di controllo dove verranno analizzati.

Per dare un ordine al sempre crescente numero di installazioni, l'intera rete di acquisizione è strutturata suddividendo l'insieme delle stazioni in reti e sotto reti, differenziate in base ai sensori installati nelle singole stazioni ed organizzate nella maniera seguente:

- rete sismica
  - accelerometri e velocimetri
    - rete GPS
      - Stazioni CGPS (Continuous Global Positioning System) dotate di ricevitori di tipo geodetico con precisione nell'ordine dei centimetri
    - rete flame
      - Sensori per l'analisi del flusso di anidride solforosa (SO<sub>2</sub>)
- rete geochimica aree vulcaniche
  - rete acque
    - Sensori per il controllo di: temperatura, conducibilità, PH, pres-

sione totale dei gas disciolti, pressione parziale di anidride carbonica, livello freatico o portata

– rete gas

Sensori per il controllo del flusso di anidride carbonica (CO<sub>2</sub>) dal suolo

– rete plume

Stazioni posizionate sui crateri sommitali, controllano il rapporto CO<sub>2</sub>/SO<sub>2</sub>

– rete temperatura

• rete geochimica aree sismiche

– rete acque

– rete gas

Sensori per il controllo dei flussi di anidride carbonica (CO<sub>2</sub>) e metano (CH<sub>4</sub>) dal suolo

È importante notare che per garantire la massima precisione durante le misurazioni, il posizionamento di una stazione è un aspetto fondamentale.

La vicinanza alla fonte del fenomeno interessato impone quindi dei limiti su come e dove installare la stazione monitorante.

Ci si potrebbe trovare in condizioni tali che la conformazione del territorio sia favorevole o sfavorevole per grandi installazioni permanenti.

Il medesimo problema è causato anche dal tipo di fenomeno interessato, questo potrebbe essere di natura transitoria e richiedere solo saltuariamente dei controlli lasciando la stazione inutilizzata per lunghi periodi.

Come soluzione per questo tipo di problematiche l'INGV ha in dotazione due tipologie differenti di stazioni monitoranti, stazioni fisse o permanenti e stazioni mobili o portatili, ognuna con i propri vincoli ma perfettamente in grado di svolgere il proprio compito in buona parte degli scenari possibili.

### 1.4.1 Stazioni fisse

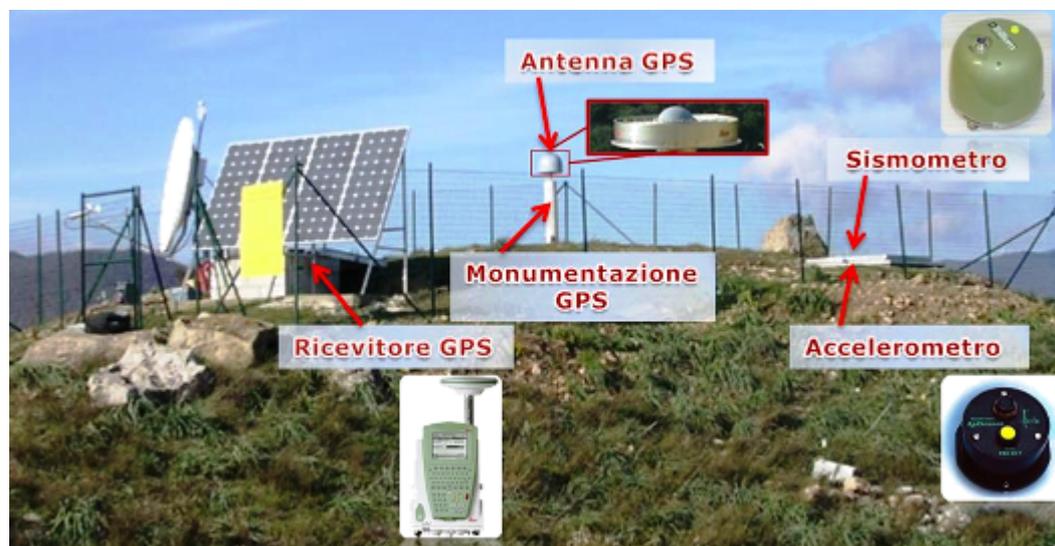


Figura 1.1: Stazione permanente di monitoraggio sismico

Sono grandi installazioni permanenti che richiedono dei lavori preliminari necessari alla preparazione del sito.

Vengono utilizzate per attività di monitoraggio continuo e prolungato ed essendo delle stazioni che dovranno rimanere operative per diversi anni si ha la necessità di trovare un luogo che non solo sia sufficientemente vicino alla zona di interesse ma che permetta anche al personale di svolgere le proprie attività in totale sicurezza.

In aggiunta a questi vincoli, c'è anche da tenere in considerazione la facile raggiungibilità del sito da parte dei macchinari necessari alla preparazione ed al trasporto delle attrezzature.

Questi sono solamente alcuni dei requisiti, più che altro legati alla scelta della località.

Ulteriori restrizioni sono poi imposte dalle corrette modalità di installazione e posizionamento di ogni singola parte della stazione.

Parti che si possono identificare con:

- Strumenti di misura che, per garantire la massima precisione, possono aver bisogno di alloggiamenti ad hoc (es. pozzetti con profondità variabile dai pochi centimetri fino a diversi metri).
- Antenne paraboliche per la connessione satellitare e ricevitori GPS che devono essere posizionati ed orientati in maniera tale che il segnale satellitare non incontri nessun ostacolo.
- Pannelli fotovoltaici utilizzati per alimentare l'intera stazione o per dotarla di un sistema di backup nel caso in cui si verificano malfunzionamenti della rete elettrica.  
Anch'essi devono essere posizionati ed orientati in maniera tale che non si creino zone d'ombra durante le ore diurne della giornata.
- Acquisitori. Apparecchiature necessarie alla conversione dei segnali analogici provenienti dai sensori in segnali digitali.
- Batterie, cavetteria ed elettroniche varie utilizzate per i collegamenti e per i circuiti di alimentazione.
- Un capanno o un contenitore per proteggere dagli agenti atmosferici e da eventuali animali tutte le parti delicate come gli acquisitori, le batterie, le elettroniche ed i modem.

Ovviamente, questo tipo di soluzione assolve perfettamente il suo compito in tutti quegli scenari che ne permettono l'installazione, dando la possibilità di massimizzare il rendimento di ogni singola componente.

Tutta questa precisione, purtroppo, porta ad una minore flessibilità ed a costi quasi doppi rispetto ad una stazione portatile ( $\approx 20.000\text{€}$ ).

Sintetizzando, una stazione fissa ha il vantaggio di:

- proteggere e posizionare in maniera ottimale i sensori.
- orientare perfettamente tutti gli apparati satellitari e fotovoltaici.
- avere eventualmente a disposizione un allaccio alla rete elettrica.

- avere accesso in maniera diretta o con un numero limitato di ripetitori a reti wifi o UMTS/GPRS.

Contemporaneamente porta con se anche degli aspetti negativi, quali:

- scelta oculata del sito (trade-off tra prossimità, raggiungibilità e posizionamento della strumentazione).
- necessità di lavori preliminari
- necessità di personale tecnico specializzato per montaggio e manutenzione
- costi doppi rispetto ad una soluzione portatile ( $\approx 20.000\text{€}$ )

### 1.4.2 Stazioni mobili



Figura 1.2: Stazione mobile di monitoraggio sismico

Soluzione che, grazie alla sua flessibilità viene solitamente impiegata in zone cui sarebbe troppo complesso o impossibile optare per una stazione fissa. Alle volte anche come supporto temporaneo a quest'ultima per permettere una maggiore precisione nella localizzazione della fonte ed un monitoraggio continuo e ravvicinato del sito.

Questo tipo di stazioni sono solitamente contenute, ad eccezione dei pannelli fotovoltaici, all'interno di apposite cassette o valigette ed hanno un peso

che si attesta intorno ai 25Kg, per la maggior parte imposto dalla batteria. Tutto il necessario deve essere facilmente trasportabile e posizionabile in qualsiasi condizione, sia che si tratti di un campo in aperta campagna che di un punto isolato sulle pendici di un vulcano.

Purtroppo però tutta questa flessibilità ha un costo in termini di precisione, sicurezza e autonomia.

Il posizionamento dei sensori non può essere pianificato ed ottimizzato, di conseguenza le misurazioni potrebbero essere disturbate causando una perdita di precisione (es. cedimenti del terreno sotto il sensore).

Inoltre, dato che l'intera stazione non può essere protetta, potrebbe essere danneggiata da animali, agenti atmosferici o eventi catastrofici.

Ultimo, ma non per importanza, è il problema dell'autonomia.

Difatti essendo una stazione mobile, la principale fonte di alimentazione è costituita da una batteria che per quanto possa essere capiente ha sempre una durata limitata ed insufficiente.

Nel tentativo di dare una stima si è analizzato il rapporto tecnico n.137 dell'INGV [15] dove vengono illustrate le componenti di una stazione mobile per il monitoraggio sismico.

Le più rilevanti ai fini di questa ricerca risultano essere:

- Batteria al piombo da 42Ah.
- Acquisitore REF-TEK 130-1 da 1,7W.
- Modem UMTS Conel UR5 da 2W.
- Gruppo sensori (voce utile per il calcolo) 0,78W.

Questa specifica stazione ha un consumo variabile dai 4W, in totale assenza di rilevamenti, agli 8W con misurazioni continue seguite da scrittura e invio di dati.

Con questi consumi, una stazione riuscirebbe a rimanere operativa per un periodo di tempo compreso tra i 2 giorni e mezzo e i 5 giorni.

Valori ampiamente inferiori ai tempi medi di utilizzo (mesi o addirittura anni).

Per ovviare a questo problema, vengono aggiunti dei pannelli fotovoltaici di potenza sufficiente a rigenerare la carica consumata durante le ore notturne o in caso di giornate scarsamente soleggiate (tra i 50W e i 70W).

In aggiunta a queste problematiche, c'è anche da considerare la manutenzione necessaria a rendere le stazioni sempre operative.

È infatti necessario raggiungere la stazione per il download dei dati, nel caso in cui non si avesse accesso alle reti di telecomunicazioni e mantenere i pannelli solari al massimo della loro efficienza, liberandone sistematicamente la superficie da accumuli di polvere, fango, neve e ghiaccio.

Sintetizzando, una stazione mobile porta degli effettivi vantaggi in aspetti come:

- Precisione nel posizionamento.
- Assenza di lavori preliminari.
- Portabilità.
- Costo dimezzato rispetto ad una stazione fissa ( $\approx 10.000\text{€}$ ).

In contrapposizione troviamo alcuni lati negativi difficilmente risolvibili:

- Esposizione della strumentazione agli agenti atmosferici ed agli animali.
- Posizionamento non ottimale dei sensori soggetti a movimenti verticali ed orizzontali che disturbano le misurazioni. causati da scosse sismiche, frane ed erosione del suolo.
- Peso 25Kg - 30Kg.
- Autonomia limitata dalla capacità della batteria e dallo stato dei pannelli fotovoltaici.
- Possibili zone d'ombra per segnali satellitari ed esposizione solare.

- raggiungibilità per manutenzione.
- Disponibilità di reti per la trasmissione dei dati.
- Necessità di personale tecnico specializzato.
- Costi. In caso di calamità perdere delle stazioni del valore di 10.000€ ciascuna è un danno economico consistente.

### 1.4.3 Stazioni a confronto

Come precedentemente accennato, queste due soluzioni vengono sistematicamente impiegate nella quasi totalità degli scenari possibili.

Una stazione fissa, grazie alla possibilità di un accesso facilitato alla rete elettrica si presta perfettamente per monitoraggi prolungati nel tempo e nel caso in cui si dovessero verificare malfunzionamenti può sempre contare sul fotovoltaico.

Sistema che per sua natura riesce a convertire più energia se la superficie irradiata è maggiore.

E dal momento che si hanno a disposizione aree più ampie, si ha la possibilità di sovradimensionare i pannelli per minimizzare l'attività di manutenzione necessaria a mantenerli puliti.

Questo non è fattibile in una stazione mobile, dato che le dimensioni e il peso sono fattori fondamentali per mantenere una certa flessibilità nel posizionamento.

Flessibilità che purtroppo si guadagna a scapito della pulizia dei rilevamenti.

Difatti un sensore ben posizionato e protetto genererà delle letture più accurate, riuscendo quasi ad eliminare i rumori di fondo.

Per questo motivo capita anche che delle stazioni mobili vengano utilizzate come supporto per stazioni fisse.

Posizionando le prime il più vicino possibile alla sorgente e lasciando alle ultime il compito di ricevere i dati ed eventualmente spedirli tramite la rete.

Un'ulteriore differenza tra le due soluzioni sta proprio nella facilità di accesso ai mezzi di comunicazione.

Una stazione fissa, avendo a disposizione più spazio e più energia, può sfruttare un collegamento satellitare.

Soluzione non praticabile per una stazione mobile che deve necessariamente ripiegare su connessioni alla rete cellulare, purtroppo non garantita per zone remote.

La conseguenza di tutto questo è la costante necessità di personale per attività di manutenzione e per il download dei dati.

Personale che deve ovviamente avere la garanzia di operare in totale sicurezza, escludendo quindi gli scenari più estremi.

#### **1.4.4 Limiti delle stazioni e soluzione proposta**

L'analisi di tutti gli scenari possibili e delle problematiche che questi possono generare, unita allo studio delle soluzioni tecnologiche attualmente utilizzate ha reso palesi alcuni limiti.

Per quanto queste possano essere precise e raffinate, hanno dei costi troppo elevati ed abbandonarle in condizioni critiche per ricavare più dati possibili non è una via percorribile.

Una possibile soluzione ci viene suggerita dalla proliferazione di nuovo hardware a basso costo e basso consumo dalle dimensioni sempre più ridotte e facilmente reperibile.

L'idea di base che ha dato vita a questo progetto è proprio quella di creare una nova categoria di stazioni che potremmo definire "usa e getta", altamente portatili e a basso costo.

Sfruttando tutta questa varietà di componenti è infatti possibile assemblare questo tipo di stazione e dotarla di un sistema operativo Linux e di sensori programmabili in base alle esigenze.

In aggiunta a questo, un ulteriore vantaggio è dato dalla possibilità di avere un controllo remoto della stazione.

Limitando ulteriormente l'intervento del personale alla sola fase iniziale di posizionamento ed eventualmente finale di recupero.



## Capitolo 2

### Stazioni usa e getta

L'idea è nata dalla necessità di avere delle stazioni monitoranti che potessero essere posizionate all'interno di zone critiche, senza per questo doversi preoccupare dell'incolumità del personale o del rischio di danneggiamento della strumentazione.

Per zone critiche si intendono tutte quelle aree nelle immediate vicinanze o addirittura all'interno del raggio d'azione di processi geologici distruttivi.

Per dare un esempio, uno scenario tipico potrebbe essere un'eruzione vulcanica imminente o già in atto e questa stazione verrebbe posizionata a pochi metri di distanza dal cratere o sul percorso che sarà seguito dalla colata lavica o dalla nube ardente.

Durante questi eventi, per motivi legati alla sicurezza, al personale è severamente vietato di recarsi nelle zone interessate. Di conseguenza anche se si fosse disposti a sacrificare una stazione, questa non potrebbe essere posizionata.

La soluzione che proponiamo è quindi quella di sfruttare, come precedentemente accennato, l'enorme diffusione di nuovo hardware commercializzato per fini hobbistici o per prototipazione.

Il punto di forza di questo nuovo settore sta nella sua semplicità e nella limitata conoscenza dell'elettronica necessaria a muovere i primi passi.

Gli unici requisiti richiesti sono l'essere in possesso di un microcontrollore

programmabile, di sensori o attuatori necessari per il progetto ed ovviamente la capacità di sviluppare del codice.

Delle conoscenze basilari in elettronica possono tornare utili ma non sono un requisito.

Difatti tutte le componenti necessarie si trovano spesso già assemblate e pronte per essere collegate e programmate.



Figura 2.1: Accelerometro/giroscopio digitale

## 2.1 Hardware

Il punto cardine di tutto il progetto sta nella possibilità di avere il controllo remoto dell'intera stazione e di tutte le sue componenti.

Per poter avere questa possibilità un microcontrollore non è ovviamente sufficiente.

Il loro funzionamento è legato ai programmi precedentemente caricati e non consentirebbero una eventuale riconfigurazione una volta posizionati.

Questo ostacolo è facilmente superabile optando per una piattaforma di sviluppo che consenta l'installazione di un vero e proprio sistema operativo.

Hardware di questo tipo prende il nome di single-board computer ed è ormai molto comune.

In commercio si trovano modelli sempre più diversi con consumi, dimensioni e potenze di calcolo adatte per qualsiasi tipo di progetto.

Alcune delle case produttrici hanno addirittura scelto di rilasciare i propri prodotti con licenze open-hardware, ovvero con la stessa filosofia che sta dietro alle licenze open-source per i software.

Chiunque può prendere i progetti e modificarli a piacimento ma deve necessariamente rilasciarli con la stessa licenza del progetto di partenza, salvo eventuali accordi tra le aziende.

Una scelta di questo tipo ha favorito enormemente la proliferazione di queste piattaforme, contribuendo al loro sviluppo.

Tra le aziende che hanno intrapreso questa strada si può trovare la Olimex con la sua serie di prodotti *OlinuXino*, la Texas Instruments con le schede della famiglia *BeagleBoard* e la Sinovoip con i modelli di *Banana Pi*.

Altre aziende, per motivi diversi, rilasciano più o meno documentazione riguardo il proprio hardware.

Alle volte perché legate da contratti con le case produttrici dei microprocessori.

Questo è il caso della Raspberry Pi Foundation con la sua serie di schede *Raspberry Pi*.

### 2.1.1 Single-board computer

La presenza di una community molto attiva e di numerosi progetti già avviati ha portato a scegliere come piattaforma per questo progetto uno dei prodotti della Raspberry Pi Foundation, il *Raspberry pi B+*.

Questo single-board computer ha delle dimensioni di poco maggiori di una carta di credito, 85.60mm x 56.5mm e pesa solamente 45g.

Attualmente sono in commercio quattro varianti di questa scheda.

Due (*Raspberry Pi A+*, *Raspberry Pi B+*) montano un system-on-chip della Broadcom (*BCM2835*) con a bordo una CPU della famiglia ARM (*ARM1176JZF-S*) a 700Mhz, una GPU e 256MB o 512MB di memoria ram.

Il quadro viene completato rispettivamente da 1 o 4 porte USB, 0 o 1 scheda ethernet 10/100Mbit, 2 uscite video (HDMI e composito), 1 uscita audio (Jack 3.5mm) ed 1 slot micro sd.

Le restanti due sono una un'evoluzione di questi due modelli (*Raspberry Pi2 B*, quad-core ARM 900Mhz, 1GB di ram) ed una loro miniaturizzazione (*Raspberry Pi Zero*, ARM 1Ghz, 512MB di ram).

Tutte e quattro permettono la connessione di sensori ed attuatori grazie a 40 pin GPIO (General Purpose Input/Output) programmabili.

Hardware	A+	B+	Pi2	Zero
				
Memoria	Micro-sd	Micro-Sd	Micro-sd	Micro-sd
Clock Cpu	700Mhz	700Mhz	900Mhz	1Ghz
Core	1	1	4	1
Ram	256MB	512MB	1GB	512MB
Usb	1	4	4	1 micro-usb
Video	HDMI	HDMI	HDMI	HDMI mini
Audio	Jack stereo	Jack stereo	Jack stereo	-
Networking	-	Rj45 Auto MDI-X	Rj45 Auto MDI-X	-
GPIO	40	40	40	40

Tabella 2.1: Raspberry Pi

### 2.1.2 Moduli radio



Figura 2.2: Modulo radio nrf24

Per dotare la stazione di connettività, si è poi scelto di aggiungere un modulo radio *NRF24L01+* con a bordo un chip prodotto dalla Nordic Semiconductor ed 1 antenna da 2db.

Avendo a disposizione delle prese ethernet o USB si sarebbe potuto pensare di riutilizzare il modem *UMTS Conel UR5* o aggiungere una soluzione WiFi 802.11 ma questo non avrebbe portato alcun vantaggio dovendo aggiungere dei dispositivi con conseguente aumento di peso e consumi.

La soluzione scelta ha invece un consumo ridotto e delle dimensioni di 45,5mm x 16,5mm e pesa 12g.

Non ostante le piccole dimensioni, il datasheet indica in condizioni ottimali (campo privo di ostacoli in linea d'aria) una copertura massima di 1000m con bitrate a 250kbit/s, 750m a 1Mbit/s e 520m a 2Mbit/s.

## 2.2 Sistema operativo

La Raspberry Pi Foundation pubblica le immagini del sistema operativo ufficiale presso la pagina <https://raspberrypi.org/downloads>.

Li si possono trovare diversi sistemi, alcuni orientati ad un utilizzo della scheda come media center che ovviamente non sono stati presi in considerazione.

La versione scelta è la Raspbian; un sistema operativo compatibile con i processori ARM basato sulla nota distribuzione *Linux Debian*.

Il vantaggio di un sistema di questo genere è dato principalmente dalla possibilità di avere una vera e propria macchina Linux completa.

Quindi si hanno a disposizione tutti gli strumenti che questo sistema operativo mette a disposizione.

Due fra tanti:

- Il gestore dei pacchetti apt, che permette di installare qualunque software sia stato rilasciato o ricompilato per questa piattaforma.
- Tutti i linguaggi che hanno a disposizione un interprete o un compilatore in grado di funzionare su cpu arm.

È importante sottolineare che oltre a sviluppare direttamente sul *Raspberry Pi* si può anche sfruttare un comune calcolatore per virtualizzare l'intero sistema (utilizzando *QEMU*) o cross-compilare qualunque codice sorgente.

Operazioni che permettono di accorciare i tempi di sviluppo.

Infatti, grazie alla virtualizzazione si può creare un ambiente all'interno del quale far funzionare il sistema operativo ed effettuare tutte le prove necessarie, senza per questo modificare il funzionamento della piattaforma.

Un discorso differente è quello riguardante la crosscompilazione.

Grazie alla quale è possibile ricompilare qualunque codice sorgente e renderlo eseguibile su un'architettura diversa da quella di origine.

Questa operazione permette quindi di riutilizzare tutto il software open-source sviluppato per sistemi Linux, anche per piattaforme differenti, senza dover ricorrere ad un compilatore nativo.

Accorciando drasticamente i tempi necessari allo sviluppo ed alla traduzione del software.

## 2.3 Connettività

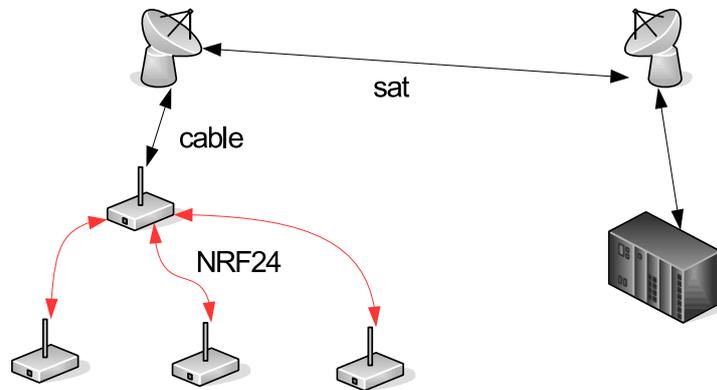


Figura 2.3: Rete stazioni monitoranti in collegamento diretto

La possibilità di utilizzare i protocolli TCP/IP ci dà ampia scelta su quali servizi la stazione potrà rendere disponibili.

Tuttavia prima di valutare l'utilizzo di questa soluzione è necessario inquadrare una possibile organizzazione della rete delle stazioni.

Innanzitutto è necessario che ogni singola stazione sia accessibile da una qualunque centrale di controllo della rete di monitoraggio nazionale.

Per permettere questo è possibile posizionare una stazione "base" presso qualunque luogo sia provvisto di un accesso ad internet, ad esempio una stazione permanente.

Questa fungerà da bridge fra la rete delle stazioni e la rete nazionale.

Da qui, se la distanza non è eccessiva, si possono già posizionare fino a 255 stazioni monitoranti.

Ognuna delle quali comunicherà direttamente con la stazione "base".

In alternativa, se la distanza da coprire è maggiore del raggio d'azione dei moduli, si possono frapporre tra la stazione "base" e la stazione monitorante fino a 254 stazioni intermedie.

Queste potranno ricoprire il ruolo di semplice ripetitore, riducendo al minimo l'equipaggiamento, o essere delle vere e proprie stazioni monitoranti con il

duplice compito di effettuare rilevamenti e far rimbalzare il segnale fino alla destinazione desiderata.

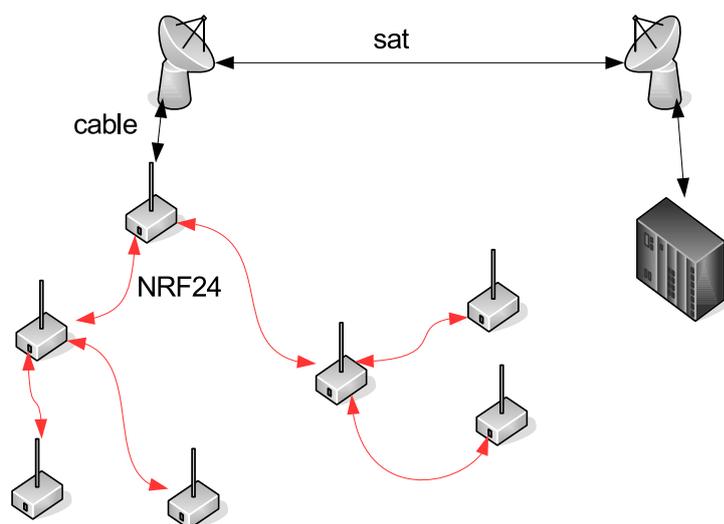


Figura 2.4: Rete stazioni monitoranti con ripetitori

Tutte le comunicazioni avvengono secondo la logica del "punto punto", ovvero i pacchetti vengono spediti direttamente alla stazione di destinazione. Sarà poi compito della stazione "base" effettuare la traduzione dell'indirizzo ip in un indirizzo mac (router).

In aggiunta a questo è anche presente la funzionalità mash, che permette ad un nodo di essere fisicamente spostato lasciando alla libreria software il compito di riconfigurare la topologia della rete.

Tutti questi aspetti sono sì fondamentali ma grazie all'utilizzo di alcune librerie software, brevemente introdotte nel capitolo successivo, non è stato necessario approfondire ulteriormente l'argomento.

Le uniche configurazioni richieste sono infatti l'assegnazione di un id e un indirizzo ip, univoci per ogni singola stazione.

## 2.4 Consumo energetico

Nel tentativo di dare una stima il più precisa possibile, sono state effettuate delle misurazioni sul *Raspberry Pi*.

Per poterlo fare è stato necessario modificare il cavo d'alimentazione per permettere l'inserimento di un multimetro in serie al capo con tensione maggiore ( $\approx 5$  Volt) e quindi poter misurare la corrente assorbita (Ampere). Per la misurazione della tensione, la procedura è stata molto più semplice poiché il *Raspberry Pi* espone dei pad collegati direttamente all'ingresso del circuito d'alimentazione (PP1=Vcc, PP3=GND).

Durante le misurazioni sono stati immaginati due scenari d'utilizzo differenti.

Il primo equivalente ad uno stato in cui la stazione non rileva nulla e quindi resta in "attesa" di possibili variazioni.

Simulato semplicemente attendendo l'avvio completo del sistema senza poi far eseguire alcun processo.

Ed il secondo equivalente ad uno stato in cui la stazione rileva continuamente dei mutamenti dell'ambiente circostante e quindi scrive in memoria.

Simulato tramite i comandi `nice yes 1>/dev/null` (carico cpu 80% - 100%) e `dd if=/dev/zero of=testwrite bs=1M count=512; sync` (salvataggio in memoria).

I risultati hanno rivelato un consumo di 0,97W ( $4,86V * 0,2A$ ) per il primo caso e di 1,29W ( $4,78V * 0,27A$ ) per il secondo.

Il valore variabile della tensione è dovuto alla stabilità dell'alimentatore, l'unica cosa da evitare per non incorrere in malfunzionamenti della memoria è di non scendere mai sotto i 4,65 Volt. Qualsiasi valore compreso tra questa soglia minima e 5 Volt è più che sufficiente.

Per evitare questa variabilità dal calcolo si è quindi scelto di utilizzare il valore indicato dalla casa produttrice di 5 Volt per ottenere i risultati finali. Di conseguenza si può affermare che il *Raspberry Pi B+* ha dei consumi pari a 1 Watt e 1,35 Watt.

Un ulteriore punto a favore di questa piattaforma, che consente di conte-

nera ulteriormente i consumi, è dato dalla possibilità di spegnere il controller USB.

Disattivando quindi tutte le 4 porte e la scheda ethernet.

Sfruttando questa possibilità si sono rieseguite le misurazioni in condizioni identiche, rivelando una diminuzione della corrente assorbita di 70mA – 80mA.

Di conseguenza i valori misurati corrispondono a 0,59W ( $4,96V * 0,12A$ ) e 0,97W ( $4,86V * 0,2A$ ) che arrotondati con lo stesso criterio delle misurazioni precedenti sono pari a 0,6 Watt e 1 Watt.

La stima dei consumi è poi completata aggiungendo l'energia consumata dal modulo radio NRF24.

Stando ai datasheet ufficiali della Nordik Semiconductor questo chip ha un assorbimento massimo di corrente pari a 13mA e va alimentato con una tensione di 3,3V.

Questi valori portano ad una stima di consumo pari a 0,043 Watt.

## 2.5 Analisi della soluzioni proposta

È bene specificare che questa nuova tipologia di stazioni non si propone come soluzione universale per tutti i problemi incontrati con le stazioni tradizionali.

Il suo scopo è solo quello di evitare il sacrificio di soluzioni più costose, che in ogni caso verrebbero impiegate, collaborando con queste e svolgendo la parte più pericolosa delle operazioni di monitoraggio.

Detto ciò, un'analisi dei vari aspetti è in ogni caso dovuta.

### 2.5.1 Autonomia

Dal paragrafo 2.4 si possono ricavare i consumi di una stazione base.

Questi sono compresi tra 0,6 Watt e 1 Watt.

Volendoli confrontare con quelli di una stazione mobile tradizionale è necessario prendere in esame le componenti che svolgono lo stesso compito.

Nello specifico abbiamo da un lato l'acquisitore *REF-TEK 130-1* e il modem *UMTS Conel UR5* e dall'altro il *Raspberry Pi* ed il modulo NRF24.

I primi hanno un consumo totale pari a 3,7 Watt mentre i secondi si fermano a 1,043 Watt.

Questo risparmio permette di ottenere la stessa autonomia utilizzando delle batterie meno capienti e quindi più leggere.



Figura 2.5: Modem umts Conel ur5



Figura 2.6: Acquisitore reftek 130-1

Per dare un esempio, si può iniziare riprendendo l'autonomia di una stazione mobile.

Questa è stata stimata in un valore compreso tra 2 giorni e mezzo e 5 giorni [1.4.2](#), dotando la stazione di una batteria da 42Ah dal peso di 13Kg.

Un'autonomia simile si potrebbe ottenere alimentando la nuova stazione con un classico power bank da soli 30Ah che riuscirebbe ad alimentarla per un intervallo di tempo compreso tra 4 giorni e mezzo e 10 giorni e mezzo.



Figura 2.7: Power bank 30000mAh con pannello solare

### 2.5.2 Costi

Utilizzando lo stesso criterio del paragrafo precedente e confrontando sempre gli stessi componenti, stavolta possiamo dare purtroppo solamente una stima dato che l'unica voce certa è quella relativa al *Raspberry Pi*.

Per le componenti della stazione mobile, essendo delle apparecchiature che hanno una nicchia di mercato molto ristretta, per avere una cifra indicativa è stato necessario estrapolare i costi da un documento ufficiale dell'INGV [21] per quanto riguarda l'acquisitore e da un sito che permette di acquistare il modem Conel.

In aggiunta a questi componenti, sono stati tenuti in considerazione anche i costi relativi all'acquisto delle batterie.

Portando la cifra totale intorno i 5850€ per una stazione mobile e circa 80€ per una usa e getta.

### 2.5.3 Controllo remoto

Grazie alle librerie software per il modulo radio NRF24 che permettono di utilizzare il protocollo TCP/IP ed alla duttilità del sistema operativo Linux abbiamo la possibilità di utilizzare tutti gli strumenti attualmente in circolazione.

Uno fra tanti: SSH.

Questo ci dà la possibilità di accedere alla stazione ed utilizzarla o configurarla come se stessimo operando fisicamente su di essa.

Eliminando quindi la necessità di attività di manutenzione ed aggiungendo la possibilità di riconfigurarla senza per questo doverla prelevare.

L'unico aspetto negativo di questa funzionalità risiede nella quantità di energia necessaria per eseguire tutte le richieste.

Difatti ogni tentativo di connessione richiederà l'esecuzione di un certo numero di processi per gestire la comunicazione, ai quali vanno aggiunti tutti quelli relativi ai comandi impartiti ed alla gestione dei sensori e dell'input/output. Tutte queste interazioni, per quanto possano essere minime o di breve durata, incideranno sull'autonomia totale della stazione.

#### 2.5.4 Portabilità

Questo probabilmente è l'aspetto più complesso e chiama in causa anche i precedenti.

Chiaramente le dimensioni della nuova stazione sono molto contenute (8,6cm x 5,6cm + 4,5cm x 1,6cm).

Il problema principale non è quanto è grande ma bensì quanto pesa.

Questo valore è legato ai consumi energetici in quanto la componente principale di questa voce è dettata dalla batteria.

Più questa deve essere capiente tanto più peserà.

Per dare un esempio, una batteria standard al piombo da 42Ah per le stazioni mobili pesa  $\approx 13\text{Kg}$ .

Fortunatamente i consumi di questa stazione sono nettamente inferiori e questo permette l'utilizzo anche di un classico power bank da scaffale.

Infatti quello utilizzato per la stima dell'autonomia [2.5.1](#) oltre a garantire la stessa durata pesa solamente 350g.

La componente peso totale è stata tenuta in forte considerazione perché si vorrebbe avere la possibilità di posizionare la stazione anche con l'ausilio di droni.

Questi velivoli hanno una capacità limitata di carico che non può oltrepassare

i 10Kg o per alcuni modelli anche 15Kg.

Quindi, un peso eccessivo potrebbe automaticamente a doverli scartare, con una conseguente perdita di flessibilità nel posizionamento.

Flessibilità necessaria se si vuole avere la possibilità di posizionare ed eventualmente recuperare la stazione in tutti gli scenari possibili.

Sia che si tratti di una zona remota che di un area prossima ad un cratere vulcanico durante un'eruzione.

Il vantaggio di avere questa possibilità comporta una netta diminuzione della necessità di eventuali interventi umani con un conseguente aumento in termini di sicurezza del personale.

## 2.6 Possibili scenari di funzionamento

Dopo questa analisi, è ormai chiaro che anche questa soluzione ha dei limiti.

Non si può abbandonare la stazione e lavorare continuamente da remoto perché questo farebbe scaricare la batteria.

Così come non si può dotare di un accumulatore maggiore perché altrimenti non sarebbe più possibile trasportarla con dei droni.

La chiave di tutto sta nell'aver ben chiaro lo scenario all'interno del quale la stazione dovrà funzionare.

Se si ha la necessità di abbandonare la stazione per lunghi periodi allora si dovranno limitare o azzerare le interazioni.

Preoccupandosi poi di recuperarla per prelevare i dati.

Se al contrario, la durata di funzionamento non è un parametro fondamentale o comunque rientra nei limiti della capacità della batteria, allora si potrà decidere di impostarla in maniera tale che trasmetta di continuo.

Un'ulteriore possibilità, che potrebbe essere una via di mezzo delle precedenti, sarebbe quella di lasciarla in attesa di un determinato evento o di un comando e non appena questa condizione si verifica la stazione provvederà in automatico alla trasmissione dei dati richiesti.

# Capitolo 3

## Stato dell'arte

Non appena terminata la fase di documentazione ed appurata l'effettiva padronanza delle nozioni richieste, il passaggio successivo non può che consistere in una loro applicazione pratica.

Per fare ciò è stato prima necessario preparare le due ipotetiche stazioni, configurare tutto il software necessario e predisporre tutti gli strumenti utili per la fase di test.

### 3.1 Installazione e configurazione delle librerie software

Lo sviluppatore canadese TMRh20 ha rilasciato tramite il blog personale [31] e un profilo su github [32] delle librerie software fondamentali per lo sviluppo di questo progetto.

All'indirizzo <http://tmrh20.github.io/RF24Installer/RPi/install.sh> è presente uno script d'installazione in bash che si occuperà di installare tutte le dipendenze (git e libncurses5-dev) e di compilare le librerie necessarie: RF24, RF24Network, RF24Mesh e RF24Gateway.

Queste 4 librerie servono a ricreare lo stack di rete seguendo il modello OSI.

NRF24L01 Radio Modules	OSI Layer 1 (Physical)
RF24 Core Radio Driver	OSI Layer 2 (Data Link)
RF24Network	OSI Layer 3 (Network)
RF24Ethernet (uIP)	OSI Layer 4 (Transport)
RF24Ethernet (core)	OSI Layer 5 (Session)
RF24Mesh -Optional-	OSI Layer 7 (Application)

Tabella 3.1: Stack di rete TCP/IP over rf24[31]

Di sicuro si sarà notata la presenza di una nuova libreria, RF24Ethernet e l'assenza di RF24Gateway.

Questo è dovuto al fatto che le librerie sono state inizialmente sviluppate per *Arduino* e successivamente estese anche ad altre piattaforme.

Per questo motivo è anche possibile creare una rete "ibrida", composta sia da macchine Linux con a bordo RF24Gateway che da microcontrollori con RF24Ethernet.

Per questo progetto, non avendo a che fare con microcontrollori si è quindi utilizzata solamente RF24Gateway.

L'installazione è piuttosto semplice e lo script si occuperà di recuperare, compilare ed installare tutti i sorgenti necessari.

Eventualmente, se lo si ritiene necessario, si possono anche scaricare le singole librerie recuperando gli indirizzi all'interno dello script.

Detto questo, una volta in possesso di tutti i sorgenti, prima di procedere è necessario apportare alcune modifiche ad un file di configurazione ed al codice sorgente di un'applicazione.

### 3.1.1 RF24Network\_config.h

La modifica di questo file è uguale per tutti i nodi della rete e consiste nel cambiare il valore della costante `MAX_PAYLOAD_SIZE` aumentandola a 1500 Byte.

Successivamente sarà sufficiente ricompilare e reinstallare la parte restante dello stack: RF24Network, RF24Mesh e RF24Gateway.

Questa procedura è stata resa più veloce grazie all'utilizzo di un makefile presente all'interno di ogni directory.

L'unica accortezza è quella di aggiungere il flag `-B` per forzare la ricompilazione e l'opzione `-C` seguita dall'indirizzo della libreria che si sta ricompilando.

```
sudo make install -B -C directory/libreria
```

### 3.1.2 RF24Gateway

A differenza del file di configurazione di RF24Network, le modifiche che seguono devono essere necessariamente diverse per ogni singolo nodo della rete ed andranno ad assegnare a ognuno un ID e un indirizzo IP univoci.

I sorgenti delle applicazioni si possono trovare nelle sotto directory `example` e `ncurses` della libreria RF24Gateway.

Entrambe hanno il compito di configurare un'interfaccia di rete TUN tramite la quale sarà poi possibile utilizzare i moduli radio per iniziare a scambiare informazioni.

L'unica differenza sta nell'interfaccia grafica. Una ne è completamente priva mentre la seconda, grazie alla libreria `ncurses`, fornisce qualche dettaglio in più riguardo lo stato dell'interfaccia di rete e dà la possibilità di riconfigurarla durante l'esecuzione.

Le variabili interessate sono `nodeID` e `ip`. La prima accetta dei valori compresi tra 0 e 255; tuttavia il valore 0 (impostazione di default) è riservato per quella stazione che andrà a ricoprire il ruolo di nodo radice di tutta la rete.

Questa restrizione non è presente per il valore dell'indirizzo ip che può essere scelto arbitrariamente.

In aggiunta a queste variabili, nel caso se ne presentasse la necessità, è possibile variare anche l'impostazione del datarate e dell'amplificatore di segnale che per default sono impostati a 1Mbit/s ed al valore massimo.

Le chiamate necessarie per fare questo sono `radio.setDataRate()` e `radio.setPALevel()` ed i parametri che accettano sono rispettivamente:

- RF24\_250KBPS, RF24\_1MBPS e RF24\_2MBPS.
- RF24\_PA\_MIN, RF24\_PA\_LOW, RF24\_PA\_HIGH e RF24\_PA\_MAX.

L'unica accortezza è quella di aggiungerle dopo che il modulo radio sia stato abilitato.

Per fare ciò basterà cercare la chiamata `gw.begin()` all'interno del `main` ed aggiungerle subito dopo.

La possibilità di far variare questi parametri durante l'esecuzione ha permesso di velocizzare le operazioni di test.

Evitando di dover compilare più copie e quindi terminare e rilanciare l'applicazione per effettuare un piccolo cambiamento.

Per sfruttare questo vantaggio bisogna prima modificare il sorgente dell'applicazione d'esempio, situato nella sotto-directory `example/ncurses/`, ed aggiungere due funzioni che si occuperanno di far variare i parametri desiderati.

Le differenze tra la versione originale e quella modificata sono elencate qui di seguito:

```

68a69,71
>
> void togglePAlevel(void);
> void toggleDataRate(void);
209a213,215
>
>     case 'p': togglePAlevel();break;
>     case 'd': toggleDataRate();break;
502a509,542
> /*****
>
> void togglePAlevel(){
>     uint8_t pa=radio.getPALevel();
>     switch(pa){
>         case RF24_PA_MIN:
>             radio.setPALevel(RF24_PA_LOW);
>             break;
>         case RF24_PA_LOW:
>             radio.setPALevel(RF24_PA_HIGH);
>             break;
>         case RF24_PA_HIGH:
>             radio.setPALevel(RF24_PA_MAX);
>             break;
>         case RF24_PA_MAX:
>             radio.setPALevel(RF24_PA_MIN);
>             break;
>     }
> }
>
> void toggleDataRate(){
>     rf24_datarate_e dr=radio.getDataRate();
>     switch(dr){
>         case RF24_250KBPS:
>             radio.setDataRate(RF24_1MBPS);
>             break;
>         case RF24_1MBPS:
>             radio.setDataRate(RF24_2MBPS);
>             break;
>         case RF24_2MBPS:
>             radio.setDataRate(RF24_250KBPS);
>             break;
>     }
> }

```

Tabella 3.2: Diff RF24Gateway\_ncurses.cpp

Oltre alle due funzioni è stata ovviamente aggiunta la possibilità di cambiare i valori, sfruttando le operazioni di input date da ncurses. Alla pressione delle lettere **p** e **d** corrispondono rispettivamente un cambio del valore dell'amplificatore ed uno del datarate.

## 3.2 Strumenti adottati

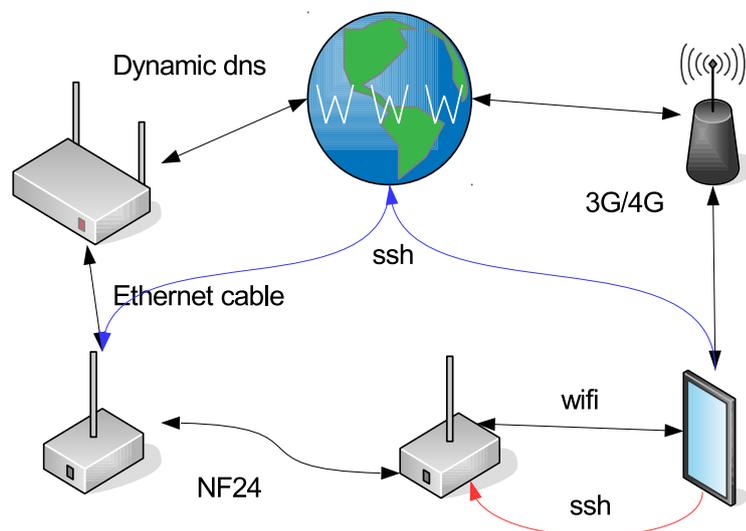


Figura 3.1: Schema connessioni test

Prima di effettuare qualunque test è necessario ovviamente immaginare uno scenario e rapportarsi con l'ambiente che sia ha disposizione.

Questa parte è stata illustrata in 4.1 e tenendo conto di ciò che si è avuto a disposizione si sono predisposte tutte le strumentazioni.

Ovviamente le stazioni dovevano essere due, entrambe composte da 1 *Raspberry Pi B+* ed 1 modulo radio *NRF24*.

La stazione "fissa" era altresì dotata di alcuni demoni in attesa di ricevere dati; quali ssh, ftp e netcat.

In più, per avere il pieno controllo è stata collegata via rete cablata ad un modem-router iscritto ad un servizio di dynamic dns e sfruttando il port forwarding (disponibile in qualunque modello anche di fascia bassa) si è abilitato l'accesso via ssh anche dall'esterno della rete locale.

Questo ha consentito di controllare istante per istante l'andamento dei test.

Differentemente, la stazione "mobile" è stata alimentata da una batteria portatile da 3000mAh, che ne assicura il funzionamento per  $\approx 7$  ore, e collegata ad una scheda di rete wifi.

Questa configurazione ha permesso di poter accedere alla stazione, sempre via ssh, ed effettuare i test, controllandone immediatamente l'esito.

Infine, come terminale di controllo è stato utilizzato uno smartphone con sistema operativo *Android*.

Grazie al quale è stato possibile avere l'accesso sia alla stazione "mobile", condividendo la connessione wifi, sia a quella "fissa" utilizzando la rete cellulare.

Queste operazioni sono state possibili grazie all'impiego alcune applicazioni:

- JuiceSSH.

Che permette di avere diversi terminali aperti contemporaneamente.

- Automate.

Applicazione che permette di creare ed eseguire processi, anche complessi, sfruttando i diagrammi di flusso e i "blocchi" che mette a disposizione.

Quest'ultima applicazione è stata usata per convertire le coordinate gps fornite dal ricevitore interno dello smartphone in unità di lunghezza.

Inizialmente il diagramma aspettava di avere una lettura gps con precisione inferiore a 3 metri.

Successivamente, dopo aver salvato il punto di partenza inviava una notifica non appena la distanza era prossima a multipli di 100m.

Una volta visualizzato l'avviso, attendeva un input da parte dell'utente.

Se questo era negativo ricominciava il ciclo calcolando la nuova distanza.

Altrimenti salvava le coordinate (raggiungimento del punto per i test) e iniziava il ciclo attendendo un nuovo responso. **3.3**

Per il calcolo della distanza tra due punti è stata usata la formula:

$$d = r * \arccos[\sin(latA) * \sin(latB) + \cos(latA) * \cos(latB) * \cos(lonA - lonB)]$$

Con  $r = 6372,795477598$  Km (raggio quadratico medio) e  $latA, lonA, latB, lonB$  rispettivamente latitudine e longitudine del primo e del secondo punto espressi in radianti.

Quindi dal momento che le letture erano espresse in gradi è stato necessario prima convertirle tramite:  $rad = (grad * \pi) / 180$ .

```
wait(precision<=3)
latA=latitude
lonA=longitude
dist=0
label: location
while(dist%100<=80){
    latB=latitude
    lonB=longitude
    dist=distance(latA,lonA,latB,lonB)
}
x=alert.response
if(x==false){
    sleep 5sec
    goto location
}
else{
    store latB
    store lonB
    goto location
}
```

Tabella 3.3: pseudo codice del diagramma di flusso

### 3.3 Script

La parte finale dei preparativi è stata quella in cui si sono messi a punto alcuni script in **bash** utili all'automatizzazione delle operazioni.

Prima di fare questo è stato comunque necessario installare i server per **ftp** e **ssh** e configurare quest'ultimo per l'autenticazione tramite chiave privata.

Questa operazione, eseguita sulla stazione "mobile", è piuttosto semplice e permette l'autenticazione senza la necessità di inserire alcuna password.

I comandi necessari sono messi a disposizione dai pacchetti **openssh**.

- Generazione delle chiavi pubblica e privata:  
ssh-keygen -b 4096
- Copia della chiave pubblica nella stazione "fissa"  
ssh-copy-id -i ~/.ssh/id\_rsa.pub pi@IP-STAZIONE-FISSA

Terminata l'installazione del software necessario si è quindi passato alla stesura del seguente script:

```
#!/bin/bash
function usage {
    echo "Executed $0 $TEST $DATARATE $PA
Usage: $0 -t test-type -b data-rate -p pa-level -u username -d direcotory -h host
test-type= scp | sftp | ftp | nc
data-rate= 0 | 1 | 2
pa-level= 1 | 2 | 3 | 4"
    exit 1
}
while getopts ":t:d:p:u:b:h:" opt;do
    case $opt in
        t)
            TEST=$OPTARG
            if [ -z $TEST ] || [[ ! $TEST =~ nc|scp|ftp|sftp|iperf ]];then
                usage
                exit 1
            fi
            ;;
        b)
            DATARATE=$OPTARG
            if [ -z $DATARATE ] || [[ ! $DATARATE =~ [0-2] ]];then
                usage
                exit 1
            fi
            ;;
        p)
            PA=$OPTARG
            if [ -z $PA ] || [[ ! $PA =~ [1-4] ]];then
                usage
                exit 1
            fi
            ;;
        u)
            USER=$OPTARG
            if [ -z $USER ];then
                usage
                exit 1
            fi
            ;;
        d)
            DEST=$OPTARG
            if [ -z $DEST ];then
                echo -e "Destination directory not set.\nUsing\
/home/$USER/nrf24-file-transfer/"
                DEST="/home/$USER/nrf24-file-transfer/"
            fi
            ;;
        h)
            HOST=$OPTARG
            if [ -z $HOST ];then
                usage
                exit 1
            fi
            ;;
        *)
            echo "Wrong option: $OPTARG"
            exit 1
            ;;
    esac
done
if [ -z $TEST ] || [ -z $DATARATE ] || [ -z $PA ];then
    usage
    exit 1
fi
#only real
TIMEFORMAT=%lR
i=1
#ordering by ascending size
for f in `ls -Sr file*`
do
    echo "----- $TEST test number; $i -----" >> log

```

```

echo "transfer $f with $TEST"
case "$TEST" in
  scp)
    TIME='{ time (scp -i ~/.ssh/id_rsa $f $USER@$HOST:$DEST &>>log) } 2>&1'
    ;;
  sftp)
    TIME='{ time (echo -e "put $f $DEST$f\nquit"|sftp -i ~/.ssh/id_rsa -o\
'StrictHostKeyChecking=no' -o 'ConnectTimeout=30' -q $USER@$HOST &>>log) } 2>&1'
    ;;
  ftp)
    TIME='{ time (echo -e "open $HOST\nuser $USER raspberry\nidle 30\nput \
$f $DEST$f\nquit"|ftp -n &>>log) } 2>&1'
    ;;
  nc)
    TIME='time (cat $f|nc -v $HOST 7770 &>>log) 2>&1'
    sleep 1.5s
    ;;
  iperf)
    iperf3 -V -t60 -O 10 -c $HOST --get-server-output --logfile log
    ;;
esac
echo -e "\ttransfer completed in $TIME"
echo -e "\nLog:"
cat log
echo "$f $TIME dr=$DATARATE pa=$PA" >> $TEST-test
i=$(( i + 1 ))
done
mv log log-$TEST-$DATARATE-$PA

```

Tabella 3.4: Script per l'esecuzione dei test

In aggiunta a questo anche un'ultimo script che richiama il precedente per ogni singola modalità di trasferimento.

```

#!/bin/bash
for t in iperf nc ftp sftp scp
do
  ./tests.sh -u ste -h 10.10.2.1 -b 0 -p 4 -t $t
done

```

Tabella 3.5: Automatizzazione dello script 3.4

# Capitolo 4

## Test e analisi dei risultati

Col fine di avere un'immagine chiara del reale funzionamento di queste stazioni è stato necessario procedere con alcuni test.

Per effettuare le prove si è dovuto prima immaginare uno scenario all'interno del quale le stazioni potrebbero trovarsi e successivamente preparare dei test rappresentativi di un'utilizzo reale.

### 4.1 Scenario

L'ambiente tipo in cui potrebbe trovarsi una stazione è paragonabile ad un classico paesaggio di campagna.

Dove non necessariamente si hanno ampi spazi liberi all'altezza del suolo, quindi le condizioni ottimali indicate dall'azienda produttrice dei moduli sono difficilmente ottenibili.

Per simulare questo scenario ci si è quindi recati in un classico parco cittadino.

La stazione "fissa" è stata posizionata al secondo piano di un edificio, sul davanzale di una finestra posta di fronte l'ingresso del parco mentre con la seconda, all'interno di un apposito box, ci si è allontanati gradualmente effettuando tutti i test a distanze multiple di 100m.

L'altezza della stazione fissa ha un impatto minimo sul calcolo della distanza; l'errore portato da questa altezza è infatti pari a  $\approx 40\text{cm}$  per la prima misurazione, valore che ovviamente diminuirà all'aumentare della distanza.

Questo scenario potrà non essere perfettamente fedele ad uno reale ma con buona approssimazione i risultati saranno molto simili.

Si è infatti tenuto conto di possibili ostacoli, qui rappresentati dalla presenza di alberi lungo la linea immaginaria di comunicazione.

L'unico fattore che potrebbe aver disturbato la trasmissione dei dati è la presenza massiccia di reti wifi 802.11.

Questi moduli comunicano infatti sulle stesse frequenze (2,4 Ghz), liberalizzate e aperte all'utilizzo di chiunque.

Tuttavia questo aspetto non è stato approfondito dal momento che non rientra negli scopi di questa ricerca.

## 4.2 Test

Dal momento che oltre a volere la possibilità di un accesso remoto si può anche avere la necessità di trasferire dei dati, sono stati presi in considerazione i protocolli più utilizzati per questo scopo.

Nello specifico si è proceduto ad effettuare i test utilizzando sia una trasmissione dati criptata che una in chiaro.

Per la parte criptata è stata usata la suite messa a disposizione dal pacchetto `openssh-server` contenente oltre al server `ssh` anche le applicazioni `scp` e `sftp`.

Per la parte in chiaro invece si è impiegato `ftp` e `netcat`.

Con l'intento di avere dei dati il più rappresentativi possibile si è scelto di effettuare 5 trasferimenti di file di dimensioni sempre crescenti: 32KB, 64KB, 128KB, 256KB e 512KB.

Per ogni protocollo utilizzato sono stati inviati dalla stazione "mobile" a quella "fissa" tutti e 5 i file, ripetendo l'operazione ogni 100m.

Sfortunatamente, contro ogni aspettativa i due moduli hanno smesso di comunicare ad una distanza ampiamente inferiore a quella indicata; gli unici trasferimenti che si sono conclusi con successo sono stati quelli compresi tra i 100m e i 300m, estremi inclusi.

Il risultato meno atteso, oltre le piccole distanze coperte, è stata la totale assenza di trasmissione con le impostazioni intermedie. Ricordiamo che i moduli dovrebbero essere in grado di coprire una distanza di 520m con un datarate di 2Mb/s, 750m a 1Mb/s e 1000m a 250Kb/s.

Contrariamente a questa indicazione, gli unici test conclusi sono stati condotti con 2Mb/s a 100m e 250kb/s a 200m e 300m. Probabilmente questi moduli oltre alle interferenze delle reti vicine, come accennato in [4.1](#), soffrono molto anche in presenza ostacoli; infatti l'unico test condotto con il datarate massimo oltre ad essere il più vicino è stato anche l'unico a non avere alcun ostacolo in linea d'aria.

Detto ciò, dai risultati dei test emerge che con un datarate teorico di 2 Mbit/s, la larghezza di banda effettiva si attesta intorno ai 100 kbit/s, per poi scendere intorno ai 40 kbit/s con un datarate di 250 kbit/s.

Le performance migliori si sono ottenute, come da aspettative, utilizzando netcat.

Questo perché a differenza degli altri protocolli, la quantità di informazioni aggiuntive (overhead) utili alla gestione della comunicazione sono mantenute al minimo.

Immediatamente dopo si trovano `ftp` seguito da `sftp` e `scp`. Non stupisce che gli ultimi due siano quelli che fanno uso della crittografia, infatti oltre a far aumentare l'overhead sono responsabili di un aumento di carico per la cpu, quindi sconsigliabili per l'utilizzo che si vuole fare di questa stazione.

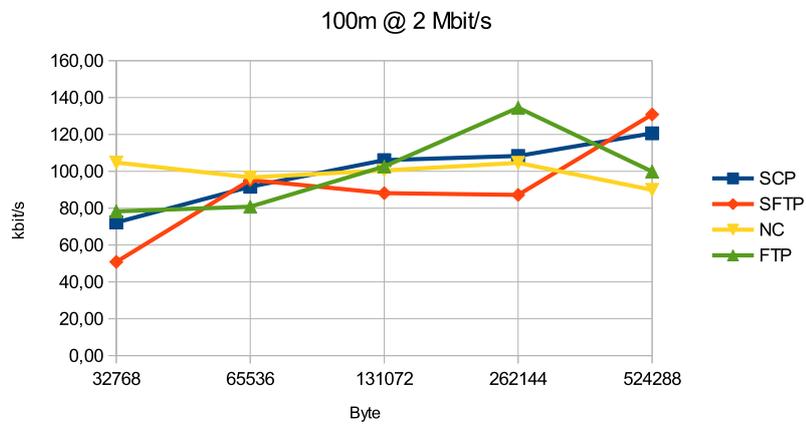


Figura 4.1: Trasferimenti a 100m e 2Mbit/s



Figura 4.2: Trasferimenti a 200m e 250kbit/s

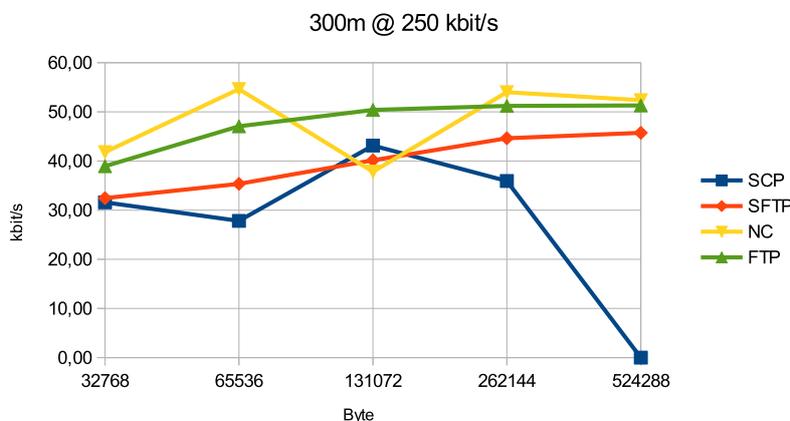


Figura 4.3: Trasferimenti a 300m e 250kbit/s

### 4.2.1 Considerazioni

Purtroppo le informazioni relative a questi moduli sono piuttosto frammentate e la confusione è aumentata dalla presenza di due modelli differenti, quello utilizzato per questo progetto ed un secondo privo di antenna e amplificatore.

Ovviamente il fabbisogno energetico differisce ma non è ancora chiaro di quanto.

Per questo motivo, per evitare problemi durante la trasmissione o scarsa copertura (cosa che si è verificata durante i test), si consiglia di dotare i moduli di un'alimentazione separata e non collegarla direttamente alla piattaforma. Nello specifico, il *Raspberry Pi* è in grado di erogare 16mA su un singolo pin GPIO, massimo 50mA tramite il pin 1 o 17 (3,3 Volt) e ipoteticamente nessun limite tramite i pin 2 e 4 (5 Volt).

Questo nel caso in cui sia collegato un solo pin dei 38 rimanenti (esclusi i pin 2 e 4).

Nel caso in cui vi fossero più collegamenti, la somma di tutte le correnti erogabili dai pin è limitata a 50 mA (meno il 2 e il 4), oltre questo limite il SOC Broadcom potrebbe surriscaldarsi e danneggiarsi.



# Conclusioni

Questo progetto è nato con lo scopo di creare un'alternativa economica e sacrificabile alle attuali stazioni di rilevamento ambientale.

Un'alternativa necessaria per permettere lo studio ravvicinato di processi geofisici ad alto potere distruttivo.

Nella ricerca di possibili soluzioni si sono tenuti in considerazione svariati parametri.

Il primo ed in assoluto il più importante riguarda la sicurezza del personale, il quale deve necessariamente avere la garanzia di operare in totale sicurezza. Questo requisito principe ci ha posto davanti ad una cascata di problematiche alle quali bisognava trovare una soluzione.

La ricerca si è quindi svolta in un costante trade-off tra funzionalità e flessibilità.

Una possibile alternativa è stata trovata nell'utilizzo di single-board computer collegati in rete tramite comunicazioni in radiofrequenza.

Successivamente si è quindi proceduto ad analizzare le performance delle trasmissioni dati per valutarne l'effettiva usabilità in scenari reali.

L'analisi dei risultati, successiva alle attività di test ha portato a trarre le seguenti conclusioni:

Nonostante le scarse prestazioni in termini di distanze coperte e larghezza di banda disponibile, le performance ottenute si ritengono sufficienti a supportare le attuali stazioni di rilevamento.

Le motivazioni che hanno portato a questa osservazione risiedono nell'ot-

timo risultato raggiunto in termini di costi, neanche paragonabili alle attuali soluzioni impiegate.

Grazie a questo aspetto è possibile infatti coprire ampie aree moltiplicando il numero di stazioni "usa e getta" e contemporaneamente mantenendo al minimo le spese da affrontare.

In aggiunta a questo si è considerato anche la quantità di dati che potenzialmente debbano essere trasmessi.

Essendo questi principalmente di piccole dimensioni, si ha la possibilità di saturare la banda disponibile continuando comunque a mantenere un flusso di dati sufficiente per una possibile attività di monitoraggio.

Questo lavoro non si propone come migliore soluzione ai problemi illustrati ma bensì vuole porre le basi per possibili sviluppi futuri e miglioramenti.

Uno di questi potrebbe essere l'implementazione di un client per il protocollo *MQTT*, ampiamente utilizzato nel settore dell'Internet Of things (IOT).

Settore che non ha a disposizione grandi capacità di calcolo e per questo richiede una larghezza di banda limitata.

# Bibliografia

- [1] <https://en.wikipedia.org/wiki/Geohazard>
- [2] <https://it.wikipedia.org/wiki/Frana>
- [3] <http://dc.engconfintl.org/geohazards/>
- [4] <http://www.sgs.org.sa/english/naturalhazards/pages/geohazard.aspx>
- [5] <http://www.sustainabledevelopmentinfo.com/what-exactly-is-a-geohazard/>
- [6] <http://www.sapere.it/sapere/strumenti/studiafacile/scienza/La-tettonica/La-sismicit-/La-classificazione-dei-terremoti.html>
- [7] <https://it.wikipedia.org/wiki/Terremoto>
- [8] <http://www.swisseduc.ch/stromboli/volcano/seismik/explanations-it.html>
- [9] <https://ingvterremoti.wordpress.com/il-monitoraggio-sismico/>
- [10] <http://www.gm.ingv.it/index.php/rete-sismica-nazionale/la-stazione-di-monitoraggio>
- [11] <http://www.fis.uniurb.it/miro/fssb/fssb.htm>

- 
- [12] <http://slideplayer.it/slide/565228/>
- [13] <http://www.earth-prints.org/bitstream/2122/8443/1/MorettiGovoni.pdf>
- [14] [http://www.idpa.cnr.it/ricerca\\_laboratori\\_MI.html](http://www.idpa.cnr.it/ricerca_laboratori_MI.html)
- [15] [http://www.earth-prints.org/bitstream/2122/5982/1/RapportoTecnicoINGV\\_137.pdf](http://www.earth-prints.org/bitstream/2122/5982/1/RapportoTecnicoINGV_137.pdf)
- [16] <http://www.trimble.com/Infrastructure/Trimble-REF-TEK-130S-01.aspx>
- [17] <http://www.everythingrf.com/products/gps-gnss-antennas/leica-geosystems/666-305-as10>
- [18] <ftp://ftp.ingv.it/pub/alberto.delladio/REF%20TEK%20Brochures/Recorder%20130S-01Brochure.pdf>
- [19] [http://www.s-telecom.ru/vendors/gsmdevice/docu/ds\\_UMTS%20Router%20UR5\\_eng.pdf](http://www.s-telecom.ru/vendors/gsmdevice/docu/ds_UMTS%20Router%20UR5_eng.pdf)
- [20] <http://www.slideshare.net/FelizianiAntonio/droni-e-campi-professionali-di-applicazione>
- [21] <http://istituto.ingv.it/1-ingv/amministrazione-trasparente-1/contenuti-at-vari/decreti-e-delibere-2015-dal-4ago/decreto-dg-481.2015.pdf>
- [22] [http://www.shomobil.com/product\\_info.php/language/en/info/p90\\_conel-ur5-v2b-set-umts-hsdpa-router--hutschiene-din-ts35---ts32.html/](http://www.shomobil.com/product_info.php/language/en/info/p90_conel-ur5-v2b-set-umts-hsdpa-router--hutschiene-din-ts35---ts32.html/)
- [23] <http://www.sunearthtools.com/it/tools/distance.php>
- [24] <http://elinux.org/RPiconfig>

- 
- [25] <https://www.raspberrypi.org/forums/viewtopic.php?f=28&t=84874>
  - [26] <http://tmrh20.github.io/>
  - [27] <http://tmrh20.github.io/RF24/>
  - [28] <http://tmrh20.github.io/RF24Network/>
  - [29] <http://tmrh20.github.io/RF24Mesh/>
  - [30] <http://tmrh20.github.io/RF24Gateway/>
  - [31] <http://tmrh20.github.io/RF24Ethernet/Overview.html>
  - [32] <http://tmrh20.blogspot.it/>
  - [33] <https://github.com/TMRh20>

