



UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Laurea Triennale in Ingegneria delle  
Telecomunicazioni

**Ricevitore MATLAB-SDR per  
Radio Data System nella radio FM**

Relatore:

**Prof. Marco Moretti**

Candidato:

**Filippo Gasperini**

ANNO ACCADEMICO 2023/2024

# Indice

<b>Introduzione .....</b>	<b>4</b>
<b>Capitolo 1 .....</b>	<b>7</b>
<b>Introduzione al Radio Data System .....</b>	<b>7</b>
1.1 Storia dello standard RDS .....	7
1.2 Funzionamento del RDS .....	9
1.2.1 Livello Fisico .....	9
1.2.2 Livello Data Link .....	14
1.2.3 Livello di Sessione e Presentazione.....	15
1.2.4 Descrizione dei servizi .....	17
<b>Capitolo 2 .....</b>	<b>19</b>
<b>Software Defined Radio e MATLAB .....</b>	<b>19</b>
2.1 RTL-SDR .....	19
2.1.1 Architettura RTL-SDR .....	20
2.2 MATLAB .....	21
<b>Capitolo 3 .....</b>	<b>22</b>
<b>Implementazione del Ricevitore RDS .....</b>	<b>22</b>
3.1 Codice MATLAB .....	22
3.2 Funzionamento del codice .....	24
3.2.1 Livello Fisico .....	24
3.2.2 Livello Data Link .....	28

3.2.3 Livello di Sessione e Presentazione.....	29
3.2.4 Analisi dei risultati.....	30
3.2.5 Simulazione .....	31
<b>Conclusioni .....</b>	<b>35</b>
<b>Ringraziamenti.....</b>	<b>37</b>
<b>Bibliografia e Sitografia .....</b>	<b>38</b>

# Introduzione

Le radiocomunicazioni hanno giocato un ruolo fondamentale nello sviluppo delle comunicazioni globali e nell'evoluzione della società moderna, con un impatto significativo sulle nostre vite e sul modo in cui riceviamo informazioni e intrattenimento.

La storia della radio ha inizio con le prime sperimentazioni alla fine del XIX secolo, quando scienziati e inventori come Guglielmo Marconi, Nikola Tesla e altri pionieri, iniziarono a esplorare le possibilità della trasmissione senza fili, dimostrando così la fattibilità delle comunicazioni wireless. Dopo la Prima Guerra Mondiale, la radio iniziò a essere usata per come la conosciamo oggi: iniziarono le trasmissioni di programmi sonori da parte di apposite stazioni emittenti, ascoltabili tramite apparecchi riceventi. Il primo servizio regolare di trasmissioni prese forma nel 1920 nel Regno Unito, per poi raggiungere gli Stati Uniti pochi mesi dopo, dove ebbe un successo rapidissimo. Le potenzialità erano evidenti: con la radio si poteva raggiungere il pubblico in qualsiasi località e in tempo reale. Nel nostro Paese l'ascesa della radio coincise con gli anni del regime fascista, dove il governo si rese presto conto delle potenzialità del nuovo strumento e ne promosse la diffusione in tutto il territorio nazionale. Già nel 1924 fu fondata l'Unione Radiofonica Italiana, che tre anni più tardi divenne Ente Italiano Audizioni Radiofoniche (EIAR), il diretto progenitore della Rai. Tra i programmi radiofonici trovarono presto spazio le cronache sportive, la diffusione della musica ed i talk, generando nuovi linguaggi e nuovi contenuti, adatti al pubblico ampio al quale si rivolgevano. [1]

Con il passare degli anni furono introdotti alcuni miglioramenti tecnici. L'innovazione più importante giunse nel 1935, quando Edwin Armstrong inventò la radio a modulazione di frequenza (FM), che si associò alla modulazione di ampiezza (AM) usata fino ad allora. Dagli anni '50 in avanti, la radio FM divenne sempre più popolare per la sua qualità audio superiore all'AM, e contribuì all'espansione delle trasmissioni stereo.

L'utilizzo di un numero sempre maggiore di frequenze per i programmi radiofonici nella gamma VHF/FM rese più difficile sintonizzare i ricevitori radio sul programma desiderato. Questo tipo di difficoltà venne risolto con l'introduzione del Radio Data System (RDS) nel 1984, che consentì alle stazioni radio di trasmettere dati digitali insieme al segnale audio, fornendo agli ascoltatori informazioni aggiuntive come il nome della stazione, il titolo del brano, le informazioni sul traffico e molto altro ancora. Questo miglioramento dell'esperienza radiofonica contribuì a consolidare ulteriormente il ruolo della radio nella vita quotidiana. [2]

Dagli anni '90 fino ai giorni nostri, con l'avvento della digitalizzazione, le trasmissioni radio si sono ulteriormente diversificate. È emerso il Digital Audio Broadcasting (DAB o DAB+) come nuova tecnologia per la diffusione di contenuti radiofonici. Il DAB+ è uno standard di radiodiffusione digitale che permette la trasmissione sonora con qualità paragonabile a quella di un compact disc, cioè audio stereofonico PCM campionato a 44,1 kHz con campioni di 16 bit. Questa tecnologia ha aperto nuove possibilità per la trasmissione radio, consentendo minore influenza di interferenze sul segnale e miglioramento dei servizi multimediali, con la possibilità di trasmettere immagini (principalmente il logo delle radio). [3]

Nonostante l'avvento del DAB e di altre tecnologie digitali, la radio FM ha dimostrato una notevole resilienza nel panorama delle trasmissioni radiofoniche, e ciò è riconducibile a diversi fattori. In primo luogo, la radio FM ha consolidato negli anni una solida presenza commerciale e infrastrutturale, motivo per cui le emittenti continuano ad investire in reti di trasmissione FM. Gli apparecchi radio FM sono presenti in numerosi dispositivi, tra cui autoradio, radio portatili, stereo domestici e smartphone dotati di ricevitori FM. Questa ampia adozione garantisce l'accessibilità del servizio a milioni di ascoltatori in tutto il mondo, senza la necessità di costosi aggiornamenti tecnologici. Inoltre, la copertura della radio FM è generalmente ampia e affidabile, con una diffusione che raggiunge anche aree remote o con connettività Internet limitata.

Questa tesi si propone di esplorare in dettaglio il Radio Data System nella radio FM, combinando elementi teorici con un approccio sperimentale, attraverso l'implementazione di un codice MATLAB per la ricezione e demodulazione del segnale RDS. Questo è reso possibile utilizzando la piattaforma di programmazione MATLAB in combinazione con una Software Defined Radio, nello specifico una RTL-SDR, tecnologia che utilizza un dispositivo USB basato sul chip Realtek RTL2832U. Questo approccio consentirà di acquisire dati in tempo reale dalla radio FM, demodulare il segnale e analizzare le informazioni contenute nel RDS.

Nel primo capitolo si parlerà della storia dello standard RDS, per poi approfondirne gli aspetti teorici, esaminando le specifiche che definiscono il funzionamento del sistema; nel secondo capitolo si discuteranno i concetti fondamentali delle SDR, illustrando come questa tecnologia consenta di utilizzare un ricevitore radio software. Sarà esaminata l'interazione tra la SDR-RTL e MATLAB. Nel terzo capitolo verrà presentata l'implementazione pratica del codice MATLAB in esame, mostrando come i dati RDS possano essere acquisiti, elaborati e interpretati per estrarre le informazioni trasmesse. Nelle conclusioni saranno evidenziati i risultati raggiunti, per poi trattare le possibili evoluzioni del RDS, analizzando il futuro delle radio tradizionali.

# Capitolo 1

## Introduzione al Radio Data System

### 1.1 Storia dello standard RDS

L'idea alla base del Radio Data System fu proposta dall'ente nazionale di radiotelevisione pubblica francese (ORTF), ora RadioFrance, a metà degli anni '70. La proposta si ispirava a un sistema di identificazione di informazioni sul traffico chiamato Autofahrer-Rundfunk-Informationssystem (ARI), sviluppato in Germania dal centro di ricerca sulle trasmissioni pubbliche (IRT) e dal produttore di autoradio Blaupunkt. Il sistema ARI indicava all'ascoltatore la presenza di informazioni sul traffico manipolando opportunamente la sottoportante a 57 kHz del segnale radio FM. Il mandato preciso affidato all'Unione Europea di Radiodiffusione (UER o EBU in inglese) nel 1974 fu quello di proporre una tecnologia più flessibile, applicabile a tutte le trasmissioni FM e che permettesse una sintonizzazione automatica del ricevitore sulla frequenza con la migliore ricezione, nel caso in cui l'emittente stesse tramettendo su più frequenze. Lo sviluppo a livello europeo durò dieci anni, con molte prove sul campo e diverse soluzioni alternative proposte. Il tipo di modulazione venne ripreso da un sistema<sup>1</sup> svedese introdotto in precedenza, mentre la codifica in banda base era di nuova concezione, sviluppata principalmente dalla British Broadcasting Corporation (BBC) e dall'IRT tedesco. Tra i criteri più severi da rispettare si richiese che i dati aggiunti alla trasmissione FM fossero completamente impercettibili, anche nella modalità di ascolto più critica, cioè in cuffia, e si consentì solo un'insignificante interferenza aggiuntiva in radiofrequenza causata dal segnale RDS. Fu così che nel 1984 l'EBU pubblicò la prima specifica<sup>2</sup> sul RDS, che diede inizio all'introduzione della tecnologia nei ricevitori FM.

---

<sup>1</sup> Sistema MBS sviluppato dalla Swedish Telecommunications Administration

<sup>2</sup> EBU Tech 3244: Specification of the radio data system RDS for VHF/FM sound broadcasting

Sul finire degli anni '80 molti stati europei, tra cui Francia, Italia e Germania, avevano introdotto il RDS nelle trasmissioni FM, e divenne chiaro che uno standard riconosciuto a livello europeo sarebbe servito a pubblicizzare il sistema e a garantire un'attività di progettazione coerente tra le diverse organizzazioni che si occupavano dei prodotti RDS. Così, nel 1990, il Comitato Europeo di Normazione Elettrotecnica (CENELEC), in stretta collaborazione con l'EBU, pubblicò lo standard<sup>3</sup> definitivo per l'Europa. Questo divenne il punto di riferimento di cui il sistema RDS aveva bisogno, stabilendo requisiti comuni per gli apparecchi radio e le emittenti radiofoniche in tutta Europa.

Dopo l'adozione dello standard CENELEC, l'RDS iniziò a essere implementato su scala più ampia nelle stazioni radio in tutta Europa, il che portò all'arricchimento della tecnologia con nuove importanti funzionalità, e suscitò interesse anche tra gli ascoltatori. Nel 1993 nacque il Forum RDS, che ancora oggi fornisce ai suoi membri una rete di contatto per lo scambio di esperienze sull'uso e la corretta implementazione della tecnologia. Dal 1999 il Forum RDS gestisce gli standard RDS della Commissione Elettrotecnica Internazionale (IEC), e si occupa di mantenere il sistema RDS come originariamente concepito dall'EBU, ma anche aggiornarlo e preservare la piena retrocompatibilità con il gran numero di ricevitori RDS esistenti, per consentire l'implementazione di nuove funzionalità nelle nuove generazioni di ricevitori. Alcuni membri del Forum sono stati promotori di nuove funzionalità per il RDS successivamente standardizzate, come il RadioText+. Ne verrà data una breve trattazione nel secondo paragrafo di questo capitolo. Inoltre, si occupano dal 2015 della standardizzazione del RDS2, l'innovativo miglioramento del RDS che è in grado di trasportare molti più dati rispetto al predecessore; questo argomento sarà trattato nelle conclusioni.

L'introduzione del RDS ha aiutato le trasmissioni FM a diventare una tecnologia di grande successo, tanto da essere considerata una “rivoluzione silenziosa”. È ipotizzabile che vivrà almeno quanto la radio FM, per la quale non è ancora possibile prevedere uno spegnimento oggi, dato che molte auto ancora

---

<sup>3</sup> CENELEC EN 50067:1990



non hanno una radio digitale e la modalità di ricezione principale rimane FM/RDS. L'aggiornamento alla ricezione della sola radio digitale nelle auto è possibile, ma non è ancora molto diffuso a causa delle difficoltà tecniche e dei costi di manodopera necessari. Pertanto, le trasmissioni FM saranno ancora presenti nel 2030 e oltre. Uno spegnimento anticipato a livello europeo non è attualmente realizzabile, a parte quello già parzialmente avvenuto in Norvegia e previsto per la Svizzera a partire dal 2025. [2, 4]

## 1.2 Funzionamento del RDS

Il Radio Data System è applicato nelle trasmissioni sonore VHF/FM nella banda da 87,5 MHz a 108 MHz, che possono trasmettere programmi stereofonici (sistemi con tono pilota) o monofonici. Gli obiettivi principali del RDS sono quelli di migliorare il funzionamento dei ricevitori FM e di renderli più facili da usare, grazie a funzioni quali l'identificazione dei programmi (*Programme Identification*), la visualizzazione del tipo di servizio (*Programme Service*) e, se supportata, la sintonizzazione automatica. Queste informazioni di base implementate nel gruppo di tipo 0 (vedi sezione 1.2.3) non sono opzionali, a differenza di molte altre possibili funzioni del RDS. [5]

### 1.2.1 Livello Fisico

Il segnale RDS è trasportato su una sottoportante che viene aggiunta al segnale stereo multiplex, in ingresso al trasmettitore VHF/FM. In Figura 1 e 2 vengono mostrati i tipici schemi a blocchi utilizzati per generare il segnale RDS, rispettivamente in trasmissione ed in ricezione.

#### 1. Frequenza e fase della sottoportante:

Durante le trasmissioni stereo la sottoportante RDS è fissata alla terza armonica del tono pilota a 19 kHz, risultando così pari a 57 kHz. La sottoportante è inoltre in fase con la suddetta armonica, con una tolleranza consentita di  $\pm 10^\circ$ . La deviazione di frequenza introdotta dalla sottoportante sulla portante del segnale FM varia da

$\pm 1$  kHz a  $\pm 7.5$  kHz, con un valore consigliato di  $\pm 2$  kHz. La massima deviazione di frequenza del segnale FM multiplex deve essere pari a  $\pm 75$  kHz, come è noto dalla teoria del broadcasting FM.

## 2. Modulazione:

La sottoportante a 57 kHz è modulata in ampiezza dal segnale RDS opportunamente sagomato e codificato (vedi p.ti 5, 6). La sottoportante è soppressa. Questo metodo di modulazione è sostanzialmente una forma di BPSK (*Binary Phase Shift Keying*).

## 3. Frequenza di clock e data-rate:

La frequenza di clock è ottenuta dividendo la sottoportante per 48, ottenendo così un data-rate di 1187.5 Hz (o 1187.5 bit/s).

## 4. Codifica differenziale:

I bit generati dalla sorgente RDS sono codificati differenzialmente, seguendo la legge in tabella:

Output precedente (tempo $t_{i-1}$ )	Input (tempo $t_i$ )	Nuovo Output (tempo $t_i$ )
0	0	0
0	1	1
1	0	1
1	1	0

Legge di Codifica/Decodifica

Che consiste nel fare uno XOR (OR esclusivo) tra il bit in ingresso ed il bit codificato dell'istante precedente, dove per istante si intende periodo di clock, ossia  $\frac{1}{1187.5}$  s  $\simeq 0.84$  ms.

## 5. Simboli bifase:

La potenza del segnale RDS a radiofrequenza viene minimizzata<sup>4</sup> associando un simbolo bifase a ciascun bit codificato. Il processo di

---

<sup>4</sup> Questa tecnica fu implementata per non creare interferenze tra il segnale RDS ed il segnale ARI che, come citato nella sezione 1.1, occupava la sottoportante a 57 kHz. Il sistema ARI è stato ufficialmente dismesso il 1 Marzo 2005.

generazione dei simboli bifase viene schematizzato in Figura 1. In pratica ogni bit di ingresso dà luogo ad una coppia di impulsi dispari,  $e(t)$ :

$$e(t) = \begin{cases} \delta(t) - \delta(t - \frac{t_d}{2}) & \text{quando l'input è "1" logico} \\ -\delta(t) + \delta(t - \frac{t_d}{2}) & \text{quando l'input è "0" logico} \end{cases}$$

Dove  $t_d = \frac{1}{1187.5} s$  è il periodo di clock.

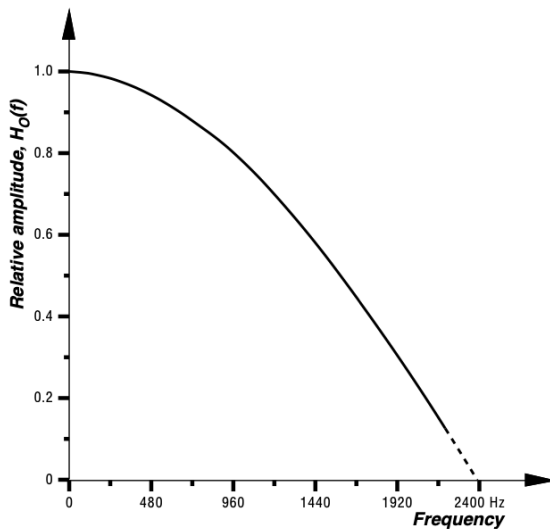
Questi impulsi sono ben visibili nel grafico ④ della Figura 1.

## 6. Filtri di trasmissione e ricezione:

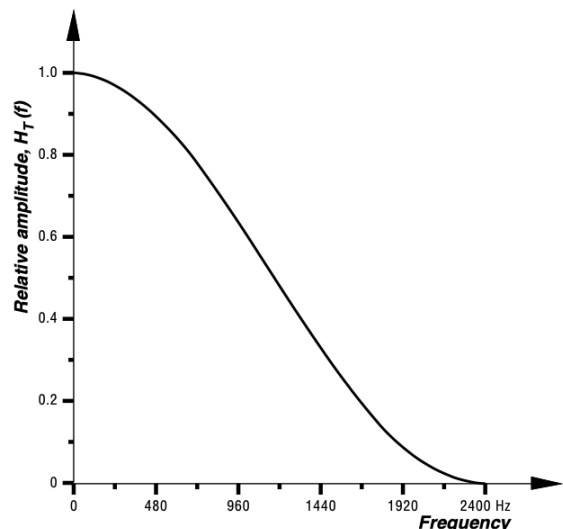
Lo spettro dei simboli bifase così ottenuti viene sagomato dal filtro  $H_0(f)$ , che conferisce al segnale una banda di circa 2.4 kHz (circa 5 kHz in banda passante):

$$H_0(f) = \begin{cases} \cos(\frac{\pi f t_d}{4}) & \text{se } 0 \leq f \leq \frac{2}{t_d} \\ 0 & \text{se } f > \frac{2}{t_d} \end{cases}$$

Va sottolineato che il filtro appena descritto viene poi moltiplicato per il filtro di ricezione, che è incluso nel decodificatore di simbolo ed è identico a quello di trasmissione. Questa operazione genera il filtro complessivo  $H_T(f)$ . Quanto descritto non è altro che una realizzazione di filtro adattato, che rispetta le condizioni di Nyquist e massimizza il rapporto segnale rumore sui simboli.



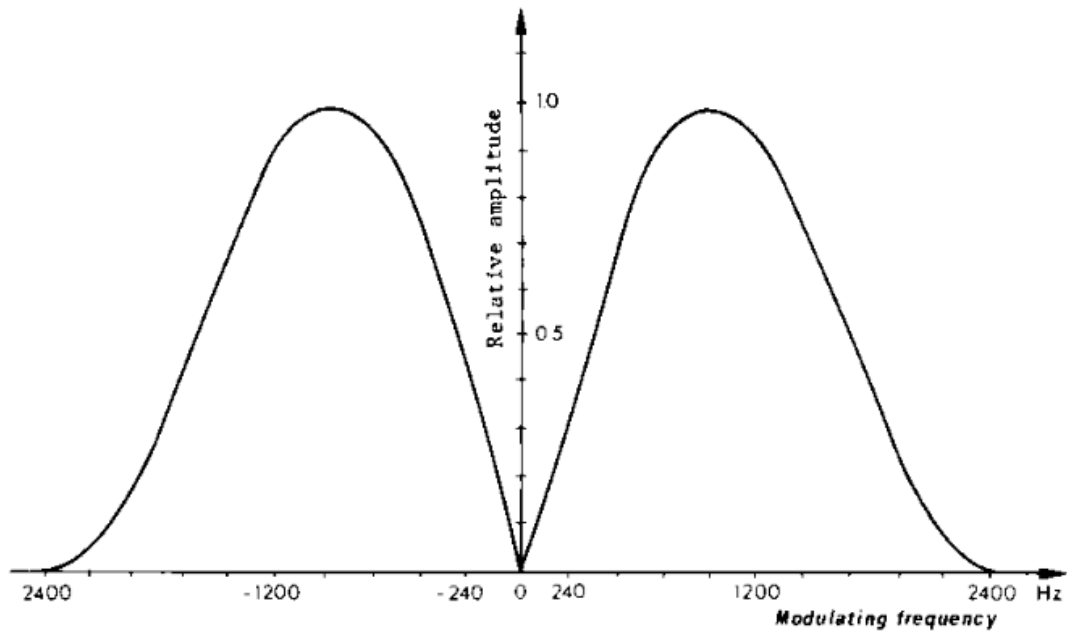
Risposta in ampiezza del singolo filtro di trasmissione/ricezione



Risposta in ampiezza complessiva

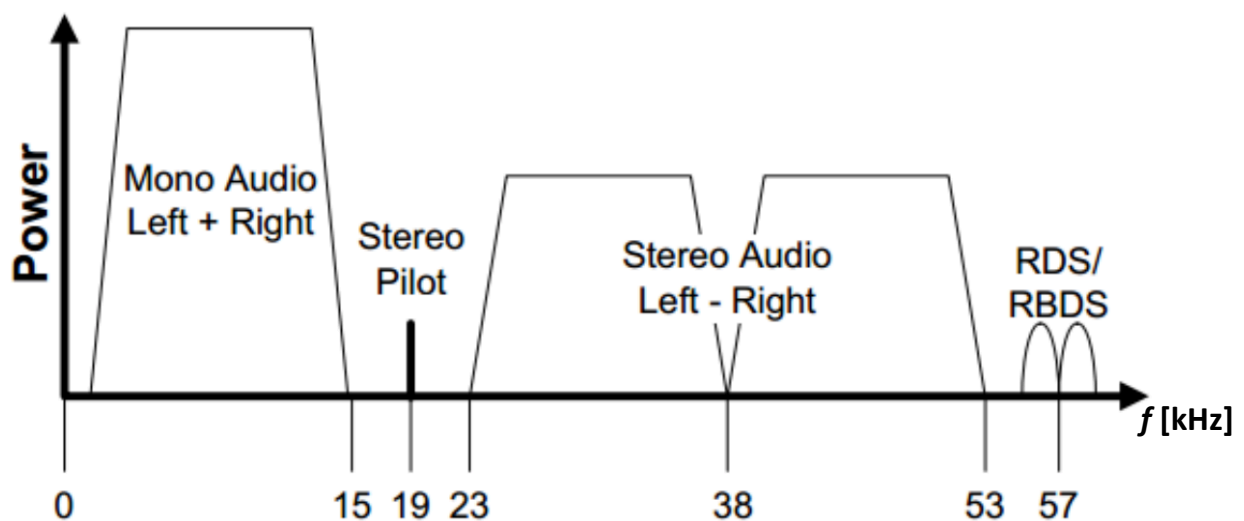


Le operazioni appena descritte, conferiscono allo spettro del segnale RDS trasmesso la seguente forma:



Spettro dei simboli RDS dopo la codifica e il filtraggio

Che viene inserito nel segnale multiplex FM Stereo alla frequenza centrale di 57 kHz:



Segnale multiplex FM Stereo

## 1.2.2 Livello Data Link

I messaggi RDS sono raggruppati secondo una precisa struttura prima di essere codificati e trasmessi, come si vede in Figura 3. L'elemento più largo della struttura è detto *gruppo*, ed è composto da 104 bit. Ogni gruppo comprende 4 *blocchi* da 26 bit ciascuno. I 26 bit di ogni blocco sono organizzati in 16 bit informativi (*Information word*) e 10 bit di controllo (*Checksum*).

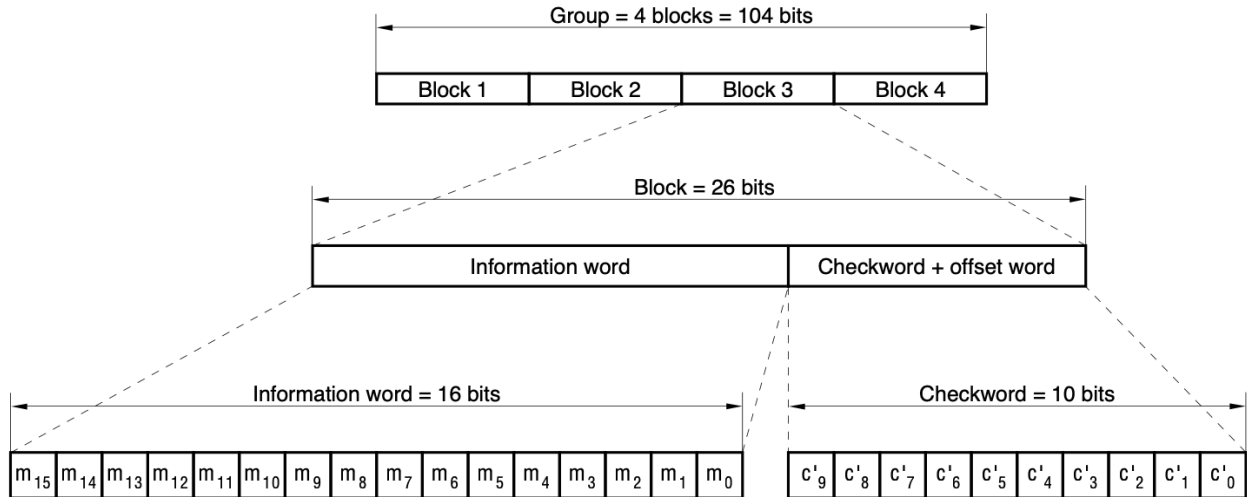


Figura 3: Struttura del gruppo RDS

Lo scopo dei 10 bit di controllo è quello di consentire al ricevitore di rilevare e correggere gli errori che si verificano durante la trasmissione. Inoltre, all'interno del controllo è inclusa la *offset word*, che serve sia ad identificare i blocchi all'interno di ciascun gruppo sia a correggere gli errori di sincronizzazione. La trattazione completa del funzionamento è discussa nei paragrafi 2.3, 2.4 dello standard CENELEC [5].

### 1.2.3 Livello di Sessione e Presentazione

Le caratteristiche del gruppo RDS sono illustrate in Figura 4. Tra le principali vi sono:

- Il primo blocco di ogni gruppo contiene sempre il codice *Program Identification* (PI).
- I primi 5 bit del secondo blocco specificano il tipo, che va da 0 a 15 con due versioni per ogni numero (i.e. 0A, 0B, 1A, 1B etc.). Questo comporta un totale di 32 tipi, descritti dalla Figura 5.
- I codici Programme Type (PTY) e Traffic Programme (TP) occupano posizioni fisse nel secondo blocco. Per questo, i codici PI, PTY e TP possono essere decodificati senza fare riferimento a nessun blocco al di fuori di quello che li contiene. Questo è essenziale per ridurre al minimo il tempo di acquisizione di questi messaggi.

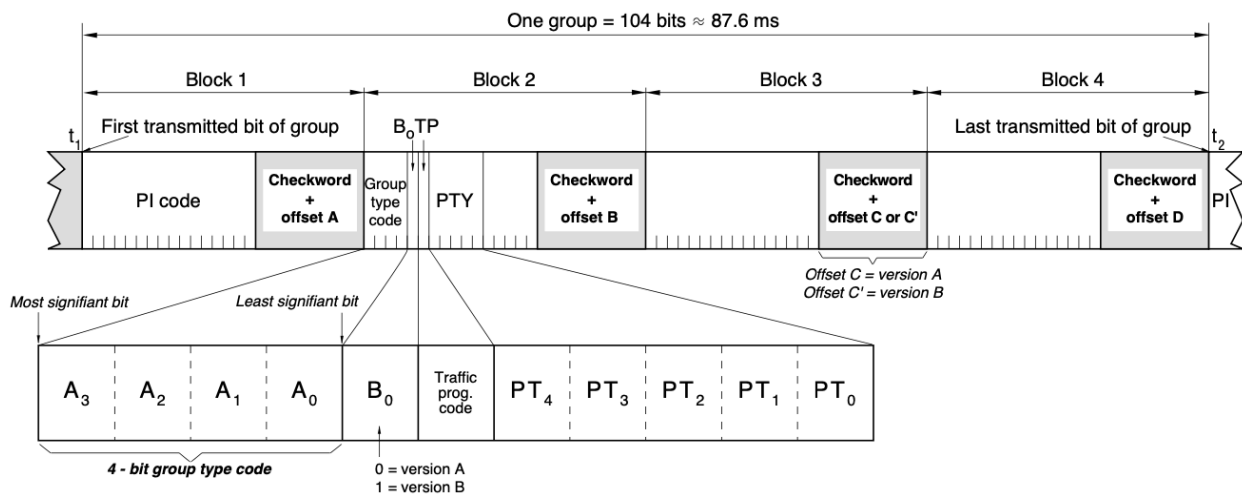


Figura 4: Dettaglio del gruppo RDS

<b>Group type</b>	<b>Group type code and version</b>	<b>Description</b>
0A	0000 0	Basic tuning and switching information with Programme Service name
0B	0000 1	Basic tuning and switching information with Programme Service name
1A	0001 0	Slow labelling codes
1B	0001 1	Open Data Applications
2A	0010 0	RadioText
2B	0010 1	RadioText
3A	0011 0	Application Identification for ODA and 16 bits of ODA data
3B	0011 1	Open Data Applications
4A	0100 0	Clock-time and date
4B	0100 1	Open Data Applications
5A	0101 0	Open Data Applications
5B	0101 1	Open Data Applications
6A	0110 0	Open Data Applications
6B	0110 1	Open Data Applications
7A	0111 0	Open Data Applications
7B	0111 1	Open Data Applications
8A	1000 0	Open Data Applications: Traffic Message Channel or if TMC not used, any other ODA
8B	1000 1	Open Data Applications
9A	1001 0	Open Data Applications
9B	1001 1	Open Data Applications
10A	1010 0	Programme Type Name
10B	1010 1	Open Data Applications
11A	1011 0	Open Data Applications
11B	1011 1	Open Data Applications
12A	1100 0	Open Data Applications
12B	1100 1	Open Data Applications
13A	1101 0	Open Data Applications
13B	1101 1	Open Data Applications
14A	1110 0	Enhanced Other Networks information
14B	1110 1	Enhanced Other Networks information
15A	1111 0	Long Programme Service name
15B	1111 1	Fast switching information

Figura 5: Tipi di gruppo RDS [6]



## 1.2.4 Descrizione dei servizi

Segue una descrizione dei principali servizi RDS offerti. [5, 6]

- Alternative Frequencies (**AF**):

L'elenco delle frequenze alternative fornisce una lista di frequenze che trasmettono lo stesso programma nella stessa area di ricezione o in aree adiacenti. Consente ai ricevitori dotati di memoria di memorizzare l'elenco, per ridurre il tempo di passaggio alla frequenza alternativa qualora fosse necessario, solitamente per la ricerca di una migliore ricezione.

- Clock Time (**CT**):

Fornisce al ricevitore l'ora e la data esatta, secondo il riferimento UTC<sup>5</sup>. Queste informazioni non sono direttamente utilizzate dall'ascoltatore ma servono al circuito del ricevitore per un'eventuale sincronizzazione.

- Extended Country Code (**ECC**):

Codice di 8 bit che identifica il Paese da cui l'emittente sta trasmettendo.

- Emergency Warning System (**EWS**):

È un servizio che abilita l'invio di messaggi di pericolo, viene utilizzato solo in caso di emergenza ed è indirizzato a speciali ricevitori.

- Open Data Applications (**ODA**):

Le applicazioni *open data* implementano servizi aggiuntivi rispetto a quelli definiti nello standard RDS. Permettono a terze parti di programmare ed inviare contenuti aggiuntivi di qualsiasi tipologia all'interno del messaggio RDS. Una delle ODA più utilizzate ad oggi è il *RadioText+*, discusso nel seguito dell'elenco.

- Programme Identification (**PI**):

Codice di 16 bit che permette l'identificazione univoca dell'emittente e del programma specifico.

---

<sup>5</sup> Universal Time Coordinated

- Programme Service (**PS**):

Fornisce gli 8 caratteri alfanumerici che indicano il nome dell'emittente radiofonica, una delle informazioni principali di questa tecnologia. Può essere integrato dal *Long PS* (LPS) se il nome della radio eccede gli 8 caratteri.

- Programme Type (**PTY**):

È un numero identificativo che specifica il tipo di programma, per fornire all'ascoltatore la tipologia di radio che sta ascoltando. Tra i più utilizzati in Italia vi sono: News, Sport, Varie, Musica Pop, Finanza, Religione.

- RadioText e RadioText+ (**RT/RT+**):

Sono entrambi utilizzati per trasmettere testo aggiuntivo, che verrà mostrato sul display del ricevitore. Tra le informazioni trasmesse più comuni vi sono il nome del brano e dell'artista in ascolto, oppure il numero di telefono della radio, per chiamate o messaggi dagli ascoltatori. Mentre il RadioText permette esclusivamente l'invio di testo alfanumerico fino a 64 caratteri, il RadioText+ consente anche la personalizzazione della posizione, la creazione di liste o sottosezioni (ad esempio separa il nome della canzone dal nome dell'artista), e varie altre modifiche più o meno visibili dall'utente finale.

- Traffic Announcement (**TA**):

Si tratta di un segnale che indica quando un annuncio sul traffico viene trasmesso. Può essere utilizzato nei ricevitori per commutare automaticamente da qualsiasi stazione all'annuncio sul traffico. Solitamente la radio portatili o le autoradio abilitate a questo servizio hanno un tasto o una impostazione per decidere se mettersi in ascolto dell'annuncio sul traffico, qualora venga trasmesso.

- Traffic Programme (**TP**):

È un flag (1 bit) che indica se il programma in ascolto sta trasmettendo informazioni sul traffico. Il flag è settato a 1 solo quando il ricevitore passa dinamicamente ad un annuncio sul traffico tramite il segnale TA.

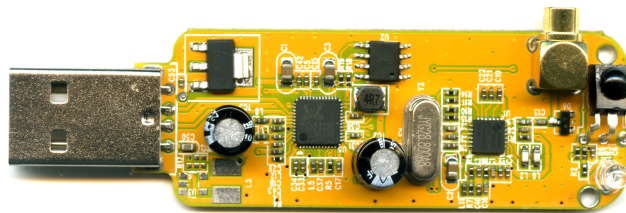
## Capitolo 2

# Software Defined Radio e MATLAB

I componenti di un sistema radio come modulatori, demodulatori e sintonizzatori sono tradizionalmente implementati in hardware, analogicamente. L'avvento dell'elettronica moderna e dei convertitori analogico-digitali ha consentito di implementare via software la maggior parte di questi componenti. Da qui il termine *Software Defined Radio*, cioè radio definita via software. Ciò consente un'elaborazione digitale del segnale semplificata, ed un notevole abbassamento dei prezzi, per tecnologie che prima erano accessibili esclusivamente agli addetti ai lavori o alla ricerca. [7]

## 2.1 RTL-SDR

Tra le SDR disponibili sul mercato, la tecnologia che si distingue per accessibilità e prezzo è sicuramente la RTL-SDR. Si tratta di un dongle USB molto economico, che, collegato al computer su cui girano appositi applicativi<sup>6</sup>, può essere utilizzato per ricevere segnali radio in tempo reale. Il termine *RTL* non è un acronimo, si riferisce al chip Realtek RTL2832U su cui è basato il dispositivo. A seconda del modello, può ricevere frequenze da 500 kHz a 1.75 GHz ma, a differenza di altre tecnologie SDR, non è in grado di trasmettere segnali. Questo ampio spettro di frequenze ricevibili permette lo studio della radio commerciale, modulata sia AM che FM, ma anche la ricezione di sistemi GSM, LTE, GPS e altro ancora.



RTL-SDR: interno del dongle USB

---

<sup>6</sup> Nel caso di questa tesi si utilizza la piattaforma di programmazione MATLAB.

## 2.1.1 Architettura RTL-SDR

Il ricevitore RTL-SDR è di tipo eterodina *low-IF*<sup>7</sup>, cioè demodula il segnale in ingresso in due passaggi: una prima demodulazione analogica, dal segnale a radiofrequenza (RF) ad una frequenza intermedia (IF) fissata a 3.57 MHz; seguita da una seconda demodulazione, questa volta digitale, da IF in banda base. È riportato in Figura 6 lo schema a blocchi dell'architettura interna. [9]

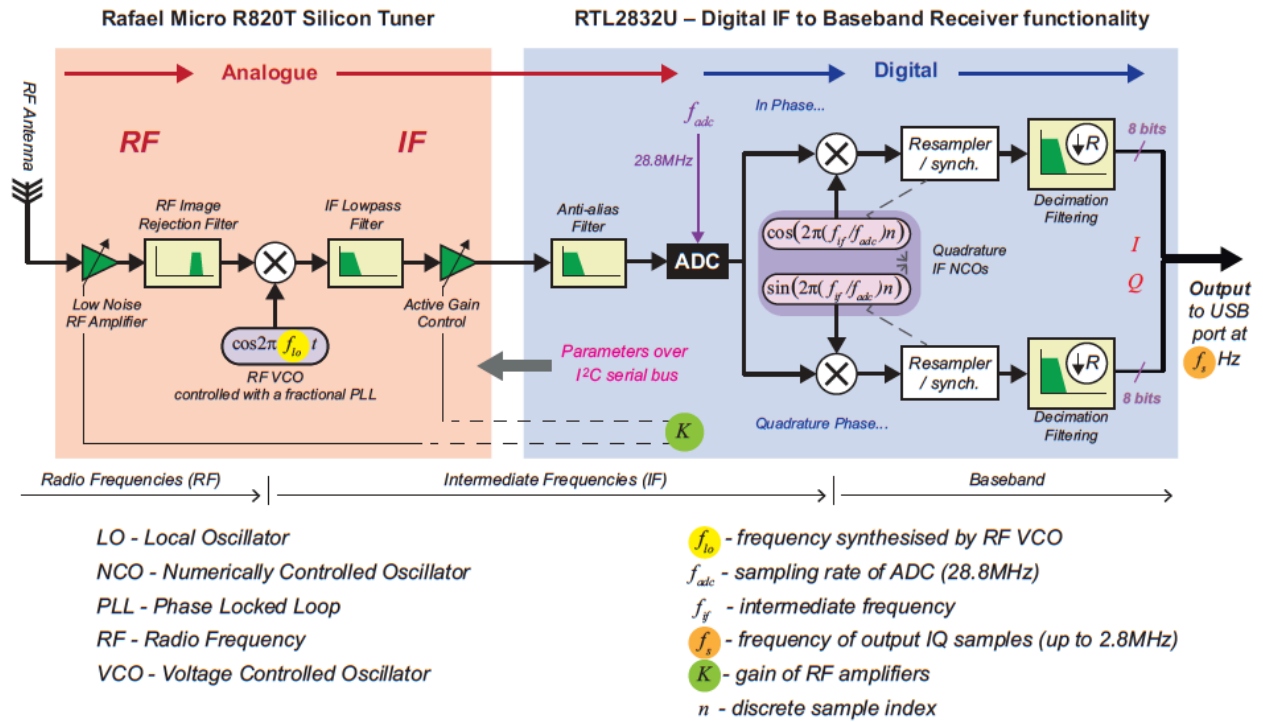


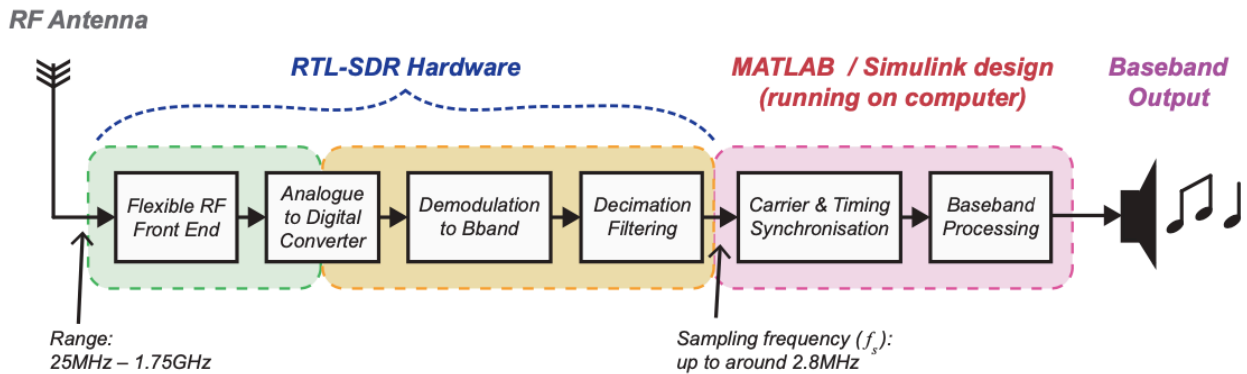
Figura 6: Schema a blocchi dell'architettura RTL-SDR

Dallo schema si vede chiaramente che, una volta traslato il segnale a IF, il convertitore analogico-digitale campiona il segnale alla frequenza di campionamento  $f_{adc} = 28.8\text{MHz}$ , per poi fare una demodulazione digitale in quadratura. I campioni I/Q così ottenuti subiscono un processo di decimazione (*downsampling*) per ridurre la frequenza di campionamento che, nel caso del progetto di questa tesi, viene portata a 228 kHz (vedi sezione 3.2.1). A questo punto i campioni I/Q sono pronti per essere inviati al computer tramite l'interfaccia USB, per poi essere processati via software.

<sup>7</sup> Ricevitore eterodina con frequenza intermedia medio-bassa, tipicamente qualche MHz.

## 2.2 MATLAB

Tra i programmi per l'elaborazione numerica del segnale in uscita dalla RTL-SDR, si è scelto di utilizzare l'ambiente di sviluppo MATLAB, perché permette di scrivere codice che può modificare ogni dettaglio della ricezione tramite Software Defined Radio. MathWorks ha rilasciato il primo pacchetto di supporto<sup>8</sup> per RTL-SDR nel 2014, e permette a MATLAB di interfacciarsi e controllare il dispositivo USB. Grazie a questo pacchetto, i campioni in banda base in uscita dalla RTL-SDR sono inviati all'ambiente di programmazione, permettendo all'utente di implementare qualsiasi tipo di elaborazione digitale del segnale tramite codice. [8]



Schema a blocchi della catena di ricezione MATLAB-SDR

MATLAB offre una piattaforma dove generare codice e costruire il ricevitore, e con l'aggiunta dei *toolbox*<sup>9</sup> mette a disposizione i mezzi necessari per implementare qualsiasi algoritmo di ricezione. È possibile costruire filtri digitali e visualizzare i segnali sia nel dominio del tempo che nel dominio della frequenza, in tempo reale mentre subiscono il processo di demodulazione. [9]

---

<sup>8</sup> Communications Toolbox Support Package for RTL-SDR Radio [10].

<sup>9</sup> DSP System Toolbox; Communications System Toolbox; Signal Processing Toolbox.

## Capitolo 3

# Implementazione del Ricevitore RDS

RDS in Europa ed RBDS in Nord America [11] sono due standard praticamente identici, che si realizzano in 3 livelli:

- Livello Fisico (Layer 1)
- Livello Data Link (Layer 2)
- Livello di Sessione e Presentazione (Layer 3)

## 3.1 Codice MATLAB

Questo codice implementa un ricevitore FM che estrae le informazioni inviate dalla sottoportante digitale RDS. Funziona sia con segnali catturati (*offline processing*) sia con segnali ricevuti real-time tramite RTL-SDR. [13]

```
% ----- INIZIO DEL CODICE ----- %

userInput = helperRBDSInit();
userInput.Duration = 60;
% userInput.SignalSource = 'File';
% userInput.SignalFilename = 'rbds_capture.bb';
userInput.SignalSource = 'RTL-SDR';
userInput.CenterFrequency = 102.5e6;

[rbdsParam, sigSrc] = helperRBDSConfig(userInput);

% Crea e configura un oggetto di sistema FM Broadcast Receiver con i
parametri RDS
fmBroadcastDemod = comm.FMBroadcastDemodulator(...
    'SampleRate',228e3, ...
    'FrequencyDeviation',rbdsParam.FrequencyDeviation, ...
    'FilterTimeConstant',rbdsParam.FilterTimeConstant, ...
    'AudioSampleRate',rbdsParam.AudioSampleRate, ...
    'Stereo',true);

% Creazione del player audio
player = audioDeviceWriter('SampleRate',rbdsParam.AudioSampleRate);

% Funzione del layer 2
datalinkDecoder = RBDSDataLinkDecoder();

% Funzione del layer 3
sessionDecoder = RBDSSESSIONDecoder();
```

```

% Registrazione del codice ODA relativo al RadioText+ (RT+):
rtID = '4BD7';
registerODA(sessionDecoder, rtID, @RadioTextPlusMainGroup, ...
@RadioTextPlus3A);

% Inizializzazione del visualizzatore che servirà per mostrare i dati RDS
viewer = helperRBDSViewer();

start(viewer)
radioTime = 0;

% MAIN LOOP
while radioTime < rbdsParam.Duration
    % Ricezione dei campioni dalla RTL-SDR
    rcv = sigSrc();

    % Demodulazione del segnale FM e riproduzione del segnale audio
    audioSig = fmBroadcastDemod(rcv);
    player(audioSig);

    % Elaborazione del livello fisico
    bitsPHY = RBDSPhyDecoder(rcv, rbdsParam, 1);

    % Elaborazione del livello data-link
    [enabled,iw1,iw2,iw3,iw4] = datalinkDecoder(bitsPHY);

    % Elaborazione del livello sessione e presentazione
    outStruct = sessionDecoder(enabled,iw1,iw2,iw3,iw4);

    % Visualizzazione del contenuto del messaggio RDS
    update(viewer, outStruct);

    % Aggiorna radioTime per proseguire il ciclo
    radioTime = radioTime + rbdsParam.FrameDuration;
end

% Rilascio dei moduli oggetto utilizzati
stop(viewer);
release(sigSrc);
release(player);

% _____ FINE _____ %

```

*Copyright 2016-2022 The MathWorks, Inc.*

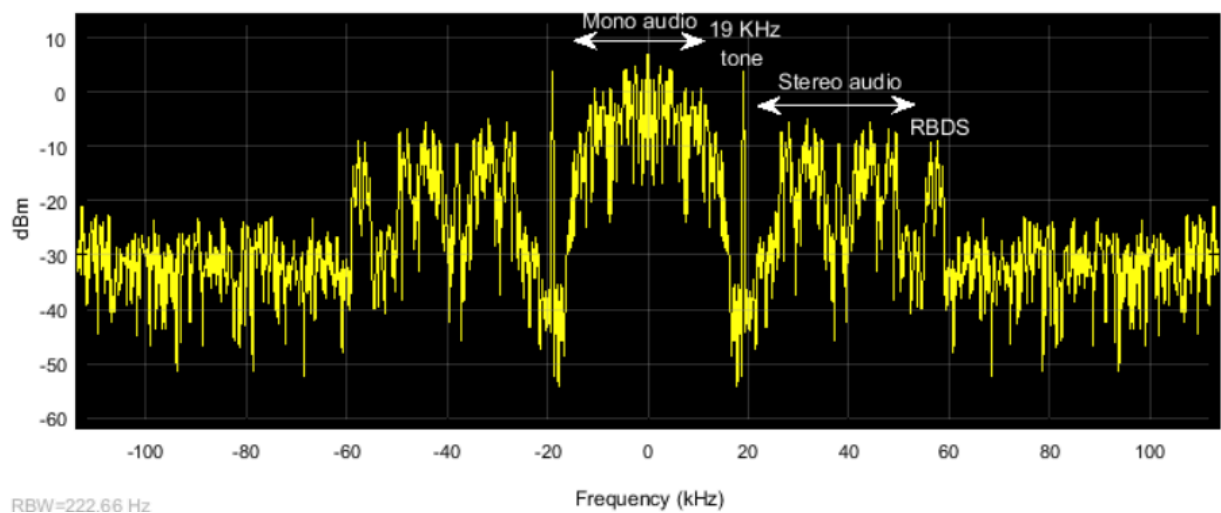
## 3.2 Funzionamento del codice

### 3.2.1 Livello Fisico

La funzione `RBDSPhyDecoder` riceve il segnale catturato dalla RTL-SDR ed esegue i seguenti passaggi:

- Demodulazione FM:

Una volta che il segnale FM è in banda base, il segnale RDS è centrato a 57 kHz, con una banda di  $\pm 2.4$  kHz. Il segnale è trasmesso con una potenza non elevata, di conseguenza è possibile che non sia presente in alcune simulazioni.



Segnale multiplex FM

Il segnale FM contiene il tono pilota a 19 kHz, che è usato come riferimento di fase e frequenza per la demodulazione coerente del segnale RDS a 57 kHz, e del segnale audio L-R a 38 kHz. I toni pilota a 38 kHz e 57 kHz sono generati duplicando e triplicando la frequenza del tono a 19 kHz.

- Filtraggio passa banda:

Viene eseguito un filtraggio a 19 kHz e 57 kHz, per isolare il tono pilota ed il segnale RDS, rispettivamente.

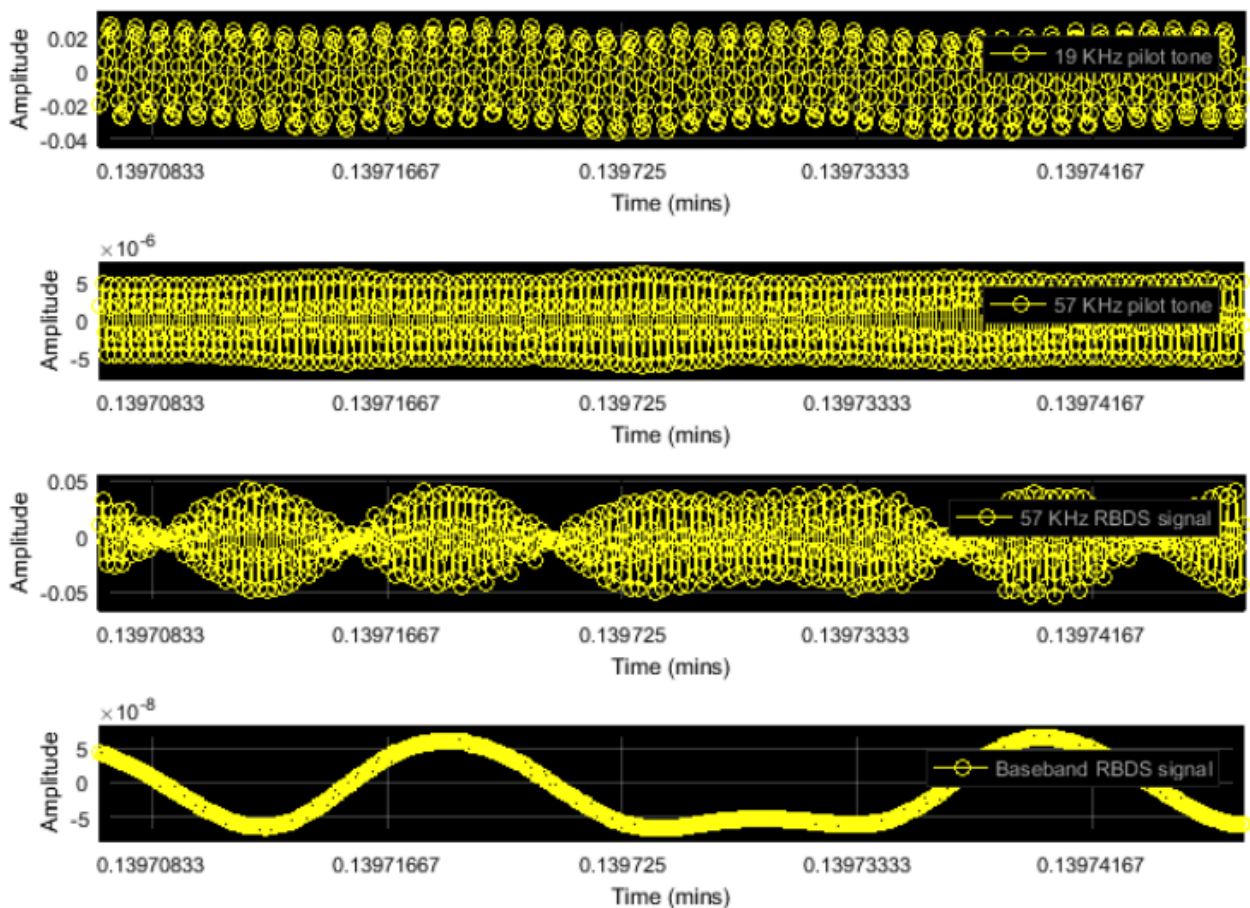


- Triplicazione di frequenza:

Il segnale analitico del tono a 19 kHz viene elevato al cubo, per triplicare la sua frequenza ed ottenere il tono pilota a 57 kHz.

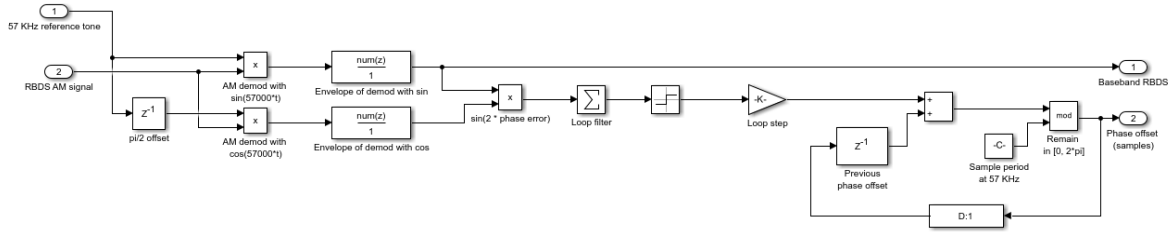
- Demodulazione AM (DSB):

I simboli RDS sono generati con un rate di 1187.5 bit/s e sono modulati DSB-SC a 57 kHz. Il segnale RDS a 57 kHz può essere demodulato coerentemente con una portante a 57 kHz fissata in fase e frequenza. Tipicamente è sufficiente il tono a 19 kHz triplicato per ottenere un buon risultato. I seguenti grafici mostrano il tono pilota a 19 kHz, quello a 57 kHz, il segnale RDS modulato a 57 kHz ed il segnale RDS demodulato in banda base.



È possibile che il segnale RDS abbia un offset di fase variabile nel tempo rispetto al tono pilota e alla sua versione triplicata. A tal proposito il ricevitore contiene un loop di Costas [12] per compensare questi offset di

fase. Segue lo schema a blocchi del loop di Costas implementato in MATLAB.



Loop di Costas implementato in [RBDSCostasLoop](#)

- Loop di Costas:

Sono implementate due demodulazioni AM ortogonali, una con un seno a 57 kHz e l'altra con un coseno a 57 kHz. La frequenza di campionamento del segnale ricevuto è pari a 228 kHz, il che significa che ci sono 4 campioni per ogni ciclo a 57 kHz ( $\frac{228}{57} = 4$ ). Perciò, un ritardo di un campione nel tono pilota a 57 kHz comporta un offset di fase di un quarto di lunghezza d'onda, e consente di generare una senoide da una cosinusoide. Il segnale demodulato con il coseno corrisponde all'output della demodulazione coerente, mentre quello demodulato con il seno viene utilizzato per la rilevazione dell'errore di fase. I prodotti tra il segnale RDS a 57 kHz e le onde seno/coseno vengono filtrati con un filtro passa basso; il prodotto dei due output del filtro è un segnale di errore. Maggiore è il segnale di errore, tanto più si ritarda il tono pilota a 19 kHz per adeguarsi all'offset di fase.

- Estrazione del Clock:

Per procedere con la decodifica dei simboli bifase, il clock RDS pari a 1187.5 Hz viene estratto dal tono pilota a 19 kHz. Si noti che  $1187.5 \text{ Hz} \cdot 16 = 19 \text{ kHz}$ . Per tener conto degli offset di fase, viene utilizzato un divisore di frequenza, [RBDSFrequencyDivider](#), per estrarre il clock dal tono pilota a 19kHz. Poiché l'operazione di divisione di frequenza fornisce più risposte corrette, il segnale RDS in banda base è usato come riferimento (dato di training) nella determinazione dell'output corretto.

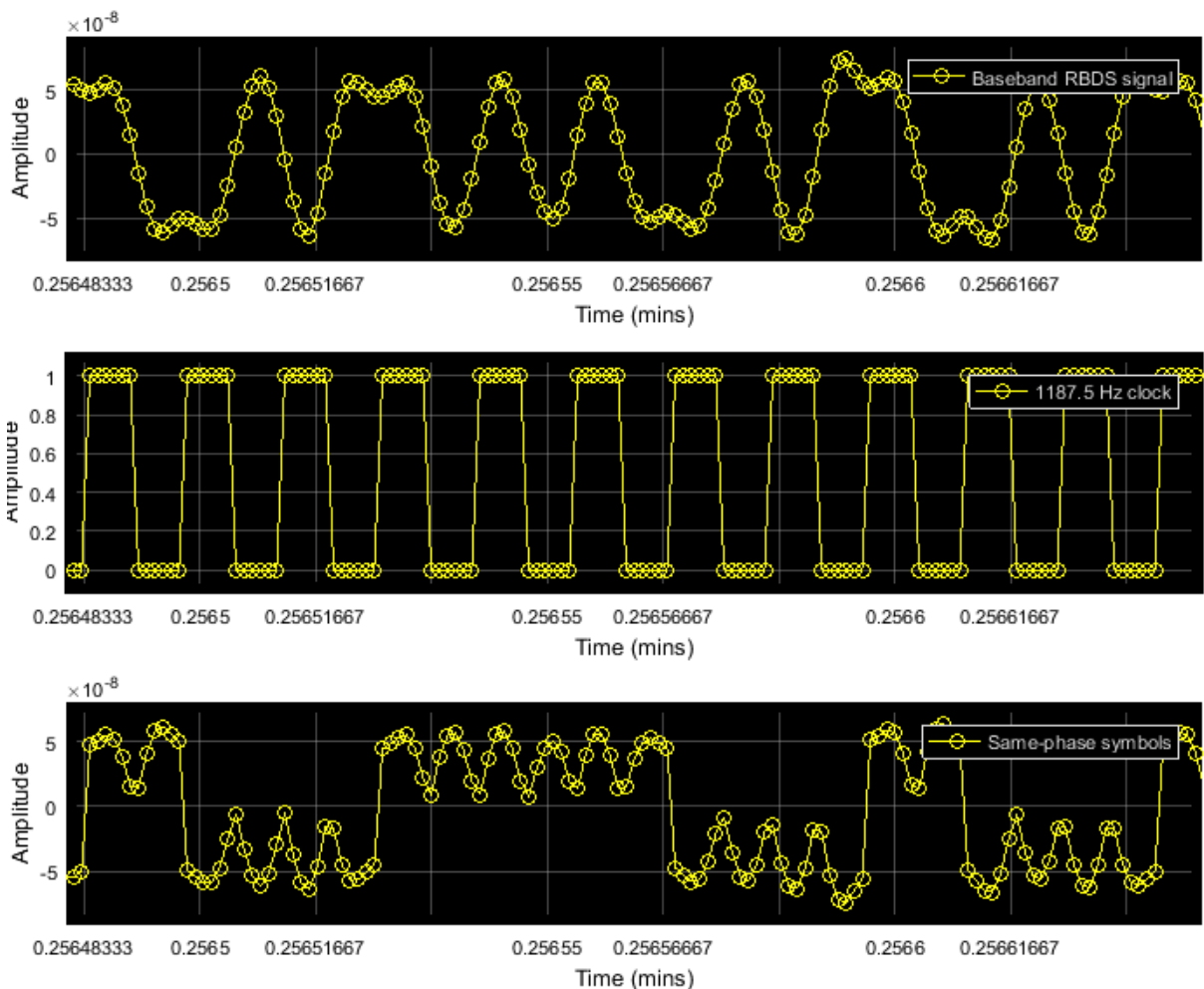
- Decodifica dei simboli:

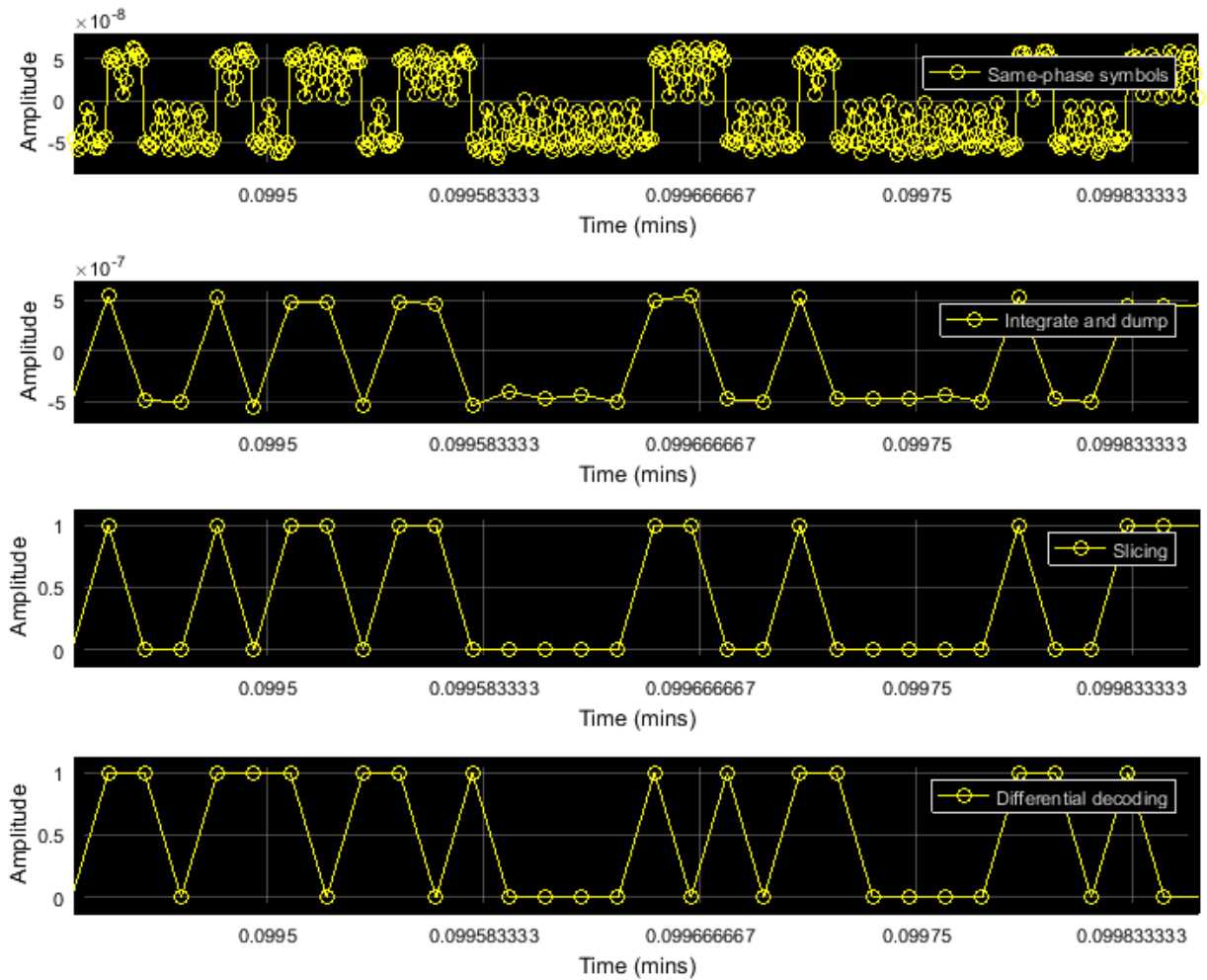
RDS utilizza una codifica bifase, anche nota come codifica Manchester. Ad ogni ciclo di clock, il simbolo RDS assume due valori di ampiezza opposti, positivo seguito da negativo, o negativo seguito da positivo. Il decodificatore nega il secondo livello di ampiezza, in modo che ogni simbolo mantenga lo stesso livello di ampiezza per l'intero ciclo di clock. Il nuovo livello di ampiezza corrisponde alla rappresentazione in bit del simbolo. Per ottenere il valore dei bit di ciascun simbolo, la forma d'onda viene integrata su ogni ciclo di clock (*integrate & dump*), e l'esito viene confrontato con zero ( *slicer*).

- Decodifica differenziale:

Infine, i bit vengono decodificati differenzialmente per ripristinare la codifica differenziale al trasmettitore.

I seguenti grafici corrispondono a quelli in Figura 2 nella sezione 1.2.1.





### 3.2.2 Livello Data Link

Questo livello viene eseguito dalla funzione [RBDSDDataLinkDecoder](#), che è responsabile della sincronizzazione e della correzione degli errori. I bit in uscita dal livello fisico sono organizzati in gruppi da 104 bit, ogni gruppo comprende quattro blocchi da 26 bit l'uno. Ogni blocco contiene 16 bit informativi e 10 bit di controllo. All'interno del controllo è inclusa la *offset word*, che serve ad identificare i blocchi all'interno di ciascun gruppo:

- Sincronizzazione:

Inizialmente, i gruppi sono identificati usando una finestra a scorrimento di 104 bit. Per ogni finestra si cercano le offset word negli ultimi 10 bit di ogni blocco da 26 bit. Una offset word è correttamente identificata se la funzione [RBDSErrorDetection](#) non riporta errori sui bit. Una volta avvenuta l'identificazione, il sistema è sincronizzato a livello di gruppo ed il

processo di finestrata si interrompe. Da qui in avanti, i prossimi 104 bit saranno trattati come il gruppo successivo.

Nel caso in cui i gruppi futuri contenessero degli errori, e le offset word non venissero identificate, la sincronizzazione potrebbe essere persa. In questi casi, questo Livello esamina dapprima la possibilità di un *bit slip* (slittamento di sincronizzazione di 1 solo bit), considerando che i primi 16 bit informativi sono sempre gli stessi per ogni gruppo (codice PI). Se questi vengono rilevati in ritardo o in anticipo di 1 bit, viene mantenuta la sincronizzazione e i limiti di ogni gruppo sono accordati di conseguenza. Se gli errori sui bit persistono per 25 gruppi consecutivi e non è possibile individuare un singolo bit slip, la sincronizzazione fallisce e il sistema riparte con l'identificazione tramite finestra a scorrimento.

- Correzione degli errori:

Il codice a correzione dell'errore nello standard RDS è un codice ciclico<sup>10</sup> (26,16). La funzione `RBDSErrorCorrection` implementa il registro descritto nell'appendice B dello standard RDS [5].

### 3.2.3 Livello di Sessione e Presentazione

Prima di passare al livello superiore, il Layer 2 rimuove i 10 bit di controllo da ogni blocco. In questo modo, il Layer 3 riceve gruppi di 64 bit, composti da quattro blocchi con 16 bit ciascuno. Lo standard RDS definisce 32 tipi di gruppo, ognuno etichettato con un numero da 0 a 15 ed una versione, A o B (si faccia riferimento alla figura 5 trattata nella sezione 1.2.3). La funzione `RBDSSessionDecoder` procede con la decodifica dei gruppi che hanno un formato fisso, cioè: 0A, 0B, 2A, 2B, 3A, 4A, 10A.

- I gruppi 0A e 0B inviano tipicamente una stringa di 8 caratteri (PS), che corrisponde al nome dell'emittente radiofonica.

---

<sup>10</sup> Per la teoria a riguardo si veda [https://en.wikipedia.org/wiki/Cyclic\\_code](https://en.wikipedia.org/wiki/Cyclic_code)

- Le implementazioni ODA possono essere aggiunte a questo codice utilizzando la funzione `registerODA`, contenuta all'interno della funzione `RBDSSessionDecoder`. Per farlo, si passa l'identificatore esadecimale dell'ODA che si vuole aggiungere (gli ID ODA sono elencati in [14]). Nel caso del `RadioText+`, si utilizza codice `'4BD7'`.

La finestra in figura contiene i dati RDS raccolti dalla simulazione.

Visualizzatore dati RDS

- Basic RDS/RBDS information:
  - Il primo campo contiene il Programme Type (PTY), contenuto nel secondo blocco di ogni gruppo. Se si ricevono gruppi di tipo 10A si ottiene una caratterizzazione aggiuntiva, ad esempio: Sport \ Calcio.
  - Il secondo campo contiene gli 8 caratteri dei gruppi 0A/0B, il nome dell'emittente (PS).
  - Il terzo campo contiene i caratteri del RadioText, quando si ricevono gruppi di tipo 2A/2B
- RadioText+ (RT+):
 

Questa sezione è caratterizzata dal contenuto dei gruppi di tipo 3A. Quando si ricevono questi dati, i caratteri RadioText sono suddivisi in due sottostringhe, che vengono aggiornate per indicare il messaggio trasportato (solitamente Artista e Titolo della canzone).
- Group type receptions:
 

Queste tabelle mostrano quali e quanti gruppi sono stati ricevuti durante la simulazione.
- Open data applications (ODA):
 

Se si ricevono gruppi di tipo 3A, questa lista mostra il nome ed il gruppo dedicato alla specifica ODA ricevuta.

Viene inoltre mostrato lo spettro del segnale multiplex FM in tempo reale.

### 3.2.5 Simulazione

Di seguito sono mostrati i risultati di una simulazione eseguita nel Maggio 2024. Il codice è eseguito su un portatile HP con sistema operativo Windows 11, su cui gira MATLAB R2022b. L'hardware RTL-SDR utilizzato è un dongle USB Andoer Mini<sup>11</sup>, munito di una antenna omnidirezionale marca Proxel<sup>12</sup> modello SRH789.

---

<sup>11</sup> <https://amzn.eu/d/0XmK76o>

<sup>12</sup> <https://www.proxelsrl.com/antenne>

1. Prima prova:

```
userInput.SignalSource = 'RTL-SDR';  
userInput.CenterFrequency = 100.5e6;
```

RDS / RBDS and RadioText Plus (RT+) Viewer

Basic RDS / RBDS information

Program type: Top 40

Program service name: DEEJAY

Radiotext: DeeJay Football Club - Fabio Caressa e Ivan Zazzaroni

RadioText Plus (RT+)

Artist: Fabio Caressa e Ivan Zazzaroni

Title: DeeJay Football Club

Group type receptions

	Receptions		Receptions		Receptions		Receptions
0A	374	8A	0	0B	0	8B	0
1A	0	9A	0	1B	0	9B	0
2A	191	10A	0	2B	0	10B	0
3A	19	11A	16	3B	0	11B	0
4A	0	12A	0	4B	0	12B	0
5A	0	13A	0	5B	0	13B	0
6A	0	14A	0	6B	0	14B	0
7A	0	15A	0	7B	0	15B	0

Open data applications (ODA)

	Name	Group Type
1	RadioText+ / RT+	11A

Figura 7

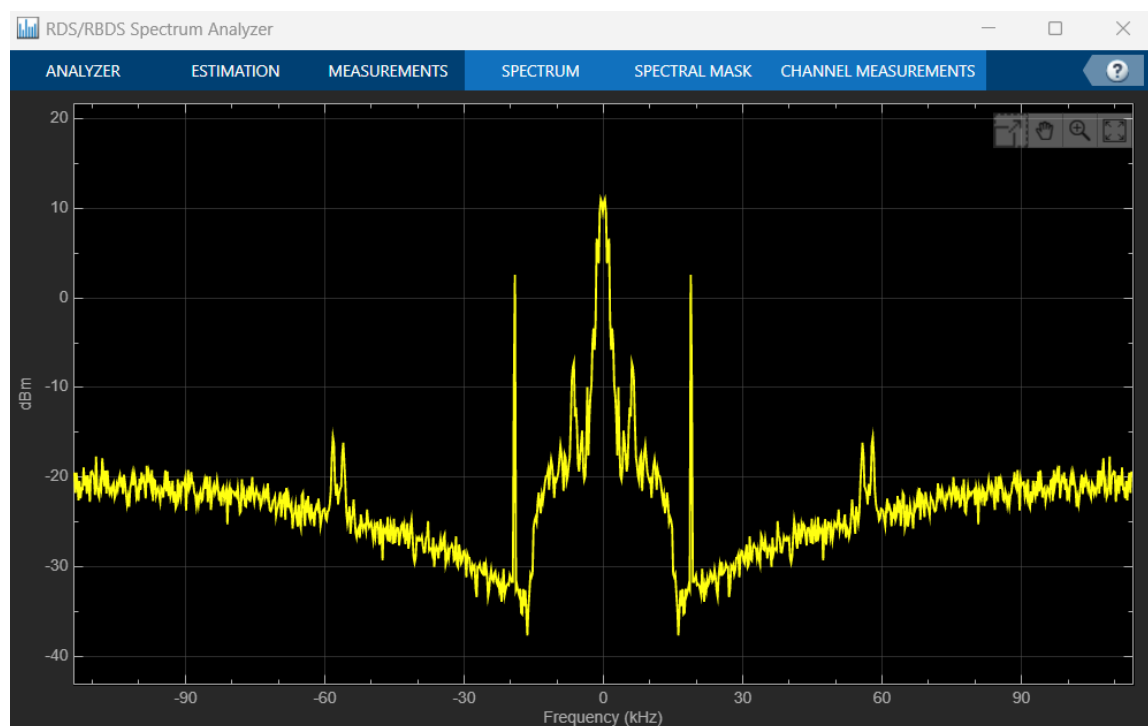


Figura 8



## 2. Seconda prova:

```
userInput.SignalSource = 'RTL-SDR';
userInput.CenterFrequency = 102.1e6;
```

RDS / RBDS and RadioText Plus (RT+) Viewer

Basic RDS / RBDS information

Program type: Country

Program service name: R.BRUNO

Radiotext: 24KGOLDN - MOOD (FEAT. IANN DIOR)

RadioText Plus (RT+)

Artist: 24KGOLDN

Title: MOOD (FEAT. IANN DIOR)

Group type receptions

	Receptions		Receptions		Receptions		Receptions
0A	386	8A	0	0B	0	8B	0
1A	0	9A	0	1B	0	9B	0
2A	226	10A	0	2B	0	10B	0
3A	6	11A	0	3B	0	11B	0
4A	2	12A	29	4B	0	12B	0
5A	0	13A	0	5B	0	13B	0
6A	0	14A	0	6B	0	14B	0
7A	0	15A	0	7B	0	15B	0

Open data applications (ODA)

	Name	Group Type
1	RadioText+ / RT+	12A

Figura 9

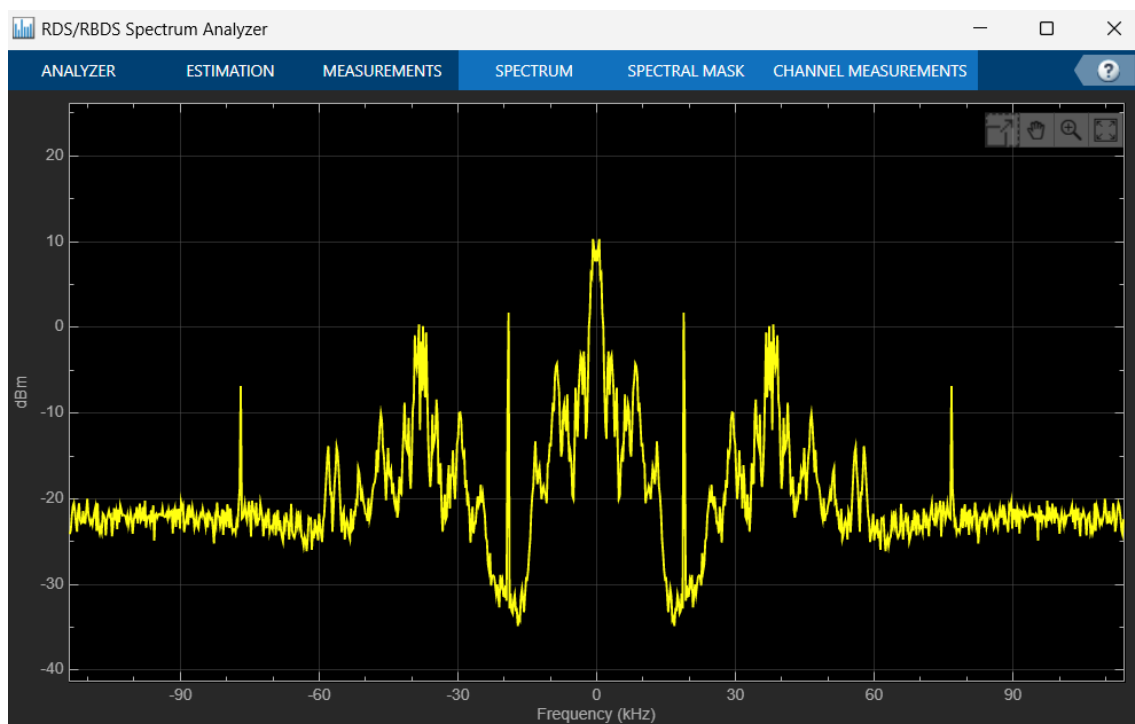


Figura 10

Per quanto riguarda la prima prova, è stata analizzata la frequenza 100.5 MHz che, come visibile in Figura 7 nel campo PS, corrisponde all'emittente Radio DeeJay. L'indicatore PTY riporta la dicitura *Top 40*, che è l'equivalente americano del nostro *Vari*. Il campo RadioText ci informa che è in onda un talk show calcistico; si può notare che l'emittente supporta la tecnologia RadioText+, infatti invia gruppi di tipo 3A, separando graficamente i nomi dei telecronisti dal titolo del programma.

Nella seconda prova è stata analizzata la frequenza 102.1 MHz che, come visibile in Figura 9 nel campo PS, corrisponde all'emittente Radio Bruno. L'indicatore PTY riporta la dicitura *Country*, che è l'equivalente americano del nostro *Musica Pop*. Il campo RadioText indica l'artista ed il titolo della canzone in ascolto; anche questa emittente supporta la tecnologia RadioText+, che in questo caso separa graficamente il nome dell'artista dal titolo del brano. Si può notare che sono stati ricevuti anche dei gruppi di tipo 4A, responsabili dell'invio del giorno e dell'ora esatta, non esplicitamente visibili nel display.

Confrontando la Figura 8 con la Figura 10, cioè lo spettro dei segnali FM ricevuti, è interessante sottolineare una differenza ben visibile. Il segnale FM relativo a Radio DeeJay non presenta la sottoportante a 38 kHz, responsabile della trasmissione stereo. Se invece facciamo riferimento al segnale FM di Radio Bruno, ecco che la sottoportante stereo L-R è presente. Questo è dovuto al fatto che Radio DeeJay sta trasmettendo un talk, mentre Radio Bruno sta trasmettendo una canzone. Infatti, quando le emittenti trattano programmi di sola voce, trasmettono solitamente in mono, perché la voce non presenta differenze significative tra canale destro e sinistro, e questo consente di risparmiare potenza in trasmissione. Al contrario, la musica è diffusa in stereo, per garantire un ascolto qualitativamente superiore.

# Conclusioni

L'utilizzo combinato delle Software Defined Radio e MATLAB è un ottimo metodo per comprendere a fondo il funzionamento delle radio commerciali. In questa tesi mi ha permesso di capire e di gestire il processo di ricezione del segnale RDS, rendendo ogni passaggio esaminabile e, all'occorrenza, modificabile. Questa versatilità si presta sia alla didattica che alla ricerca nel campo delle radiocomunicazioni, visto l'ottimo rapporto tra prezzo dei componenti e varietà di servizi implementabili.

Come evidenziato in precedenza in questo studio, la radio FM vivrà ancora per molti anni. Nonostante la tecnologia RDS sia inevitabilmente legata alla vita della radio FM, i progetti e la ricerca a riguardo non si sono fermati, anche grazie all'ottimo lavoro svolto dal RDS Forum. A tal proposito, è stato completato lo sviluppo della nuova versione del Radio Data System, *RDS2*. Questa tecnologia, retrocompatibile con i ricevitori FM/RDS, risolve il problema principale che affligge il RDS, cioè la quantità limitata di dati trasferibili, nettamente inferiore rispetto al DAB+. Con l'aggiunta di tre sottoportanti (posizionate a 66.5 kHz, 71.25 kHz e 76 kHz), si riesce a quadruplicare la quantità di dati trasmessi. Queste sottoportanti aggiuntive utilizzano la stessa struttura di gruppo del RDS 'base', che resta invariato a 57 kHz, ed hanno il vantaggio di non dover trasportare le informazioni principali quali PI, PS, PTY, il che le rende totalmente disponibili al trasporto di ODA (vedi sezione 1.2.4).

I servizi aggiuntivi introdotti dal RDS2 includono: informazioni più dettagliate sul traffico urbano, loghi delle stazioni radio mostrate sul display, *slideshow* con testo, immagini e riferimenti a Internet, avvisi di emergenza potenziati. Il servizio può anche essere utilizzato per aggiornare l'autoradio a distanza, un'ottima proposta vista la rapida evoluzione delle applicazioni.

Tuttavia, RDS2 non è ancora sul mercato, perché i costi per la sua implementazione si mantengono molto bassi solo se prodotto in grandi quantità,

il che richiede una *killer feature*<sup>13</sup>, non ancora identificata con chiarezza. Questo è riconducibile anche ai pregiudizi di molti addetti ai lavori, che ritengono il RDS, ed in generale la radio FM, una tecnologia superata dalle radio digitali come DAB+. Spesso però manca l'onestà di ammettere che anche il DAB è una tecnologia ormai 'vecchia', che lotta da anni per ottenere il primato del broadcasting non riuscendoci fino in fondo. Infatti, le principali emittenti hanno già evidenziato che il futuro della radio tende verso i servizi di streaming e la telefonia mobile 5G, per una progressiva sostituzione del DAB. [15]

L'idea che mi sono fatto riguardo il futuro della radio, nonostante la mia modesta esperienza, è una soluzione eterogenea delle tecnologie attualmente presenti. Ritengo che non si arriverà ad una tecnologia unica che comporti lo spegnimento delle altre, per un fattore economico, per gli accordi tra le società che regolano la radiodiffusione, e anche per un fattore sociale. Pertanto la radio FM/RDS, la radio DAB+, la radio in streaming via Internet e le nuove tecnologie coesisteranno tra loro. La vera sfida non sarà la ricerca della tecnologia trasmissiva migliore, bensì il rinnovamento dei palinsesti radiofonici, per contrastare il progressivo disinteresse del pubblico verso le trasmissioni radio, a favore di altri servizi multimediali come lo streaming video.

---

<sup>13</sup> Caratteristica talmente necessaria da aggiudicarsi la produzione su larga scala.

# Ringraziamenti

Al mio relatore, Professor Marco Moretti, per avermi fatto appassionare alla materia e per il tempo che ha investito in questa tesi.

Alla mia famiglia, per l'amore e le possibilità che mi ha dato.

Ai miei amici, senza di loro sarei perso.

Grazie.

# Bibliografia e Sitografia

- [1] Fonzo, E. (20 Aprile 2023). Storia della radio dalle origini a oggi. Geopop. <https://www.geopop.it/storia-dellinvenzione-della-radio-dalle-origini-a-oggi/>
  
- [2] RDS Forum - The Association of RDS Users. <https://www.rds.org.uk/>
  
- [3] Wikimedia Foundation. (1 Maggio 2024). Digital Audio Broadcasting. Wikipedia. [https://en.wikipedia.org/wiki/Digital\\_Audio\\_Broadcasting](https://en.wikipedia.org/wiki/Digital_Audio_Broadcasting)
  
- [4] D. Kopitz, & B. Marks (1999). RDS, the Radio Data System. Artech House.
  
- [5] CENELEC EN 50067:1998 - Specifications of the Radio Data System (RDS) for VHF/FM broadcasting, European Committee for Electrical Standardisation (CENELEC), 35B rue de Stassart, B-1050 Brussels, Switzerland, April 1998.
  
- [6] Updated Compilation of all RDS Forum Final Drafts for the RDS/RDS2 standard IEC 62106, RDS Forum Office – Geneva, Switzerland.
  
- [7] About RTL-SDR, RTL-SDR.COM. (5 Settembre 2023). <https://www.rtl-sdr.com/about-rtl-sdr/>
  
- [8] R. W. Stewart et al., "A low-cost desktop software defined radio design environment using MATLAB, simulink, and the RTL-SDR," in IEEE Communications Magazine, vol. 53, no. 9, pp. 64-71, September 2015, doi: 10.1109/MCOM.2015.7263347.

- [9] R. W. Stewart, K. W. Barlee, D. S. W. Atkinson, L. H. Crockett, & A. G. Broadhurst (2022). Software Defined Radio using MATLAB & Simulink and the RTL-SDR, 2nd Ed. Department of Electronic and Electrical Engineering, University of Strathclyde.
  
- [10] MATLAB RTL-SDR support package. <https://it.mathworks.com/hardware-support/rtl-sdr.html>
  
- [11] EIA/NAB, National Radio Systems Committee: United States RBDS Standard version 2.0, Specification of the Radio Broadcast Data System (RBDS), 1998.
  
- [12] Wikimedia Foundation. (24 Luglio 2023). Costas Loop. Wikipedia. [https://en.wikipedia.org/wiki/Costas\\_loop](https://en.wikipedia.org/wiki/Costas_loop)
  
- [13] RDS/RBDS and RadioText Plus (RT+) FM Receiver. MATLAB & Simulink, MathWorks Italia. <https://it.mathworks.com/help/comm/ug/rds-rbds-and-radiotext-plus-rt-fm-receiver.html>
  
- [14] DSM Archive. DSM Archive | National Radio Systems Committee | Setting Standards for the Future of Radio. <https://www.nrscstandards.org/committees/dsm/archive/archive.asp>
  
- [15] Kopitz, D. (Febbraio 2023). RDS – The Radio Data System: a milestone in the history of broadcasting traffic information. In Historische Grundlagen der mobilen Gesellschaft (pp. 265-276). Nomos Verlagsgesellschaft mbH & Co. KG.