

Modulo SDR

Unità	Argomento
1	Primo flowgraph, sorgenti, sink e moltiplicazione
2	Ricevitore AM da registrazione IQ
3	Segnali audio, FFT, guadagno, filtri ed equalizzatore
4	Traslazione, tuning, filtraggio e decimazione
5	Ricevitore FM a larga banda
6	Banda occupata e visualizzazione di file IQ
7	AM, OOK, PM e FM
8	Nyquist, aliasing, overflow e sintonia automatica
9	Catena FM completa simulata

Significato delle sottocartelle

- **base** : flowgraph di partenza da modificare;
- **dimostrazioni** : flowgraph già completi da eseguire e analizzare;
- **dati** : registrazioni IQ, WAV e archivi sorgente;
- **soluzioni** : risultati con cui confrontare il proprio lavoro;
- **script_generati** : script Python prodotti da GNU Radio Companion.

Hardware SDR

E' possibile utilizzare l'sdr fisico in dotazione. Senza hardware è comunque possibile studiare la catena usando i file IQ registrati. Quando si usa un dispositivo reale:

- iniziare con un guadagno moderato;
- verificare la frequenza centrale e il sample rate supportato;
- evitare sample rate superiori alle capacità del computer;
- controllare i messaggi di overflow;

Nozioni fondamentali

Flowgraph e flusso dei campioni

Un flowgraph GNU Radio è una rete di blocchi collegati. Ogni blocco riceve campioni, li elabora e produce nuovi campioni.

Una catena tipica è:

```
1 Sorgente -> elaborazione -> visualizzazione o uscita
```

Un ricevitore radio più realistico è:

```
1 Sorgente IQ -> sintonia -> filtro -> riduzione del sample rate
2               -> demodulatore -> ricampionamento audio -> altoparlante
```

La correttezza del flowgraph dipende non soltanto dai blocchi scelti, ma anche dal tipo di dato e dal sample rate in ogni tratto.

Tipi di dato: float e complex

Un segnale `float` contiene un solo valore reale per campione. È comune per l'audio e per un messaggio in banda base.

Un segnale `complex` contiene due componenti:

$$x[n] = I[n] + jQ[n]$$

Le componenti I e Q permettono di rappresentare ampiezza e fase e di distinguere frequenze positive e negative rispetto alla frequenza centrale. Le registrazioni SDR sono normalmente complesse.

Regola pratica:

- audio e messaggi: spesso `float` ;
- segnali RF o in banda base complessa: spesso `complex` ;
- una porta arancione e una porta blu, o due tipi differenti, non possono essere collegate senza un blocco di conversione.

Frequenza centrale e frequenza relativa

Una registrazione IQ non memorizza direttamente ogni frequenza RF come numero assoluto. Memorizza una finestra di spettro centrata su una frequenza di sintonia.

Se la frequenza centrale è f_c e la stazione è a f_s , la sua posizione nel file IQ è:

$$f_{rel} = f_s - f_c$$

Esempio del ricevitore AM:

- centro della registrazione: 900 kHz;
- stazione desiderata: 880 kHz;
- posizione relativa: $880 - 900 = -20$ kHz.

La stazione compare quindi a meno 20 kHz rispetto allo zero del grafico.

Mixing e traslazione in frequenza

Moltiplicare un segnale complesso per un oscillatore complesso sposta tutto lo spettro:

$$y[n] = x[n]e^{j2\pi f_{LO}n/f_s}$$

Lo spostamento risultante è f_{LO} . Per portare a zero una componente posta a f_{rel} si usa:

$$f_{LO} = -f_{rel}$$

Nel caso AM la stazione è a meno 20 kHz, quindi un oscillatore a più 20 kHz la porta a zero.

Un **Frequency Xlating FIR Filter** esegue insieme tre operazioni:

1. traslazione in frequenza;
2. filtraggio del canale;
3. decimazione.

È quindi preferibile a una catena separata quando si vuole costruire un ricevitore efficiente.

Sample rate e teorema di Nyquist

Con sample rate f_s , la rappresentazione non ambigua di un segnale reale è limitata a frequenze inferiori a $f_s/2$.

$$f_{max} < \frac{f_s}{2}$$

Per un segnale complesso la finestra visualizzata normalmente va da $-f_s/2$ a $+f_s/2$, con larghezza totale f_s .

Se una componente supera il limite rappresentabile, appare a una frequenza diversa: questo fenomeno è l'aliasing.

Decimazione

La decimazione di fattore M conserva un campione ogni M :

$$f_{s,out} = \frac{f_{s,in}}{M}$$

Prima della decimazione occorre un filtro passa-basso che elimini le componenti non rappresentabili al nuovo sample rate. Il blocco **Keep 1 in N** mostra l'effetto della riduzione, ma da solo non è un decimatore sicuro per un segnale a banda larga perché non applica il filtro anti-alias.

Interpolazione e ricampionamento razionale

Un ricampionatore razionale usa un fattore di interpolazione L e un fattore di decimazione M :

$$f_{s,out} = f_{s,in} \frac{L}{M}$$

Esempio:

$$400000 \frac{32}{400} = 32000 \text{ campioni/s}$$

Questa è la conversione utilizzata nei primi ricevitori AM e FM per raggiungere il sample rate audio di 32 kHz.

Filtri FIR

I parametri essenziali di un filtro sono:

- frequenza o frequenze di taglio;
- larghezza di transizione;
- sample rate;
- guadagno;
- eventuale fattore di decimazione.

Una transizione stretta produce in genere un filtro più selettivo ma richiede più coefficienti e più calcolo. La frequenza di taglio deve essere coerente con il contenuto utile, non soltanto con il sample rate.

Guadagno lineare e decibel

Per un rapporto di ampiezza A :

$$G_{dB} = 20 \log_{10}(A)$$

Esempi:

Guadagno lineare	Guadagno in dB
0,5	circa -6,02 dB
1	0 dB
2	circa +6,02 dB
10	+20 dB
100	+40 dB

Raddoppiare l'ampiezza non significa raddoppiare i dB. Il decibel è una scala logaritmica.

Spettro e FFT

Il [Time Sink](#) mostra l'ampiezza nel tempo. Il [Frequency Sink](#) calcola una FFT e mostra l'energia distribuita in frequenza.

Per interpretare correttamente uno spettro occorre conoscere:

- sample rate del sink;
- dimensione della FFT;
- finestra applicata;
- valore medio e scala in dB;
- natura reale o complessa del segnale.

Un tono puro produce un picco. Un segnale reale mostra normalmente una simmetria tra frequenze positive e negative, mentre un segnale complesso può occupare una sola direzione dello spettro.

Modulazione AM e OOK

Una forma di AM con portante può essere scritta come:

$$s(t) = A_c[1 + \mu m(t)] \cos(2\pi f_c t)$$

Il termine costante **1** mantiene la portante e impedisce, entro certi limiti, che l'involuppo cambi segno. Se l'ampiezza del messaggio è troppo grande, si ha sovrarmodulazione e l'involuppo non rappresenta più correttamente il messaggio.

L'OOK è un caso digitale semplice: la portante è presente per un simbolo e assente per l'altro.

Modulazione FM e sensibilità digitale

Nella FM l'informazione modifica la frequenza istantanea. Il parametro di sensibilità del modulatore digitale dipende dalla deviazione massima Δf e dal sample rate:

$$k = \frac{2\pi \Delta f}{f_s}$$

Nel flowgraph finale dell'unità 8:

- deviazione: 20 kHz;
- sample rate: 360 kHz.

Quindi:

$$k \approx \frac{2\pi \cdot 20000}{360000} \approx 0,349$$

Impostare la sensibilità a **1** senza collegarla al sample rate rende difficile interpretare la deviazione reale.

Banda di una FM

Una stima della banda occupata è data dalla regola di Carson:

$$B \approx 2(\Delta f + f_{m,max})$$

Per una radiodiffusione FM con deviazione elevata, la banda del canale è molto maggiore della sola banda audio. Per questo i flowgraph usano un canale di circa 150 kHz e un sample rate intermedio di 400 kS/s.

Metodo generale per affrontare le attività

Primo passaggio: identificare le grandezze

Prima di aggiungere blocchi, annotare:

- sample rate della sorgente;

- tipo del segnale;
- frequenza centrale;
- frequenza del canale desiderato;
- larghezza del canale;
- sample rate richiesto dal blocco successivo;
- sample rate finale dell'audio.

Secondo passaggio: costruire una tabella dei sample rate

Esempio di ricevitore FM:

Punto della catena	Sample rate
File IQ	8 MHz
Dopo decimazione per 20	400 kHz
Uscita del demodulatore con decimazione audio 1	400 kHz
Dopo ricampionamento 32/400	32 kHz
Audio Sink	32 kHz

Ogni incoerenza in questa tabella indica un errore nel flowgraph.

Terzo passaggio: calcolare la traslazione

Usare sempre una convenzione esplicita:

$$f_{shift} = f_{canale} - f_{centro}$$

Poi verificare il segno osservando lo spettro prima e dopo il mixer. Se il canale si allontana dallo zero, invertire il segno.

Quarto passaggio: filtrare prima di ridurre il rate

Il filtro deve conservare il canale utile e attenuare ciò che produrrebbe aliasing dopo la decimazione.

Quinto passaggio: osservare ogni stadio

Durante lo sviluppo collegare temporaneamente sink a:

- ingresso IQ;
- uscita del mixer;
- uscita del filtro;
- uscita del demodulatore;
- audio finale.

Quando il ricevitore non funziona, questa osservazione progressiva è più efficace della modifica casuale dei parametri.

Unità 1 — Verifica e dimostrazioni introduttive

Tipo di attività: verifica dell'ambiente e dimostrazioni già complete.

Verifica guidata: [verify.grc](#)

Obiettivo

Verificare che GNU Radio e l'uscita audio funzionino generando un tono sinusoidale a 1 kHz.

Catena

```
1 Signal Source a 1 kHz -> Audio Sink a 32 kHz
```

Spiegazione

La sorgente produce 32.000 campioni al secondo. Poiché il tono è a 1.000 Hz, ogni periodo è rappresentato da:

$$\frac{32000}{1000} = 32 \text{ campioni}$$

Il valore è ampiamente superiore al minimo teorico di due campioni per periodo, quindi la forma d'onda è rappresentata correttamente.

Verifiche

- si deve udire un tono stabile;
- il volume della sorgente deve essere moderato;
- il sample rate del [Signal Source](#) deve coincidere con quello dell' [Audio Sink](#).

Errori comuni

- dispositivo audio non disponibile;
- ampiezza troppo elevata e clipping;
- sample rate non supportato dalla scheda audio.

Dimostrazione: [source_sink.grc](#)

Obiettivo

Comprendere il modello sorgente-sink e l'uso del [Throttle](#).

Catena

```
1 Constant Source complessa -> Throttle -> QT GUI Time Sink
```

La sorgente produce il valore complesso costante `3.14`. Il `Throttle` limita il flusso a 32 kS/s. Senza hardware, GNU Radio potrebbe generare campioni alla massima velocità possibile, consumando inutilmente la CPU.

Nel `Time Sink` la parte reale appare costante a 3,14 e la parte immaginaria a zero.

Dimostrazione: `simple_multiply.grc`

Obiettivo

Applicare un guadagno costante a un flusso.

Catena

```
1 3,14 -> moltiplicazione per 3 -> 9,42
```

Il risultato atteso è:

$$3,14 \cdot 3 = 9,42$$

La dimostrazione introduce il principio secondo cui molti blocchi GNU Radio operano campione per campione.

Unità 2 — Laboratorio guidato: ricevitore AM da file IQ

Tipo di attività: laboratorio ricostruito dalla soluzione di riferimento. Non è presente uno starter incompleto dedicato.

File principali

```
1 unita/02_laboratorio_am_guidato/dati/am_broadcast_02_c900k_s400k.iq
2 unita/02_laboratorio_am_guidato/soluzione_di_riferimento/am_rx.grc
```

Obiettivo

Estrarre una stazione AM a 880 kHz da una registrazione:

- centrata a 900 kHz;
- campionata a 400 kS/s;
- memorizzata come campioni complessi.

Catena della soluzione

```
1 File IQ
2 -> mixer con oscillatore complesso
3 -> filtro passa-basso
4 -> demodulatore AM
5 -> ricampionatore 400 kHz -> 32 kHz
6 -> Audio Sink
```

Passaggio 1: leggere lo spettro

Il `Frequency Sink` collegato direttamente al file permette di individuare la stazione a meno 20 kHz:

$$880 \text{ kHz} - 900 \text{ kHz} = -20 \text{ kHz}$$

Passaggio 2: portare il canale a zero

L'oscillatore è impostato a:

```
1 center_freq - freq = 20 kHz
```

La moltiplicazione sposta la stazione da meno 20 kHz a zero.

Passaggio 3: selezionare la banda audio

Il filtro passa-basso ha:

- taglio a 5 kHz;
- transizione di 1 kHz;
- guadagno 10;
- nessuna decimazione.

Il filtro elimina le altre stazioni e conserva la banda laterale necessaria alla voce AM.

Passaggio 4: demodulare

Il blocco `AM Demod` trasforma il segnale complesso modulato in ampiezza in un segnale reale audio. La soluzione mantiene un channel rate di 400 kS/s e applica la conversione di rate dopo la demodulazione.

Passaggio 5: ricampionare

Il `Rational Resampler` usa interpolazione 32 e decimazione 400:

$$400000 \cdot \frac{32}{400} = 32000$$

L'uscita è quindi compatibile con l'`Audio Sink` a 32 kHz.

Cosa osservare

- prima del mixer: stazione spostata rispetto allo zero;
- dopo il mixer: stazione centrata;
- dopo il filtro: un solo canale;
- dopo il demodulatore: spettro audio prevalentemente sotto 5 kHz.

Problemi tipici

- segno errato dell'oscillatore;
- file source configurato come `float` invece di `complex`;

- mancato ricampionamento verso 32 kHz;
- filtro troppo largo, con interferenze;
- filtro troppo stretto, con audio ovattato.

Unità 3 — Dimostrazioni ed esperimenti su segnali, spettro, guadagno e filtri

Tipo di attività: flowgraph completi da eseguire, modificare e osservare; non sono esercizi con soluzione separata.

Dimostrazione ed esperimento: [single_tone.grc](#)

Obiettivo

Generare un tono regolabile e confrontare frequenza, periodo e percezione uditiva.

Con un tono iniziale di 500 Hz e sample rate di 32 kHz:

$$N = \frac{32000}{500} = 64 \text{ campioni per periodo}$$

Aumentando la frequenza, il periodo si accorcia e il suono diventa più acuto.

Dimostrazione ed esperimento: [single_tone_fft.grc](#)

Obiettivo

Confrontare dominio del tempo e dominio della frequenza.

Il [Time Sink](#) mostra la sinusoide; il [Frequency Sink](#) mostra un picco a 500 Hz. Poiché il segnale è reale, la rappresentazione matematica contiene anche la componente simmetrica a meno 500 Hz, anche se la configurazione del sink può mostrare soltanto la metà positiva.

Dimostrazione ed esperimento: [voice_fft.grc](#)

Obiettivo

Osservare che la voce non è un singolo tono, ma un segnale con energia distribuita nel tempo e nella frequenza.

Procedura

1. correggere il percorso del WAV;
2. eseguire il file in ripetizione;
3. osservare il [Time Sink](#) ;
4. osservare il [Frequency Sink](#) ;
5. individuare la parte dello spettro in cui si concentra la maggior parte dell'energia.

Il file WAV è campionato a 32 kHz, quindi la massima frequenza teoricamente osservabile è 16 kHz.

Dimostrazione: `cmajor.grc`

Obiettivo

Costruire un segnale composto sommando tre sinusoidi.

Le frequenze sono circa:

- 523,25 Hz;
- 1318,5 Hz;
- 1568 Hz.

Nel dominio del tempo la somma appare complessa; nello spettro si distinguono chiaramente i tre picchi. Questo esercizio mostra perché l'analisi in frequenza è utile per separare componenti che nel tempo si sovrappongono.

Esperimento: `gain.grc`

Obiettivo

Studiare il guadagno lineare.

Il segnale in uscita è:

$$y[n] = Gx[n]$$

Confrontare i due `Time Sink`, uno all'ingresso e uno all'uscita. La frequenza non cambia; cambia soltanto l'ampiezza.

Se il segnale viene inviato all'audio, un guadagno eccessivo può produrre clipping. Per un segnale normalizzato è prudente mantenere l'ampiezza tra meno 1 e più 1.

Esperimento: `gain_db.grc`

Obiettivo

Osservare nel `Frequency Sink` la relazione logaritmica tra guadagno lineare e dB.

Il sink dispone di due ingressi, così è possibile confrontare segnale originale e amplificato. Provare valori di guadagno 0,5, 1, 2, 10 e 100 e confrontarli con la formula:

$$20 \log_{10}(G)$$

Dimostrazione: `cmajor_lpf.grc`

Obiettivo

Conservare soltanto la componente più bassa.

Il taglio a 550 Hz lascia passare il tono a 523,25 Hz e attenua quelli a 1318,5 Hz e 1568 Hz.

Dimostrazione: `cmajor_lpf2.grc`

Obiettivo

Conservare le prime due componenti.

Il taglio a 1350 Hz lascia passare 523,25 Hz e 1318,5 Hz, mentre attenua 1568 Hz. La distanza tra 1350 Hz e 1568 Hz non è molto grande; la larghezza di transizione di 100 Hz è quindi importante.

Dimostrazione: [cmajor_hpfp.grc](#)

Obiettivo

Conservare la componente più alta.

Il taglio a 1500 Hz attenua i due toni inferiori e lascia passare quello a 1568 Hz.

Dimostrazione: [cmajor_bpfp.grc](#)

Obiettivo

Isolare il tono centrale.

La banda passante da 1250 Hz a 1350 Hz include 1318,5 Hz e rifiuta gli altri due toni.

Dimostrazione: [cmajor_brfp.grc](#)

Obiettivo

Eliminare una regione di frequenze.

La banda rifiutata da 600 Hz a 1500 Hz attenua il tono a 1318,5 Hz. Il tono a 523,25 Hz rimane sotto la banda rifiutata, mentre quello a 1568 Hz rimane sopra.

Esperimento guidato: [equalizer.grc](#)

Obiettivo

Costruire un equalizzatore a tre bande.

Le bande sono:

Banda	Intervallo
Bassi	20–400 Hz
Medi	400–2600 Hz
Alti	2600–10000 Hz

Il WAV viene inviato in parallelo ai tre filtri. Ogni ramo applica un guadagno indipendente, poi i risultati vengono sommati.

Concetto importante

Un equalizzatore non “crea” frequenze mancanti. Amplifica o attenua ciò che è già presente in ciascuna banda.

Esperimenti consigliati

- mettere bassi, medi e alti al 50 per cento;
- azzerare una banda alla volta;
- portare una sola banda al 100 per cento;
- osservare insieme effetto uditivo e spettro.

Unità 4 — Laboratorio: traslazione, tuning e decimazione

Tipo di attività: laboratorio con flowgraph base, dati e soluzioni di riferimento.

Laboratorio guidato: `freq_shift.grc`

Obiettivo

Spostare lo spettro di una registrazione IQ mediante moltiplicazione complessa.

Catena

```
1 File IQ -> Throttle -> Multiply con oscillatore complesso -> Frequency Sink
```

Il parametro iniziale `tune_freq` è meno 3,5 kHz. Confrontare lo spettro originale e quello traslato.

Metodo

1. individuare un picco nello spettro originale;
2. annotarne la frequenza;
3. applicare uno shift noto;
4. misurare la nuova posizione;
5. verificare che la differenza coincida con lo shift.

Laboratorio guidato: `tuner.grc`

Obiettivo

Costruire un semplice sintonizzatore digitale.

La soluzione usa:

- file IQ a 1 MS/s;
- frequenza di sintonia 435 kHz;
- oscillatore a meno 435 kHz;
- filtro passa-basso con banda di canale 60 kHz.

Catena

```
1 File IQ -> mixer a -435 kHz -> filtro passa-basso -> sink
```

Il mixer porta il canale desiderato a zero. Il filtro, con taglio a metà della larghezza di canale, conserva circa più o meno 30 kHz.

Perché il filtro viene dopo il mixer

Prima del mixer il canale utile è lontano dallo zero. Un filtro passa-basso lo eliminerebbe. Dopo la traslazione il canale è centrato e può essere isolato con un passa-basso.

Esperimento guidato: [decimation.grc](#)

Obiettivo

Osservare l'effetto di una decimazione di fattore 4.

Il sample rate passa da 32 kS/s a:

$$\frac{32000}{4} = 8000 \text{ S/s}$$

Il segnale di prova è a 1 kHz, quindi dopo la decimazione rimangono otto campioni per periodo. Il tono resta rappresentabile perché 1 kHz è inferiore al nuovo limite di Nyquist di 4 kHz.

Esperimento sull'aliasing

Aumentare la frequenza del tono oltre 4 kHz. Il blocco [Keep 1 in N](#) non applica un filtro anti-alias, quindi la componente apparirà a una frequenza errata. Questo comportamento prepara l'esercizio 9.

Collegamento con [am_rx.grc](#)

Il ricevitore AM viene riproposto come base da migliorare. La lezione dell'esercizio è che sintonia, filtro e riduzione del rate possono essere integrati in modo più efficiente, invece di elaborare tutto il flusso a 400 kS/s fino alla fine.

Unità 5 — Laboratorio: ricevitore FM

Tipo di attività: costruzione guidata a partire da una catena AM, con due soluzioni di riferimento.

Preparazione della registrazione

Estrarre:

```
1 unita/05_laboratorio_ricevitore_fm/dati/fm_capture.zip
```

in modo da ottenere:

```
1 unita/05_laboratorio_ricevitore_fm/dati/fm_c96M_s8M.iq
```

La registrazione è centrata a 96 MHz e campionata a 8 MS/s. La stazione iniziale è a 94,9 MHz.

Soluzione di riferimento: [fm_rx.grc](#)

Calcolo della posizione del canale

$$94,9 \text{ MHz} - 96 \text{ MHz} = -1,1 \text{ MHz}$$

La stazione è a meno 1,1 MHz nella registrazione.

Catena

```
1 File IQ
2 -> mixer complesso a +1,1 MHz
3 -> passa-basso da 75 kHz con decimazione 20
4 -> WFM Receive a 400 kS/s
5 -> ricampionamento a 32 kS/s
6 -> Audio Sink
```

Riduzione del sample rate

Il filtro decima di 20:

$$\frac{8 \text{ MHz}}{20} = 400 \text{ kHz}$$

Questo rate è indicato come `quad_rate` nel demodulatore WFM.

Il ricampionatore usa 32/400:

$$400000 \frac{32}{400} = 32000$$

Soluzione migliorata: `fm_upgrade.grc`

Obiettivo

Sostituire mixer e filtro separati con un `Frequency Xlating FIR Filter`.

Il blocco usa:

- frequenza di traslazione `freq - center_freq`, cioè meno 1,1 MHz;
- banda di canale 150 kHz;
- cutoff a metà banda, 75 kHz;
- decimazione `8 MHz / 400 kHz = 20`;
- larghezza di transizione `150 kHz / 20 = 7,5 kHz`.

La differenza di segno rispetto al mixer separato dipende dalla convenzione interna del blocco.

La verifica corretta consiste nell'osservare che il canale selezionato arrivi centrato a zero.

Vantaggi

- meno blocchi;
- meno elaborazione separata;
- parametri collegati a variabili;
- sintonia modificabile dalla GUI;
- controllo del volume tramite moltiplicazione costante.

Diagnosi dei problemi

Se non si sente audio:

1. verificare che il file sia stato estratto nella posizione corretta;
2. controllare nel primo sink che la stazione esista;
3. verificare la posizione a meno 1,1 MHz;
4. controllare che l'uscita del filtro sia a 400 kS/s;
5. controllare il `quad_rate` del WFM Receiver;
6. controllare che l'audio finale sia a 32 kS/s;
7. abbassare o aumentare il volume senza superare il range audio.

Unità 6 — Dimostrazioni e strumenti: banda occupata e visualizzazione dei file

Tipo di attività: esperimenti di osservazione e strumenti diagnostici riutilizzabili.

Esperimento dimostrativo: `bandwidth.grc`

Obiettivo

Studiare la banda utile di un segnale audio applicando un filtro passa-basso al file WAV.

Il cutoff iniziale è 16 kHz, uguale al limite di Nyquist per un file a 32 kS/s. Ridurre progressivamente il cutoff e osservare:

- quali componenti spariscono dallo spettro;
- quando la voce perde brillantezza;
- quando l'intelligibilità comincia a diminuire.

L'esperimento dimostra che banda teoricamente disponibile e banda realmente necessaria non sono la stessa cosa.

Strumento dimostrativo: `file_viewer.grc`

Obiettivo

Costruire un visualizzatore generico per file IQ.

Catena

```
1 File Source complessa -> Throttle a 8 MS/s -> Frequency Sink
```

Il `Throttle` è necessario perché un file letto dal disco non impone automaticamente un ritmo reale. Il parametro del `Frequency Sink` deve essere 8 MHz; altrimenti l'asse delle frequenze risulta sbagliato anche se i campioni sono corretti.

Uso come strumento diagnostico

Prima di progettare un ricevitore:

1. aprire la registrazione nel viewer;
2. confermare sample rate e tipo;
3. individuare il canale;
4. stimarne la larghezza;
5. calcolare la frequenza relativa;
6. solo dopo costruire mixer e filtri.

Unità 7 — Laboratorio misto: modulazioni analogiche e OOK

Tipo di attività: alcuni flowgraph sono esercizi con soluzione; altri sono dimostrazioni da osservare.

Esercizio: `amp_mod_begin.grc`

Obiettivo

Costruire una modulazione di ampiezza moltiplicando un messaggio per una portante.

Il messaggio è un'onda triangolare a 1 kHz. La portante complessa è a 10 kHz.

Una semplice moltiplicazione produce una modulazione a portante soppressa se il messaggio ha media nulla. Nel flowgraph iniziale il messaggio presenta un offset negativo, utile per osservare che un inviluppo che attraversa o supera lo zero genera inversioni di fase e non può essere recuperato correttamente con una semplice rivelazione d'involuppo.

Soluzione: `amp_mod_final.grc`

Correzione

La soluzione aggiunge una costante al messaggio prima della moltiplicazione:

```
1 messaggio -> Add Const 1 -> conversione complessa -> moltiplicazione per portante
```

L'involuppo diventa proporzionale a $1 + m(t)$. La variabile `amplitude` permette di osservare il confine tra modulazione corretta e sovr modulazione.

Verifica con il modulo

Il blocco `Complex to Mag` calcola l'involuppo del segnale modulato. Con modulazione corretta, la sua forma segue il messaggio traslato verso valori non negativi.

Dimostrazione ed esperimento: `amp_mod_ook.grc`

Obiettivo

Realizzare una modulazione On-Off Keying.

Un'onda quadra controlla la portante:

- valore alto: portante presente;
- valore basso: portante assente.

Osservare l'uscita nel tempo. Il modulo del segnale mostra chiaramente gli intervalli ON e OFF.

Dimostrazione ed esperimento: `phase_mod.grc`

Obiettivo

Confrontare una modulazione di fase con una rappresentazione ottenuta moltiplicando una portante.

Il blocco `Phase Mod` produce direttamente un segnale complesso la cui fase dipende dal messaggio. Convertendo la parte reale in un segnale osservabile nel tempo si nota che l'ampiezza rimane circa costante, mentre cambiano gli attraversamenti dello zero e quindi la fase istantanea.

Dimostrazione ed esperimento: `fm_mod.grc`

Obiettivo

Studiare un modulatore di frequenza e osservare il legame tra messaggio e frequenza istantanea.

Il messaggio triangolare lento rende facile vedere la compressione e l'espansione dei cicli nel `Time Sink`.

Il parametro `sensitivity=1` è intenzionalmente astratto. Per ottenere una deviazione fisica definita si passa alla soluzione successiva.

Esercizio di osservazione: `fm_waterfall.grc`

Obiettivo

Osservare nel waterfall come lo spettro FM si sposti nel tempo seguendo il messaggio.

Il waterfall usa:

- asse orizzontale: frequenza;
- asse verticale: tempo;
- intensità: potenza.

Con un messaggio triangolare lento, la frequenza istantanea disegna una traiettoria periodica.

Soluzione parametrizzata: `fm_waterfall_final.grc`

Parametrizzazione fisica

La soluzione introduce `deviation=20e3` e calcola:

```
1 sensitivity = 2*pi*deviation/samp_rate
```

Con sample rate 360 kHz la sensibilità è circa 0,349 radianti per campione per unità di messaggio.

Esercizio e soluzione: [fm_rx.grc](#) / [fm_rx_final.grc](#)

Obiettivo

Aggiungere un punto di osservazione dopo il demodulatore FM per studiare lo spettro del segnale recuperato.

La soluzione collega un [Frequency Sink](#) reale all'uscita del [WFM Receive](#). Questo permette di distinguere:

- banda RF prima della demodulazione;
- spettro audio dopo la demodulazione;
- effetto del ricampionamento finale.

Unità 8 — Esperimenti diagnostici e mini-progetto

Tipo di attività: dimostrazioni intenzionali di aliasing e overflow, seguite da un mini-progetto sulla sintonia automatica.

Dimostrazione: [nyquist.grc](#)

Obiettivo

Produrre deliberatamente aliasing.

La sorgente genera un tono complesso a 32 kS/s. Il blocco [Keep 1 in N](#) conserva un campione ogni 32, quindi il rate effettivo diventa:

$$\frac{32000}{32} = 1000 \text{ S/s}$$

La banda rappresentabile dopo la decimazione è da meno 500 Hz a più 500 Hz.

La frequenza del tono può variare tra 0 e 2 kHz. Finché resta sotto 500 Hz, appare nella posizione corretta. Oltre 500 Hz, si ripiega periodicamente nella banda visibile.

Esempi

Frequenza originale	Frequenza osservata dopo campionamento a 1 kS/s
200 Hz	200 Hz
400 Hz	400 Hz
600 Hz	-400 Hz
800 Hz	-200 Hz

1000 Hz	0 Hz
1200 Hz	200 Hz

Per un segnale complesso, la frequenza aliasata può essere espressa scegliendo il multiplo di f_s che riporta la componente nell'intervallo da meno $f_s/2$ a più $f_s/2$.

Dimostrazione: [fm_rx_ovf_hackrf.grc](#)

Obiettivo

Osservare un overflow aumentando il sample rate a 20 MS/s.

Un overflow indica che il sistema non riesce a consumare i campioni alla velocità con cui arrivano. Alcuni campioni vengono persi, con possibili interruzioni audio e discontinuità nello spettro.

Possibili cause

- sample rate USB troppo alto;
- filtro molto costoso;
- FFT troppo grandi o aggiornate troppo spesso;
- più sink GUI contemporanei;
- computer o bus insufficienti;
- altri processi che consumano risorse.

Strategie

1. ridurre il sample rate;
2. diminuire il numero o la frequenza di aggiornamento dei sink;
3. decimare il prima possibile;
4. allargare ragionevolmente la transizione del filtro;
5. evitare elaborazioni duplicate ad alto rate;
6. chiudere applicazioni non necessarie.

Configurazione di confronto: [fm_hackrf.grc](#)

Il flowgraph riprende il ricevitore hardware a 8 MS/s. La presenza del flowgraph con overflow permette il confronto tra una configurazione sostenibile e una troppo impegnativa.

Esercizio e soluzione: [fm_hackrf_auto_cf.grc](#)

Obiettivo

Calcolare automaticamente la center frequency in funzione della stazione desiderata:

```
1 center_freq = freq - samp_rate/4
```

Con:

- `freq = 94,9 MHz` ;
- `samp_rate = 8 MHz` ;

si ottiene:

$$94,9 \text{ MHz} - 2 \text{ MHz} = 92,9 \text{ MHz}$$

La stazione si trova quindi a più 2 MHz rispetto al centro, cioè a un quarto del sample rate.

Perché non porre sempre il canale esattamente a zero hardware

Alcuni ricevitori presentano una componente continua o artefatti vicino allo zero dello spettro. Posizionare il canale lontano dal centro e traslarlo digitalmente può ridurre l'impatto di questi disturbi.

Vantaggio della sintonia automatica

Quando cambia `freq`, cambia anche `center_freq`, mantenendo il canale nella stessa posizione relativa. Il filtro traslante continua quindi a lavorare con un offset prevedibile.

Unità 9 — Dimostrazione integrativa della catena FM

Tipo di attività: flowgraph completo già fornito, da usare per esperimenti di sintonia e rumore.

Dimostrazione completa: `fm_tx.grc`

Obiettivo

Simulare una catena completa:

```
1 WAV -> modulatore WFM -> ricampionamento -> traslazione RF
2      -> aggiunta di rumore -> selezione del canale
3      -> demodulatore WFM -> Audio Sink
```

Il flowgraph non utilizza un sink hardware di trasmissione: la catena RF è simulata internamente.

Stadio 1: sorgente audio

Il file `HumanEvents_s32k.wav` è campionato a 32 kHz. Questo coincide con `audio_samp_rate`.

Stadio 2: modulatore WFM

Il blocco `WFM Transmit` converte l'audio a un `quad_rate` di 320 kS/s e usa:

- deviazione massima 75 kHz;
- costante di de-enfasi o pre-enfasi configurata con $\tau = 75 \mu s$ secondo il comportamento del blocco;
- audio rate 32 kHz.

Il rapporto tra 320 kS/s e 32 kS/s è 10.

Stadio 3: ricampionamento a 8 MS/s

Il `Rational Resampler` interpola di:

$$\frac{8 \text{ MHz}}{320 \text{ kHz}} = 25$$

Il segnale modulato passa quindi da 320 kS/s a 8 MS/s.

Stadio 4: traslazione alla frequenza del canale

La frequenza centrale simulata è 100 MHz, mentre la trasmissione è impostata a 98,1 MHz.

L'oscillatore usa:

$$98,1 - 100 = -1,9 \text{ MHz}$$

La moltiplicazione sposta il segnale FM a meno 1,9 MHz nella banda base complessa simulata.

Stadio 5: canale con rumore

Un generatore di rumore gaussiano complesso viene sommato al segnale. La variabile `noise` permette di studiare il degrado del rapporto segnale-rumore.

Esperimento:

1. impostare rumore quasi nullo;
2. ascoltare e osservare lo spettro;
3. aumentare gradualmente il rumore;
4. individuare il punto in cui l'audio diventa rumoroso;
5. confrontare l'effetto FM con quanto ci si aspetterebbe da una semplice modulazione di ampiezza.

Stadio 6: ricevitore

Il `Frequency Xlating FIR Filter` seleziona `freq_rx`, decima da 8 MS/s a 320 kS/s e porta il canale a zero.

Il `WFM Receive` usa:

- quad rate 320 kS/s;
- audio decimation $320000/32000 = 10$.

L'uscita è quindi direttamente a 32 kS/s e può essere collegata all' `Audio Sink`.

Esperimenti di sintonia

Trasmittitore e ricevitore allineati

Impostare:

```
1 freq_tx = freq_rx
```

L'audio deve essere recuperato correttamente.

Ricevitore disallineato

Spostare `freq_rx` di alcune decine di kHz. Il canale entra progressivamente nella transizione del filtro e la qualità peggiora.

Canale fuori banda

Spostare `freq_rx` oltre la larghezza del filtro. Il ricevitore non deve più recuperare il segnale utile.

Concetti riassunti dal laboratorio finale

La dimostrazione combina quasi tutte le nozioni precedenti:

- audio reale;
- modulazione FM;
- conversione di sample rate;
- traslazione complessa;
- canale rumoroso;
- selezione con filtro traslante;
- decimazione;
- demodulazione;
- uscita audio.