

Este Material é Fruto do Programa BIA(Bolsas de Iniciação Acadêmica), da Universidade Federal do Ceará.



ESTUDO DE SINAIS

COM :

GNU RADIO

Uma imersão no estudo de sinais
utilizando o software GNU Radio

Autores :

Acrisio Caxias
Brenno Azevedo

Orientado por :

Dr. Yuri Lima

1º Edição



Prefácio :



Olá, como vai querido leitor ? Acho importante que tenha chegado até aqui ! Saiba que você é a pessoa mais importante para nós neste momento, sua atenção é vital para nossa existência, vamos juntos iniciar uma jornada incrível de aprendizado :

Seja Bem Vindo(a) !!!



Dito isso, vamos acertar alguns pontos relevantes, a nossa jornada de aprendizado será trilhada de forma dinâmica e divertida, o principal objetivo é que possamos juntos compreender sobre sinais. Para isso iremos percorrer um caminho em conceitos como serie de Fourier, transformada de Fourier, filtros, modulações, amostragem, convolução e afins. A plataforma GNU Radio será nossa principal aliada, ela nos permitira visualizar na prática a partir de exemplos, todos estes conceitos citados, de forma a facilitar o aprendizado.



Organização :

Bem, nossa jornada se dará a partir da dinâmica de um game, na qual teremos que passar por 6 fases, cada uma ira lhe atribuir algumas habilidades , vejamos os desafios :

1° - Fundamentos Base de Sinais ;

2° - A prática Inicial ;

3° - Explorando os Filtros ;

4° - O Que São os Ruidos ;

5° - Navegando nas Modulações ;

6° - Conclusão ;

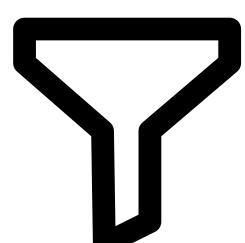
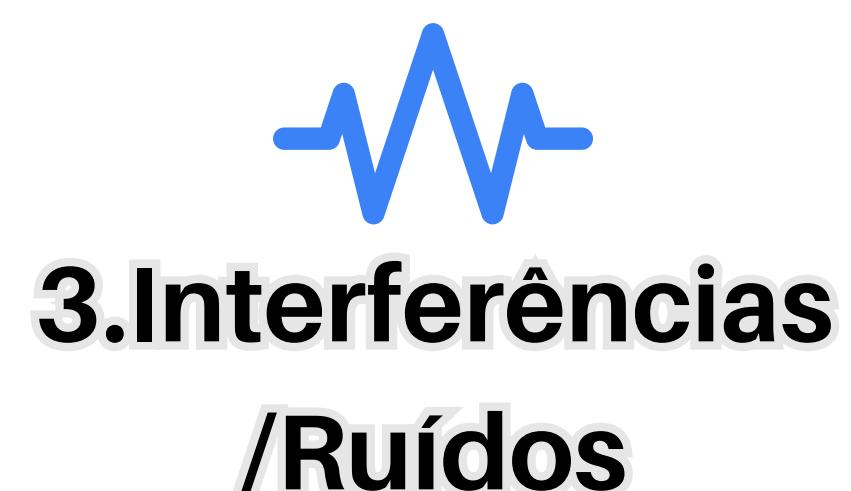
Há, ja ia esquecendo, ao final de cada fase teremos uma batalha. As batalhas se constituirão de exemplos a serem aplicados no GNU Radio, o objetivo de todas as missões é de vencer todas as duvidas ;).

Sigam-me os bons, vamos !?



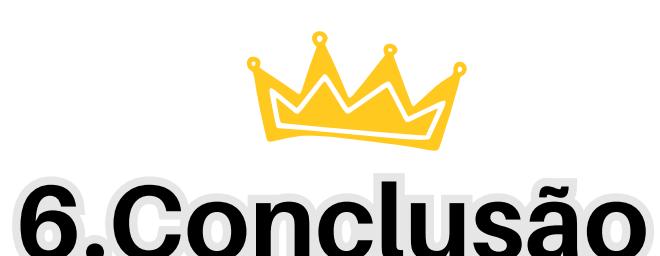
|||| Progresso:

Ao longo dessa jornada, você vai poder acompanhar seu progresso, para não correr o risco de se perder viu. ☺



4. Filtros

5. Modulações



1.0 - Conhecendo o GNU Radio :



O GNU Radio é um software de código aberto que possui um conjunto de ferramentas amplamente utilizadas para o desenvolvimento de sistemas de processamento de sinais de rádio. Ele traz diversas vantagens, como um ambiente flexível para a criação, simulação e implementação de sistemas de comunicação por rádio definido por software (SDR) e é usado por engenheiros, pesquisadores e entusiastas em todo o mundo.



Curiosidade : Utiliza a linguagem python , é estruturado em blocos, possui recursos gráficos para simulações ;

1.1 - Vamos a instalação :

- Vamos iniciar com a instalação, vá até a pagina inicial para realizar o download (<https://wiki.gnuradio.org/index.php/InstallingGR>) ;
- Selecione o sistema operacional de sua maquina ;
- Siga o guia de instalação baseado no seu sistema ;
- Uma vez instalado, fique a vontade para navegar na aplicação e conhecer suas funcionalidades (qualquer duvida, consulte o forum “gnuradio.org”);

1.2 - Como utilizar :

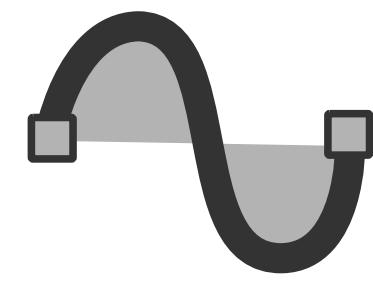
- Para criar um novo projeto, clique em File > New > QT Gui ;
- Certifique-se se salvar o arquivo corretamente na pasta de sua preferência ;
- Utilize sempre a extensão padrão .grc ;
- Monte sua aplicação com os bloquinhos, lembre-se de seguir o fluxo de blocos de entrada, processamento e saída;
- Obá, agora é só clicar na setinha de executar e visualizar sua aplicação em funcionamento ;



Acesse os tutoriais : (abre.ai/estudodesinaiscomgnu)



2.0 - Fundamentos de Sinais:



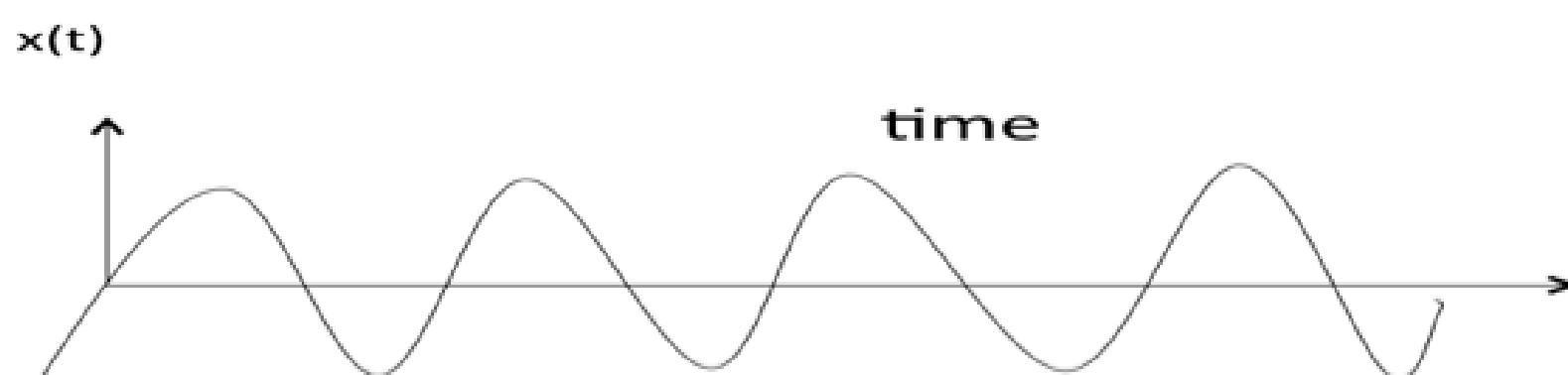
2.1 - O que é um sinal?:

Então vamos lá conhecer alguns conhecimentos básicos de sinais!

Basicamente, um sinal é um fenômeno físico que varia no tempo e tem como objetivo transmitir informações.

Exemplo: sinal de voz, sinal de vídeo, sinais em fios telefônicos, etc.

Nota: O ruído também é um sinal, mas a informação transmitida pelo ruído é considerada indesejável.



Esses sinais possuem aplicações bem vastas em muitas áreas, principalmente nos dias de hoje. Não atoa que a área de processamento de sinais é bem requisitada.

Exemplos de utilização: Sistemas de comunicação, processamento de áudio e imagem, medicina, automação industrial e controle de processos, entre outros.

2.2 - Classificação dos sinais:

2.2.1 - Sinal par e ímpar:

Diz-se que um sinal é par quando satisfaz a condição:

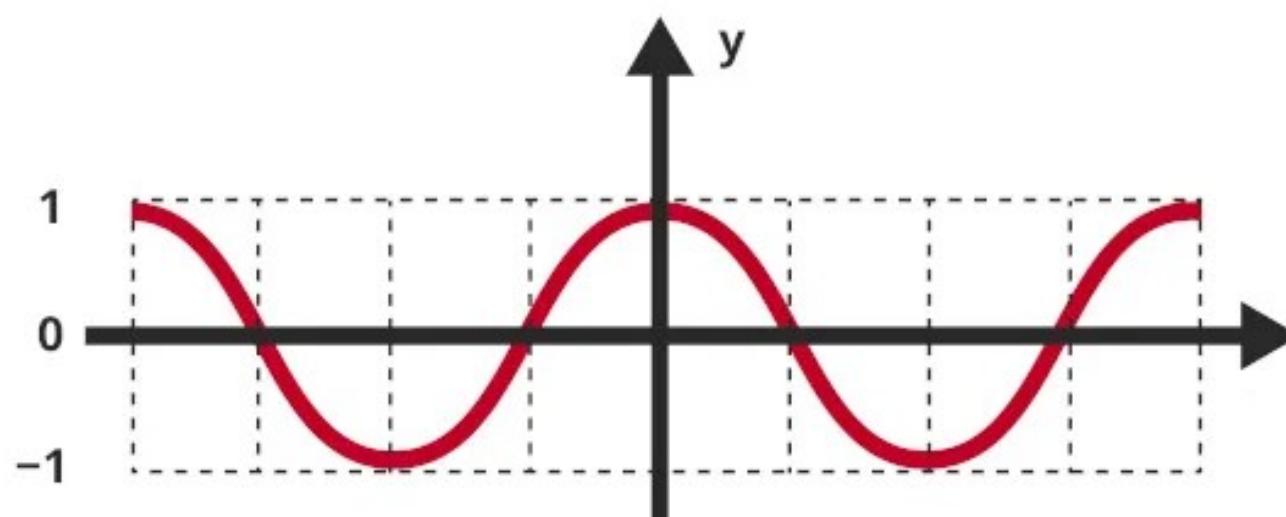
$$x(t) = x(-t)$$

Isso quer dizer que ele é simétrico!

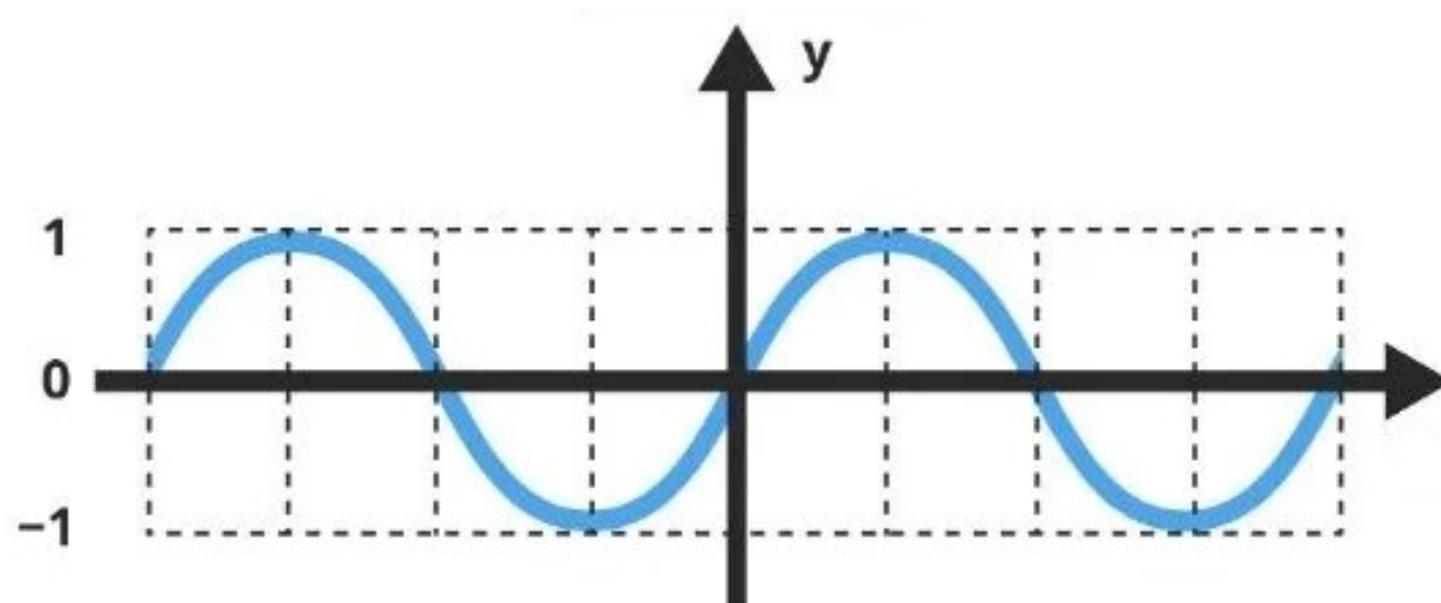
Ou seja, que seu lado esquerdo é igual ao lado direito, tendo o eixo y como ponto central.

Um exemplo bastante simples é o da função cosseno:

Perceba a simetria dessa cossenoide



- Sabendo disso, um sinal ímpar é... Isso mesmo!! Ele é assimétrico. Assim como essa senoide da imagem a seguir:



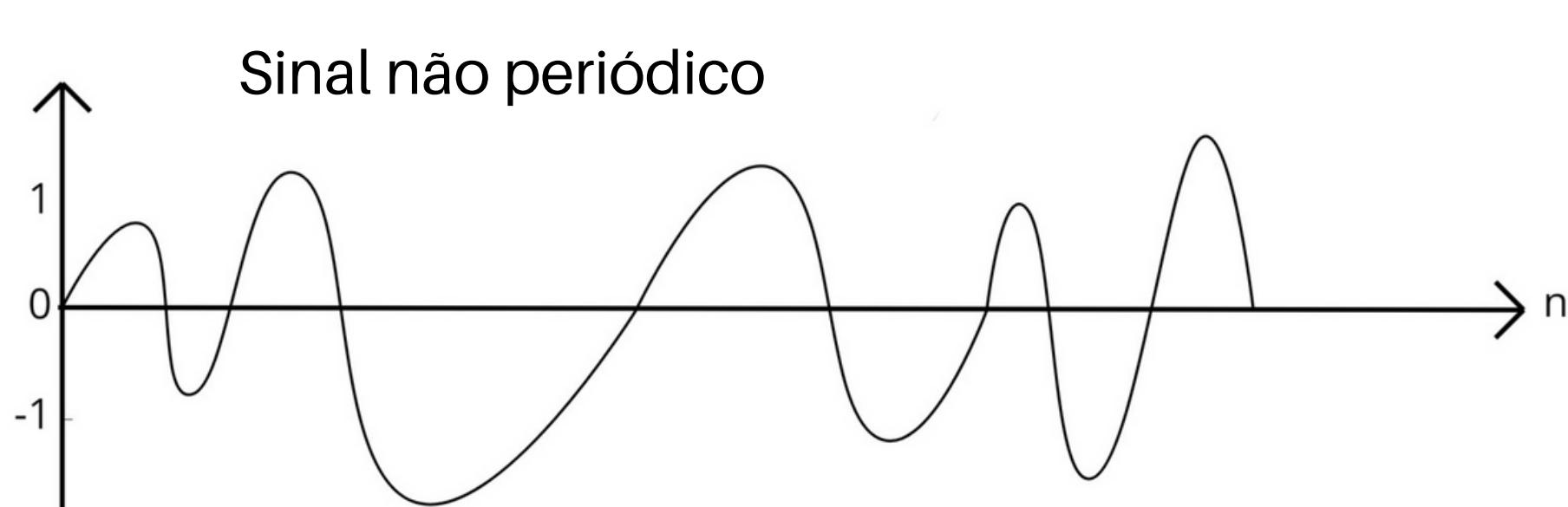
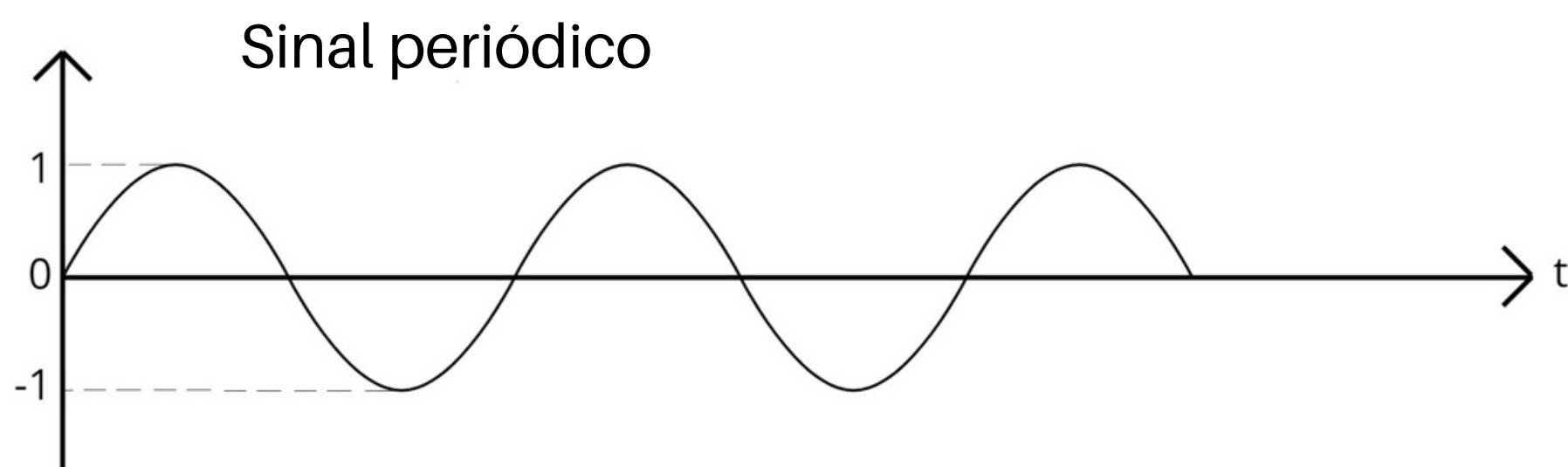
- Os lados são semelhantes, mas não são simétricos, isso por que não acontece a característica de “espelhamento” de um lado com o outro.

2.2.2 - Sinais Periódicos e Aperiódicos:

Um sinal periódico é um sinal que se repete em intervalos regulares de tempo. O período e a frequência de um sinal periódico são inversamente relacionados: um sinal com período curto tem alta frequência e um sinal com período longo tem baixa frequência.

Eles são importantes em muitas aplicações porque possuem um espectro de frequência bem definido e podem ser facilmente analisados usando ferramentas de análise de Fourier. Já um sinal aperiódico é um sinal que não se repete em intervalos regulares, e portanto não possui uma frequência fundamental.

Exemplos de sinais não periódicos incluem ruído branco, impulsos e formas de onda arbitrárias. Sinais não periódicos são frequentemente encontrados em aplicações do mundo real, como fala, música e ruído ambiental.



2.2.3 - Sinal digital e analógico:

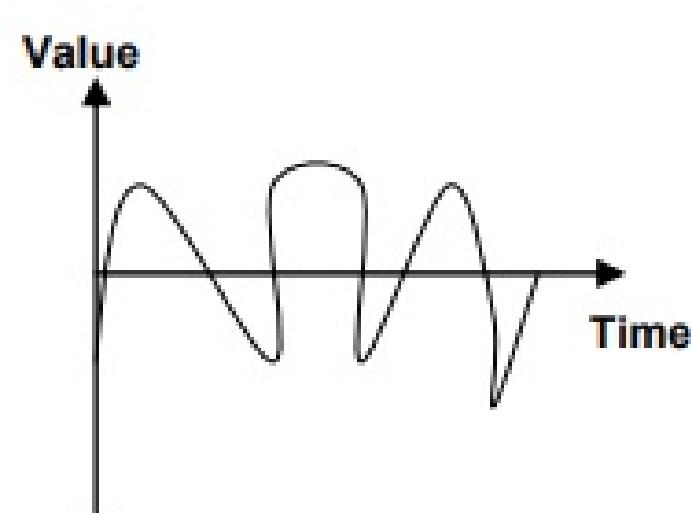
- **Sinal analógico:**

Os sinais analógicos têm infinitos níveis de intensidade durante um período de tempo. Quando a onda se move do valor A para o valor B, ela passa e inclui um número infinito de valores ao longo de seu caminho.

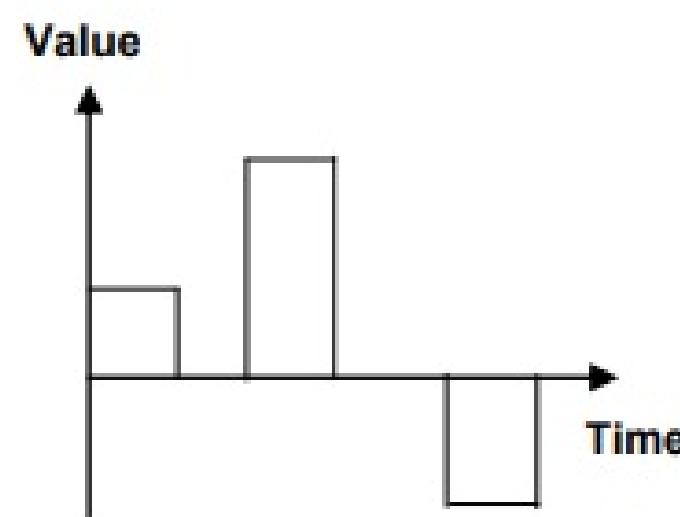
- **Sinal digital:**

Os sinais digitais podem ter apenas um número limitado de valores definidos. Embora cada valor possa ser qualquer número, geralmente é tão simples quanto 0 ou 1.

A maneira mais fácil de mostrar sinais é plotá-los em um par de eixos perpendiculares. O eixo vertical representa o valor ou força de um sinal, ja eixo horizontal representa o tempo.



a. Analog Signal



b. Digital Signal

2.3 - Operações com sinais:

Precisamos também observar algumas operações básicas que aprendemos quando estamos vendo sobre sinais :

- **Escala de amplitude:**

- O que acontece quando multiplicamos uma constante por um sinal, é que a amplitude é escalonada por esse fator.

- **Adição:**

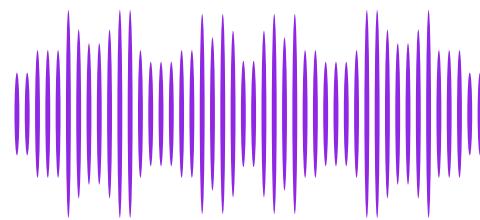
- A adição de dois sinais nada mais é do que a adição de suas amplitudes correspondentes.

- **Multiplicação:**

- Assim como na adição, ao fazer essa operação, ocorre a multiplicação das amplitudes dos dois sinais.

Diante dessas operações, principalmente nas de adição e multiplicação, um novo desenho do sinal é formado. Mas não se preocupe, veremos exemplos com mais calma no próximo estágio com o nosso querido GNU radio.

2.5 - Domínio da Frequência:

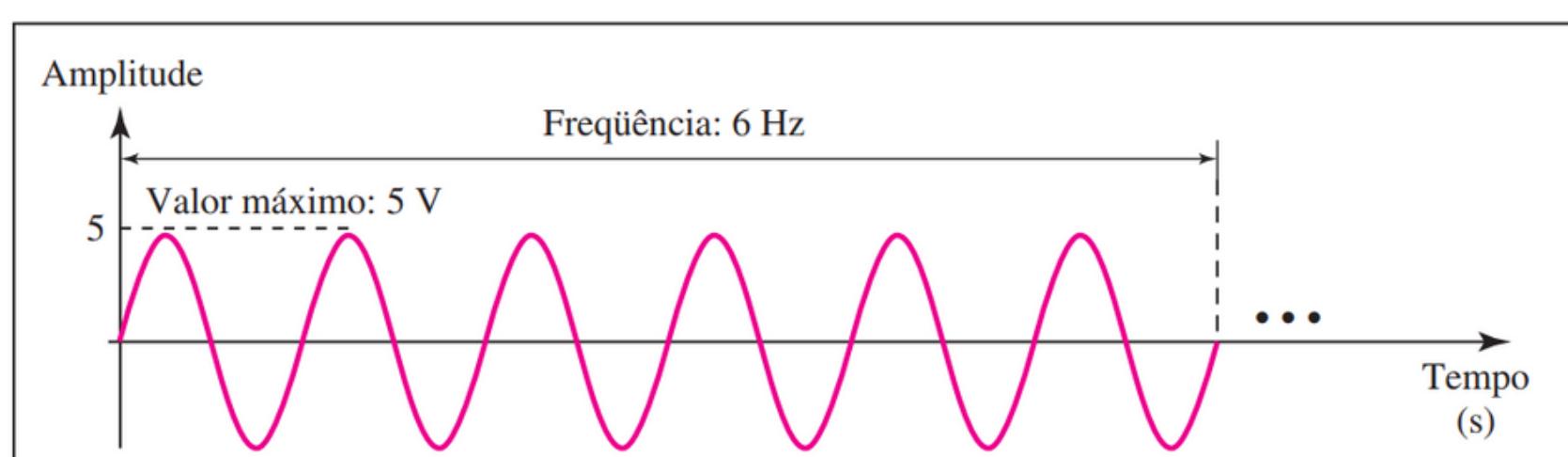


Muito bem, agora começaremos a nos aprofundar em sinais 😊. Para isso conhiceremos o domínio da frequência, que nada mais é que uma forma diferente de visualizar um sinal em um gráfico. E com certeza você já o viu em algum momento de sua vida. Lembra daquele gráfico de um equalizador de áudio? é algo semelhante à aquilo, assim como na imagem a seguir.

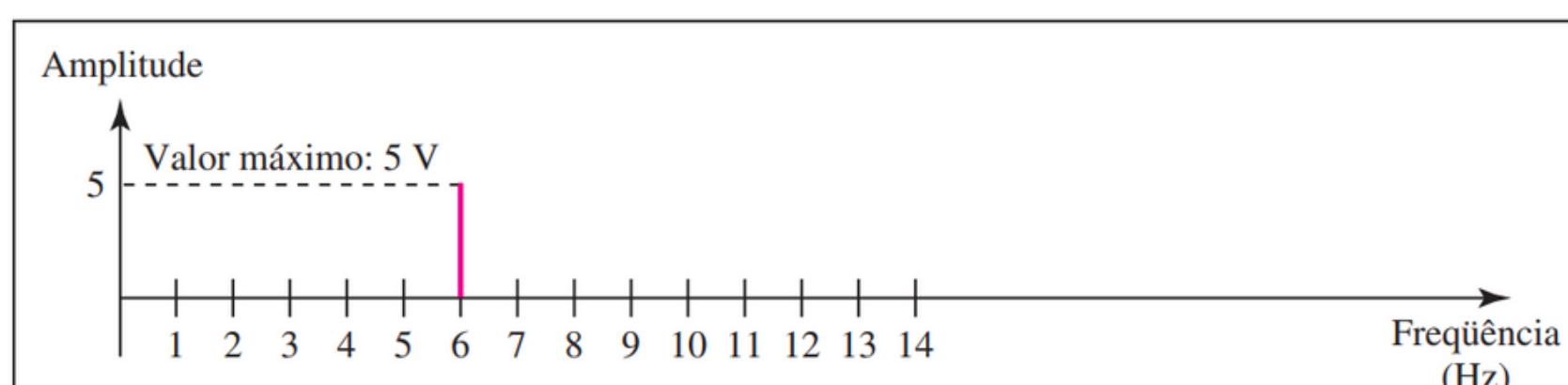


Para ser mais claro, esse gráfico é construído com a relação Amplitude X Frequência, ou seja, cada bloco da imagem anterior representa uma dada frequência, e sua “altura” é sua amplitude.

Mas se ainda estiver confuso, veja esse comparativo da imagem abaixo, que mostra uma onda senoidal no domínio do tempo (Amplitude X Tempo) e no domínio da frequência (Amplitude X Frequência)

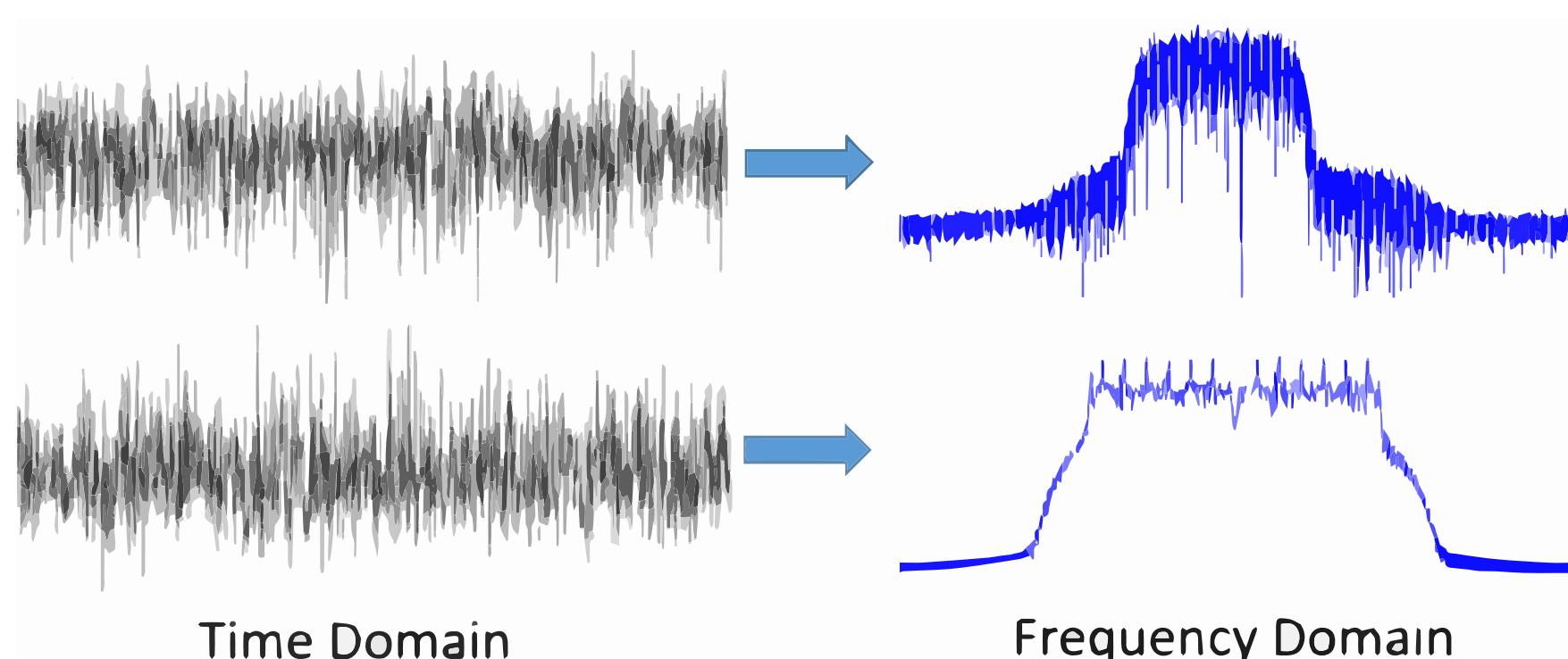


a. Uma onda senoidal no domínio do tempo (valor máximo: 5 V, freqüência: 6 Hz)



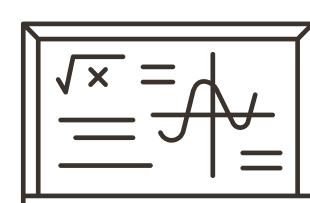
b. A mesma onda senoidal no domínio da freqüência (valor máximo: 5 V, freqüência: 6 Hz)

Tá, mas qual a vantagem de se usar esse gráfico ao invés do tradicional? Bom, a resposta é que em análise de sinais, muitas vezes uma sinal fica muito mais fácil de ver, interpretar e diferenciar quando ele é mostrado na frequência do que no tempo.

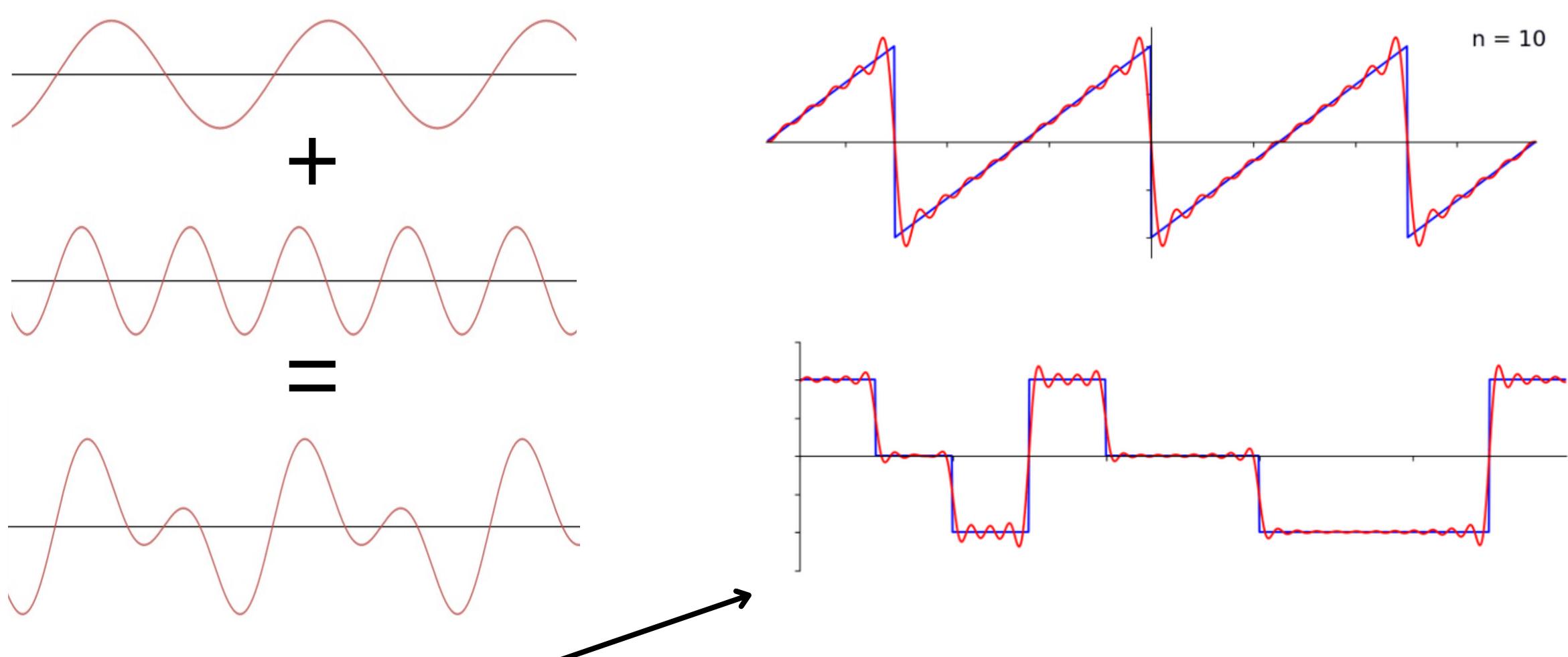


Perceba que esse dois sinais são diferentes, no entanto, no tempo, eles ficam bem parecidos. Mas na frequência a diferença fica bem mais evidente. E outro ponto importante, é que a mudança de fase de um sinal ficará mais evidente também.

2.6 - Série de Fourier:



Série de Fourier se trata da capacidade de representar sinais na forma de somas de ondas senoidais, e quanto mais senoides são adicionadas, mais próximo ela ficará do sinal original. E da mesma forma, podemos pegar um sinal e dividi-lo em outros sinais.



Perceba que na imagem acima, sinal em azul se trata do que queremos “alcançar”, no entanto, devido às transições bruscas, o sinal em vermelho nunca será exatamente igual. 😬

2.7 - Transformada Rápida de Fourier:

Agora, caro leitor, é necessário que tenha conhecimento da Transformada Rápida de Fourier (FFT). Mas o que seria uma transformada? 😊

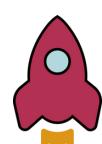
Bom, ela se trata, basicamente de levar o sinal que queremos do domínio do tempo para o domínio da frequência, e o contrário também é possível. Que envolve vários cálculos e propriedades. Mas não se preocupe, não entraremos em detalhes neste momento. 😊

A transformada rápida é simplesmente um algoritmo para calcular a Transformada de Fourier discreta mencionada.

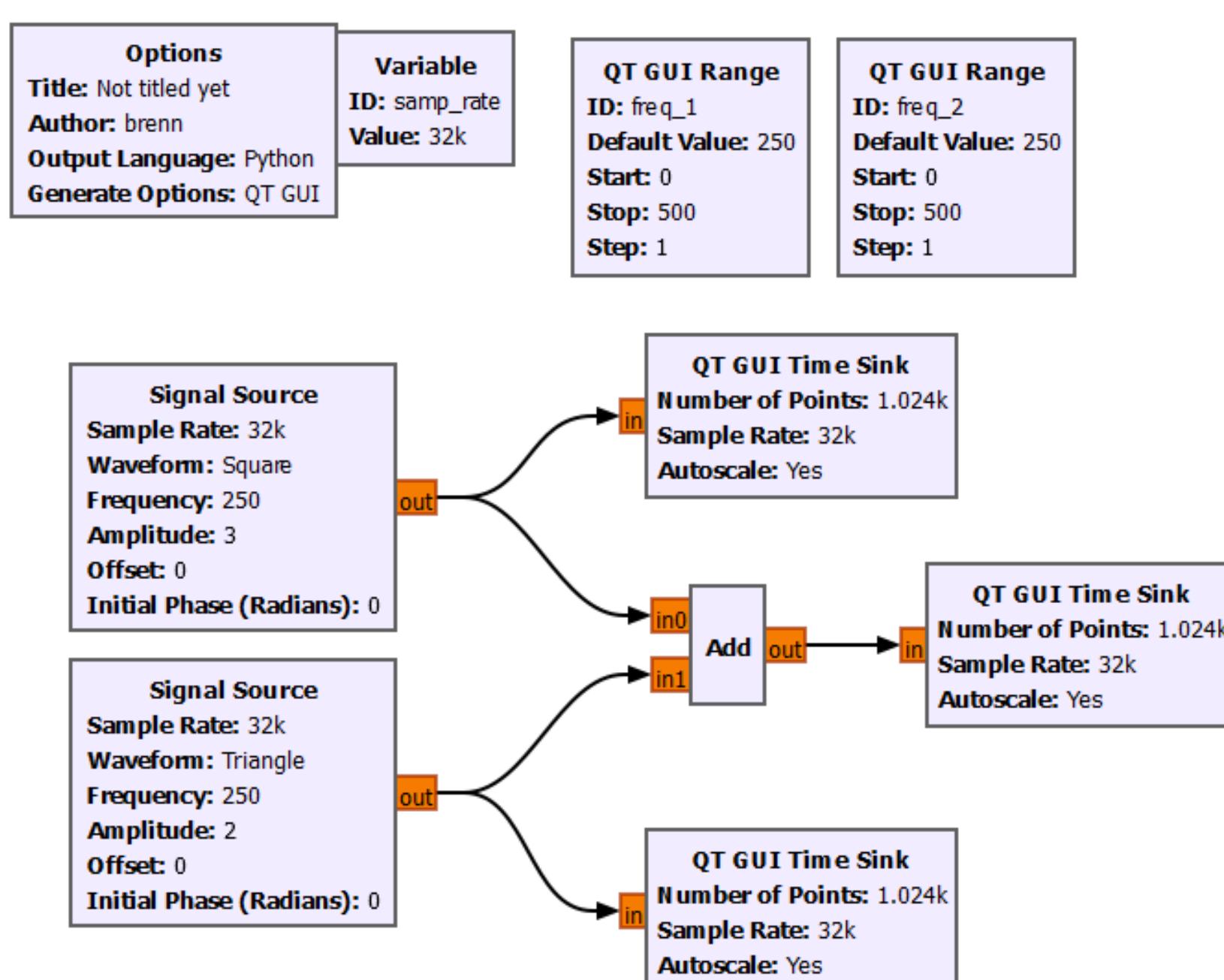


- Essa função recebe um vetor de amostras do sinal que desejamos transformar, e retorna a versão no domínio da frequência desse vetor.

Mas olha só, no nosso querido GNU Radio, essa função já é aplicada automaticamente ao utilizarmos o bloco para mostra(plotar) o sinal no domínio do tempo. Como veremos nos exemplos mais adiante.



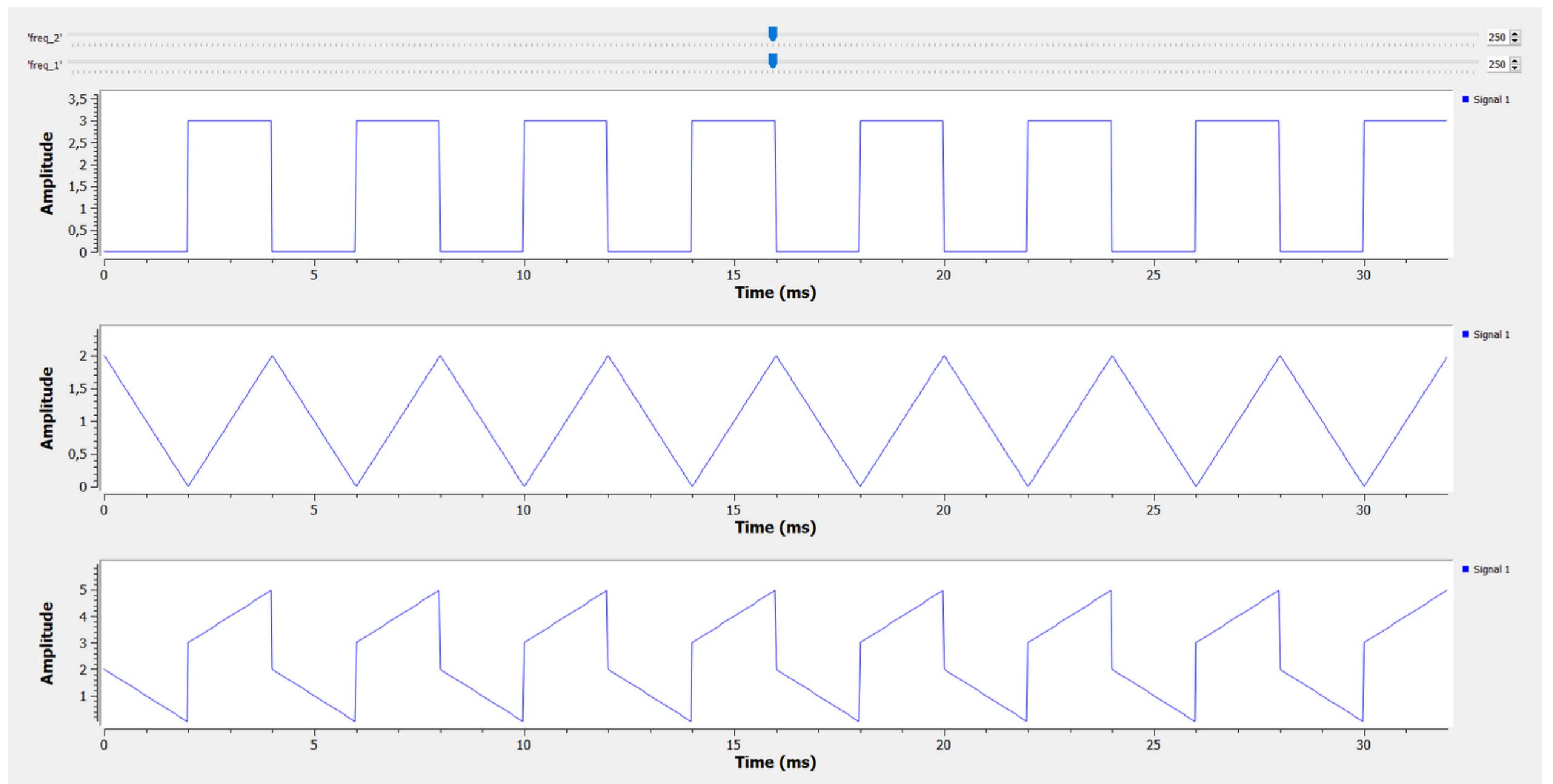
2.7 - Eba, vamos praticar !!! Vejamos o exemplo :



- Você consegue identificar o que este fluxograma está simulando?
Isso mesmo! Está representando uma operação de adição de sinais!!

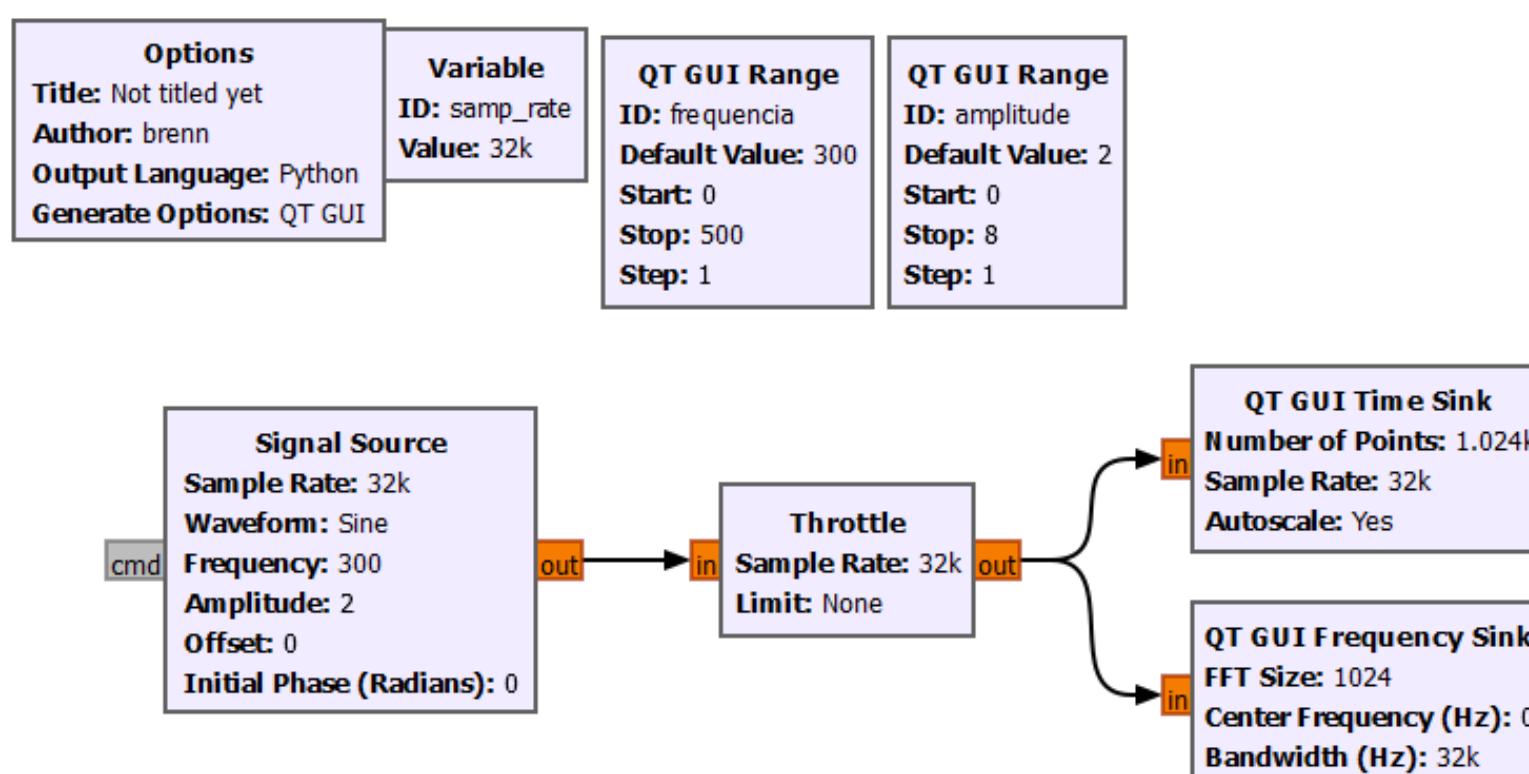
Vejamos o que temos de características:

- **Bloco QT GUI Range:** ele cria uma variável para nós que pode ser usada para modificarmos a frequência (ou qualquer outro atributo numérico) enquanto executamos o fluxograma;
- Temos duas fontes de sinais. Uma gerando uma onda quadrada e outro gerando uma onda triangular. Onde ambas estão sendo plotadas pelo **QT GUI Time Sink**, que é o responsável por gerar o nosso gráfico no domínio do tempo.
- **Bloco Add:** como o nome já sugere, ele se propõe a fazer a adição do sinal e retornar um resultado que também plotamos no domínio do tempo.



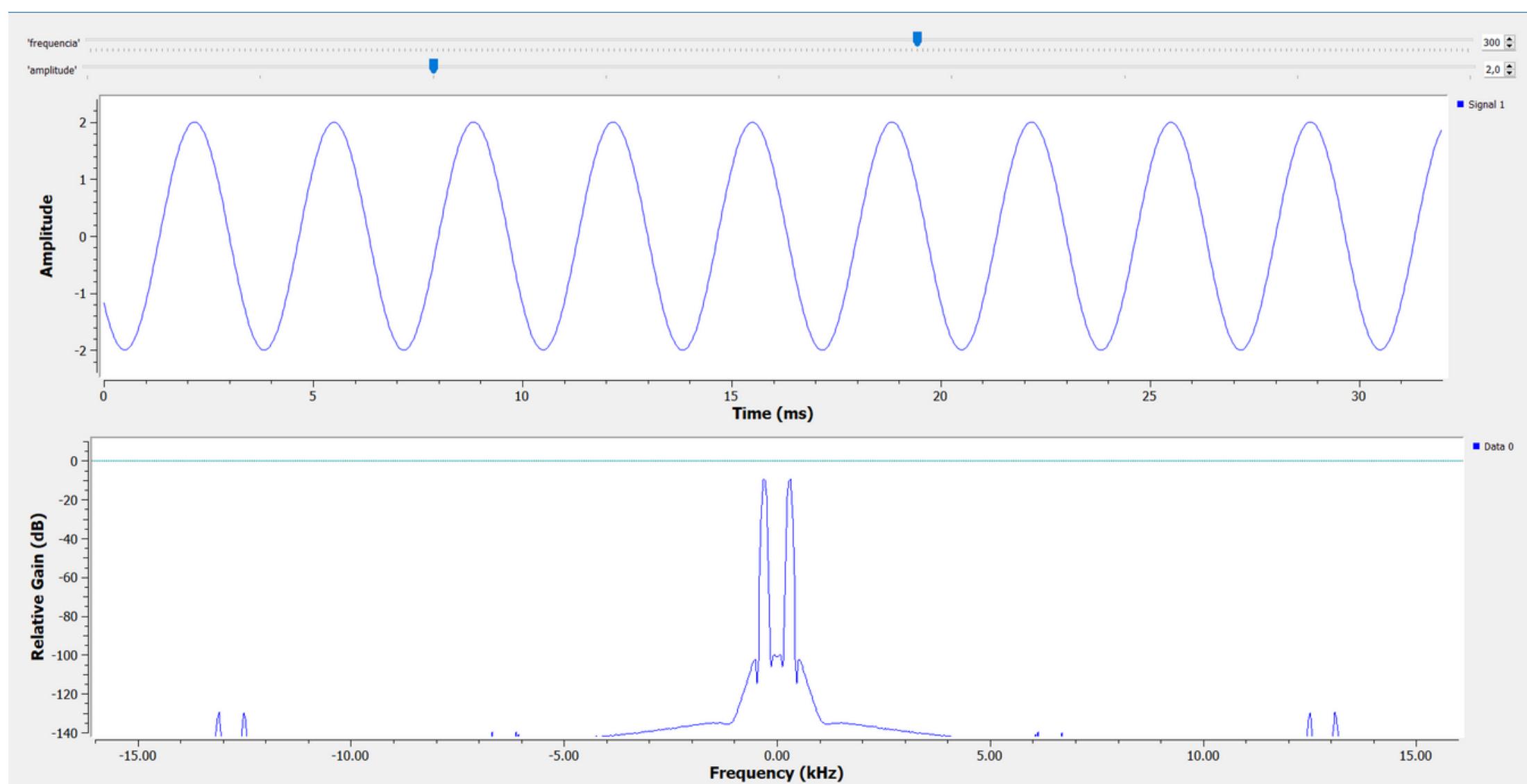
- Este é o resultado!
- Ainda nessa execução, podemos modificar as frequências dos dois sinais e ver o resultado de sua soma em tempo real. Legal né?!

Ainda continuando os exemplos deste tópico. Vejamos agora como é a visualização de um sinal no domínio do tempo e no domínio da frequência de forma simultânea.

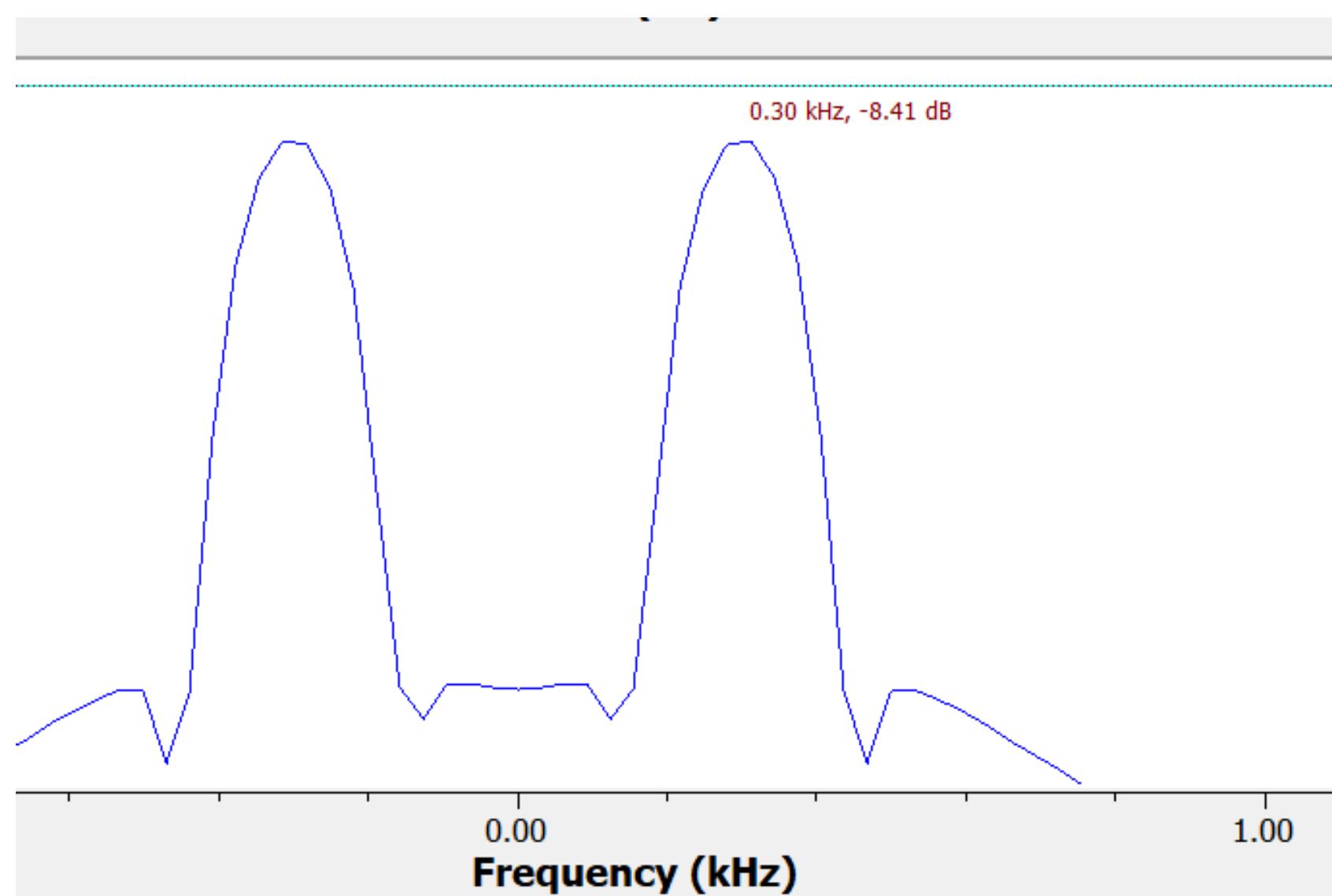


Características deste fluxograma:

- Bloco QT GUI Range: assim como no exemplo anterior, utilizados esse bloco para modificarmos a frequência do sinal e vermos o resultado ao vivo;
- Bloco Throttle: ele limita a quantidade de amostras para que não sobrecarregue o poder computacional do seu pc.



Ao colocarmos o mouse sob o pico da direita no domínio da frequência, observamos que seu valor é o que está definido na barra da frequência logo a cima.

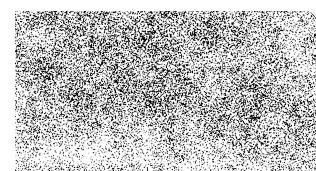


Como ela está definida em “300”, no gráfico apresentado pelo GNU Radio, é informado a informação em Hz, neste caso, 0.30 kHz.



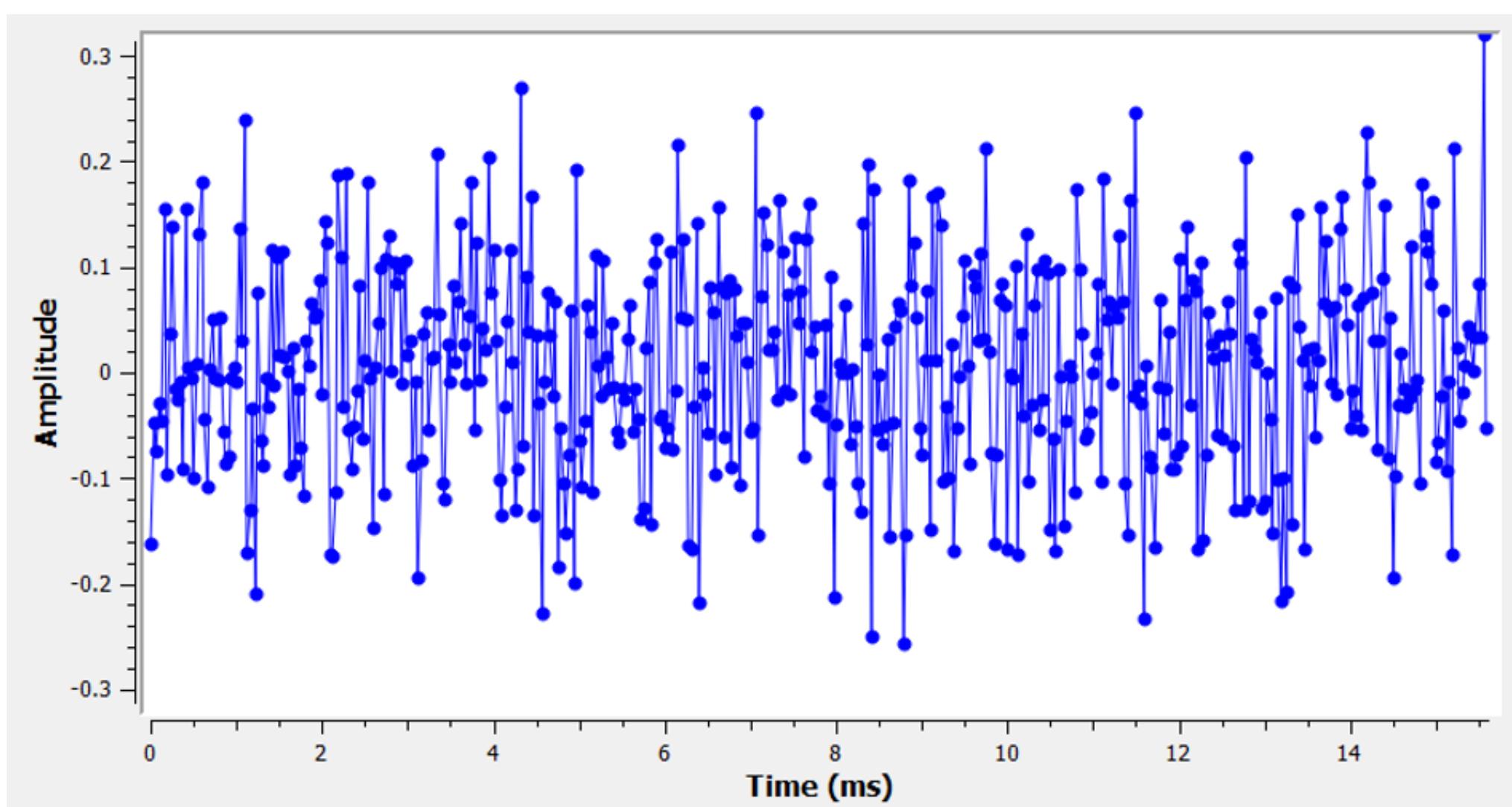
E la vamos nois, melhor que ver é fazer (<https://abre.ai/somasinais>);
(<https://abre.ai/multiplicasinais>);
(<https://abre.ai/domfreq>);

3 - Ruído:



Pois bem, vamos iniciar uma nova fase para desvendarmos um pouco sobre ruídos em telecomunicações. Podemos classificar como qualquer interferência indesejada ou sinal não desejado que prejudique a qualidade da comunicação. O ruído pode distorcer informações, reduzir a taxa de transferência de dados e afetar a qualidade da comunicação, exigindo técnicas de mitigação e correção de erros em sistemas de telecomunicações.

Aqui podemos ver um sinal com ruído :



3.1 - Classificação dos tipos de ruídos :

Em geral possuímos dois tipos, os internos e externos :

Ruido Interno : É gerado quando a fonte pertence ao sistema de telecomunicações, normalmente é advindo de componentes internos como cabos, amplificadores, circuitos chaveados e afins ;

Ruido Externo : Como sugere o nome, esse tipo de ruído é gerado por meios externos, podemos compreender melhor a partir de exemplos de geradores, na qual podemos citar ruído cósmico de fundo, ruído das estrelas (incluindo o Sol), eletricidade estática, raios, ignição de motores ;

- Há, podemos também classifica-los em relação ao tempo, que pode ser denominado **continuo** quando o ruído esta sempre presente no sistema. Ou **impulsivo**, que ocorre em forma de manifestações repentinas, basicamente constituindo-se em impulsos discretos que interferem no sinal transmitido.

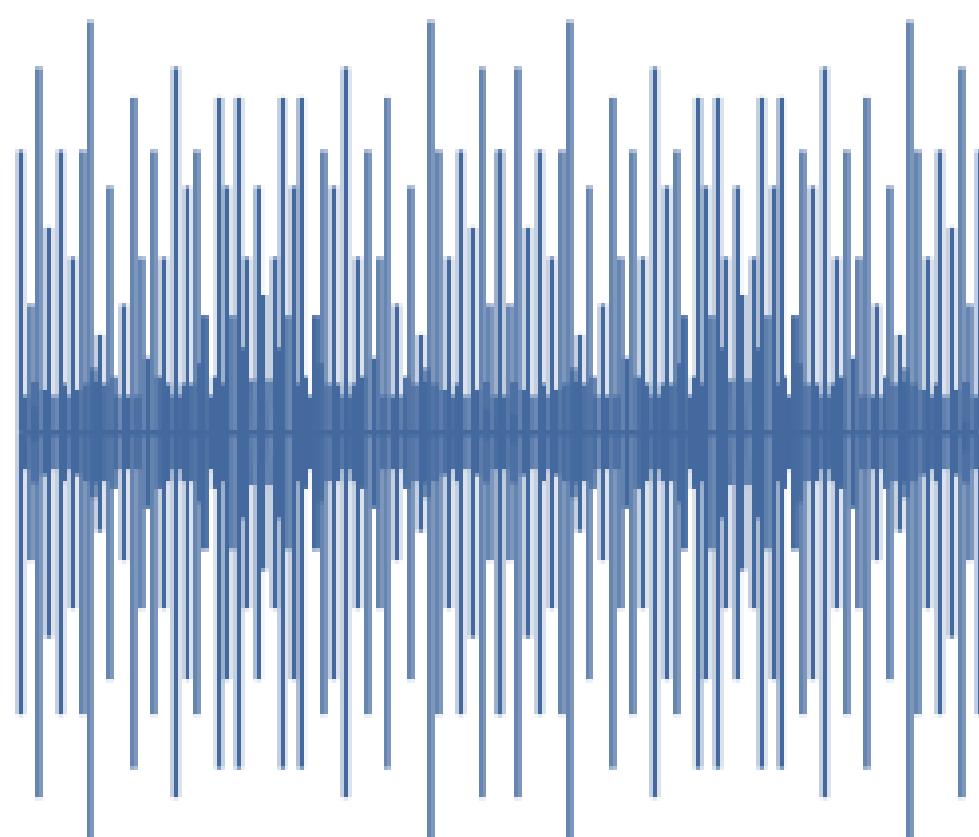
Já em relação a frequência, temos o **ruído branco e colorido**, vejamos um pouco sobre eles :

- **Branco** : Apresenta componentes em todo o espectro de frequência, apesar de na prática não existir essa características completamente, sendo um conceito mais utilizados em estudos ;
-Exemplo: Na prática encontramos esse ruído em estudos, normalmente para simular sons da natureza como é o caso do AWGN(Ruido Branco Gaussiano Ativo);

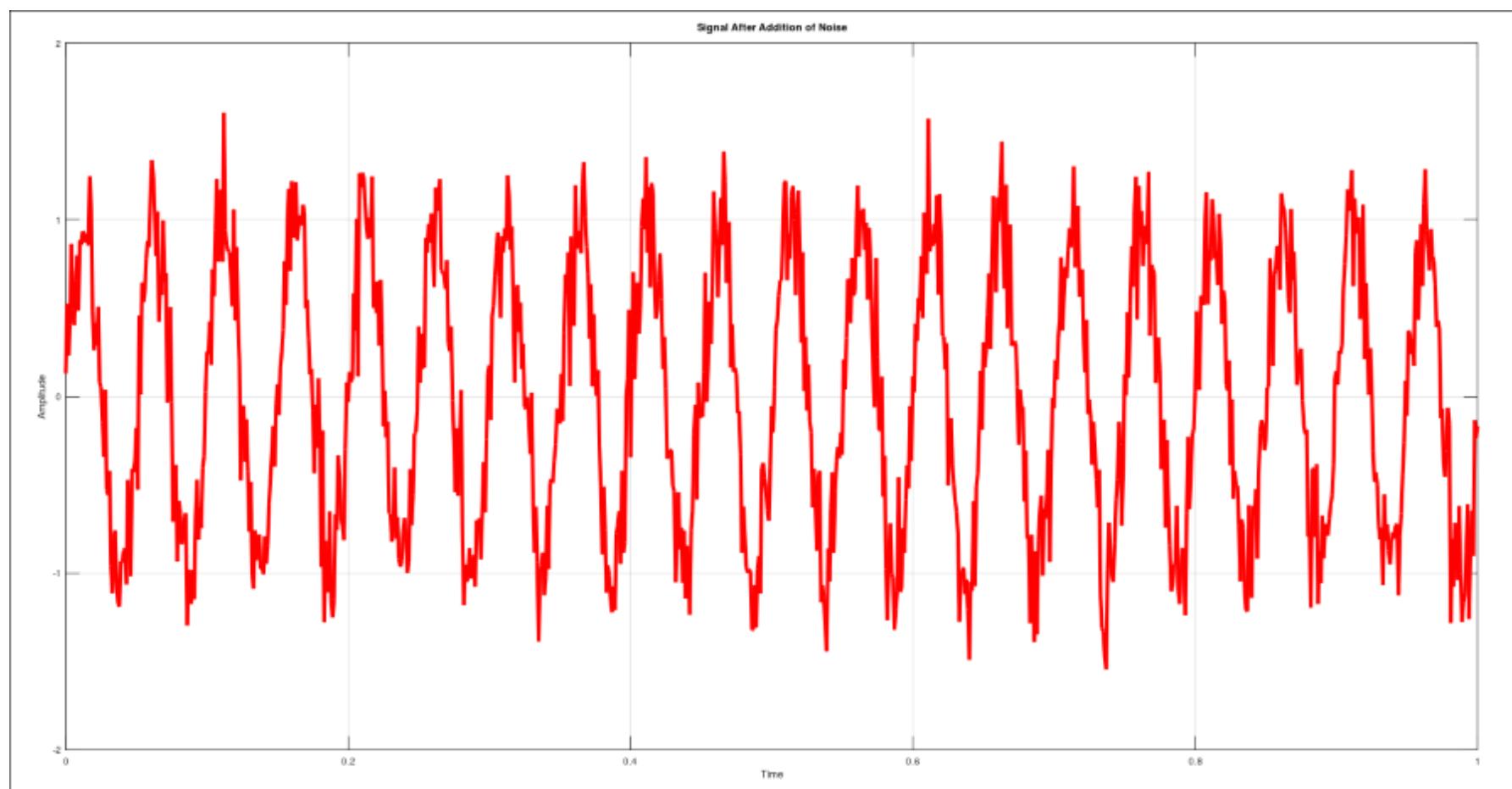
- **Colorido** : Apresenta componentes freqüenciais somente em uma ou algumas freqüências.

3.2 - Alguns tipos de ruídos mais encontrados :

Ruído Térmico : É produzido pelo aquecimento dos condutores e semicondutores do próprio sistema de telecomunicações. A agitação dos elétrons devido ao aquecimento gera sinais elétricos aleatórios nesses sistemas.



AWGN (Ruido Branco Gaussiano) : O ruído gaussiano é amplamente utilizado em engenharia, pesquisa e comunicações para modelar e simular situações. É importante para testes estatísticos, análises de sistemas de comunicação e avaliação de desempenho na presença de interferências e perturbações. Em essência, o ruído gaussiano é uma forma de introduzir aleatoriedade controlada em estudos e experimentos, o que o torna uma ferramenta valiosa. Suas principais características são a amplitude constante e distribuição simétrica, se parece com o exemplo abaixo :

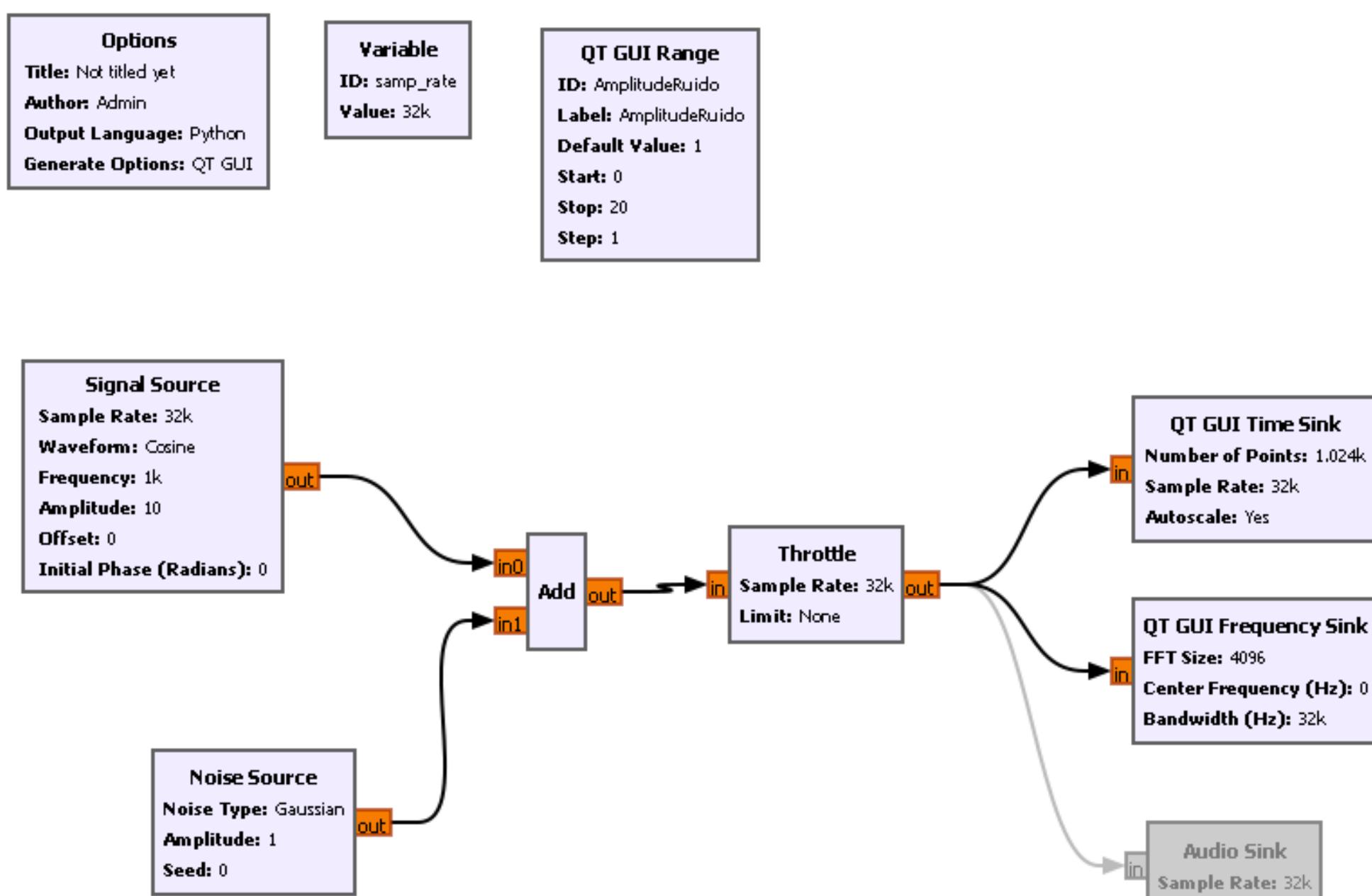


Apesar de termos esses e outros tipos de ruídos, o mais comum quando trabalhamos na prática, é o térmico. Ele está sempre presente em qualquer sistema eletrônico e afeta principalmente comunicações de banda larga, como rádio, TV e comunicações de dados.

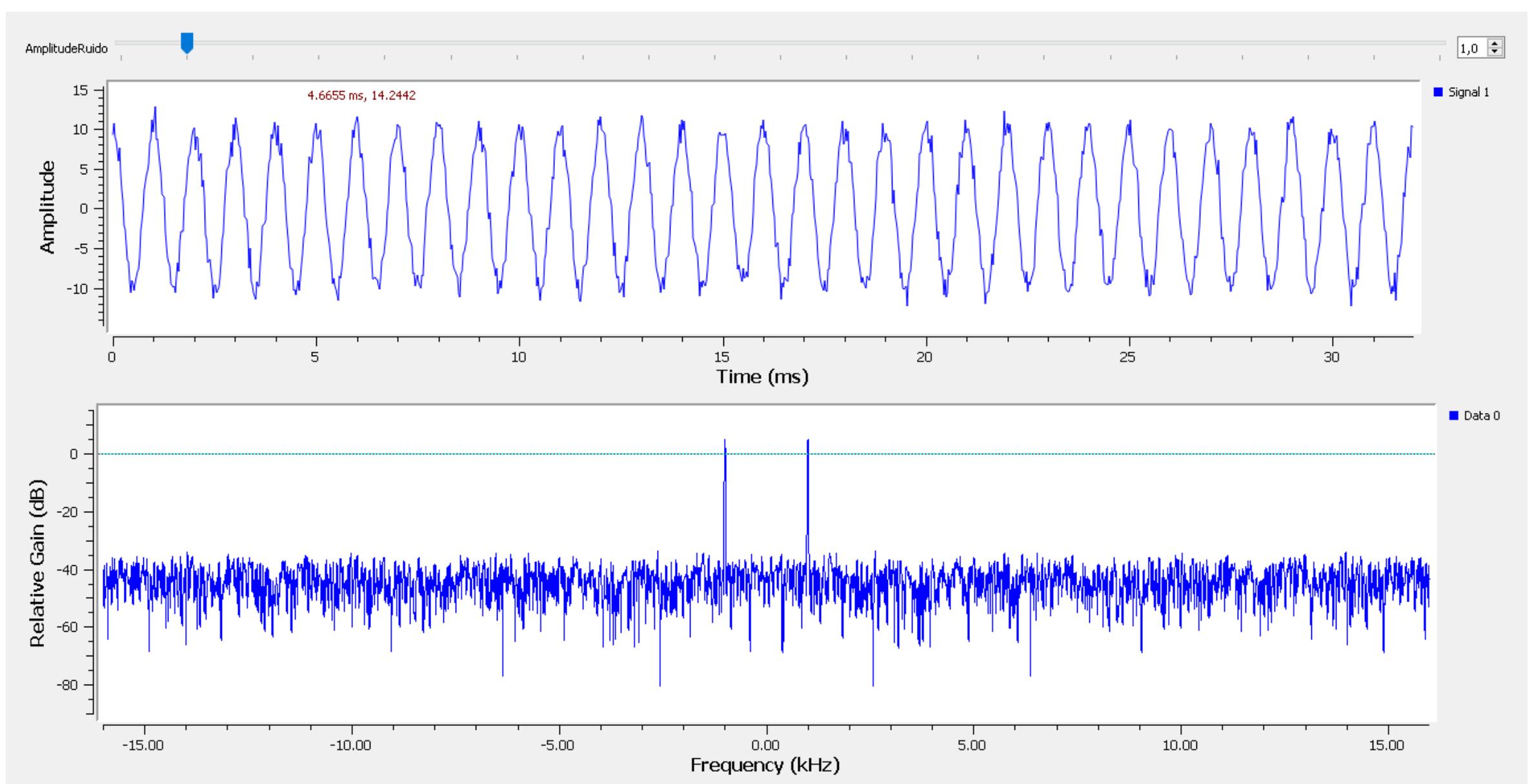
3.3 - Tratamento dos ruídos :

Apesar de não conseguirmos eliminar os ruídos completamente, conseguimos reduzi-los drasticamente por meio de algumas técnicas como amplificação de sinal, filtragem, códigos de correção de erro, modulação adaptativa, cancelamento de ruídos e afins.

3.4 - Partiu praticar !!! Veja o ruido em ação :



Veja só como é legal, entramos com um sinal cossenoide e adicionamos a ele um ruído gaussiano (pode ser alterado), por fim utilizamos os blocos QT para visualizar em relação ao tempo e frequência. Também é possível alterar a amplitude do ruído para visualizar sua influência no sinal, vejamos:



→ E lá vamos nós, melhor que ver é fazer (<https://abre.ai/naturezaruido>);

4 - Filtros:

Bom caro leitor, agora passaremos para mais assuntos abordados em estudos de sinais. No nosso próximo tópico entraremos em noções básicas de filtros.

Filtros são utilizados em muitas disciplinas. Como por exemplo em processamento de imagens, em que se faz o uso intenso de filtros 2D.

Seu funcionamento é exatamente como você está pensando nesse exato momento. É basicamente a mesma ideia de você utilizar um filtro todas as manhãs para fazer seu café, que filtra os sólidos dos líquidos.

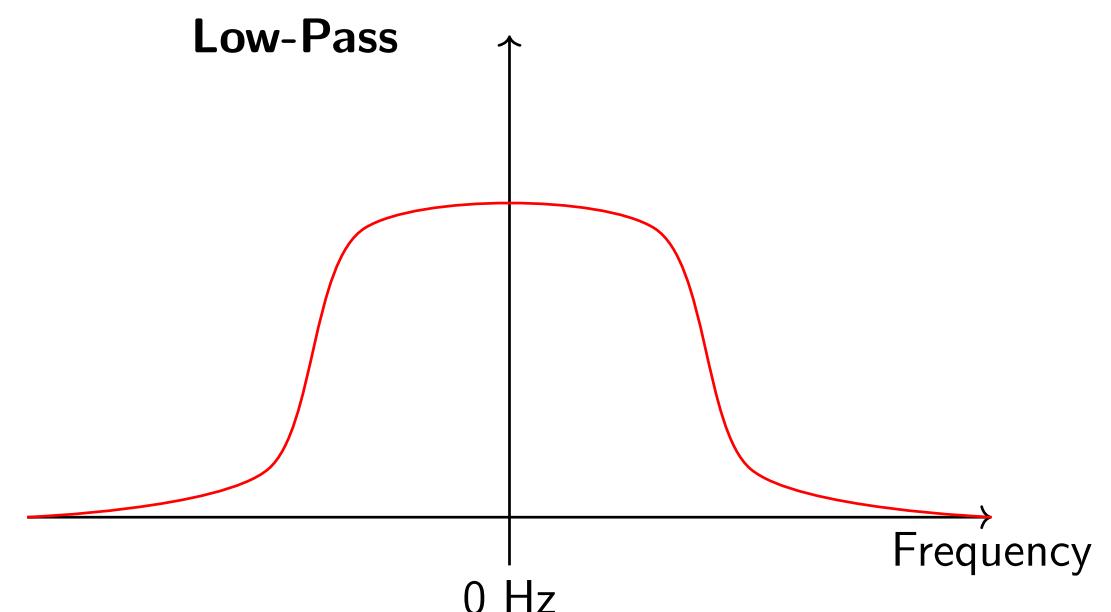
Em processamento digital de sinais, os filtros são principalmente utilizados para:

- Separação de sinais que foram combinados;
- Remoção de excesso de interferência/ruído após receber um sinal;
- Restauração de sinais que foram distorcidos de alguma forma;

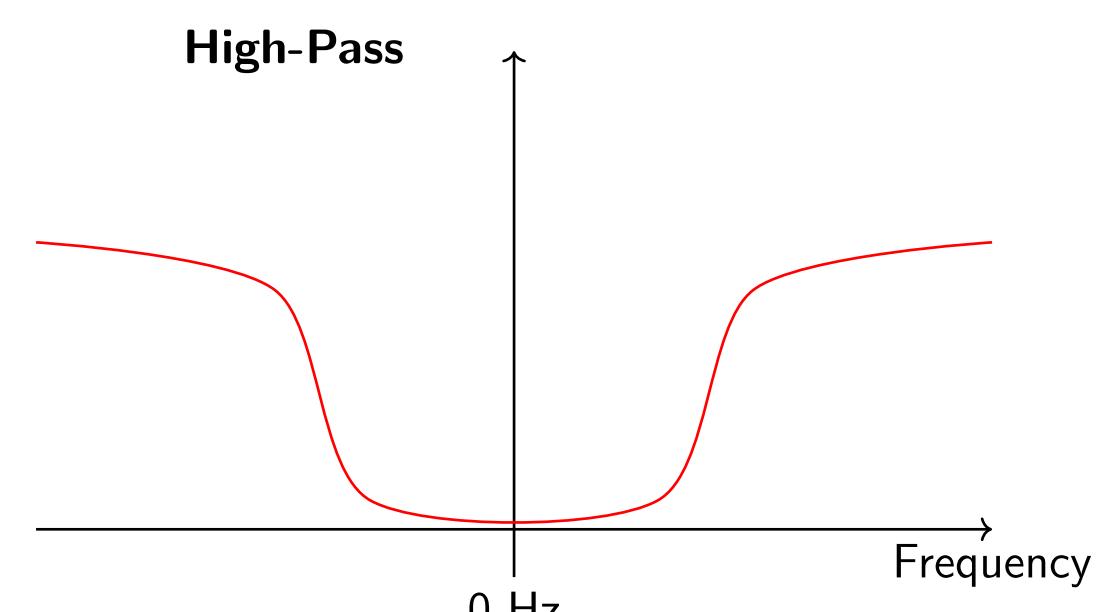


E para isso, temos alguns tipos de filtros, que se adequam melhor dependendo da situação:

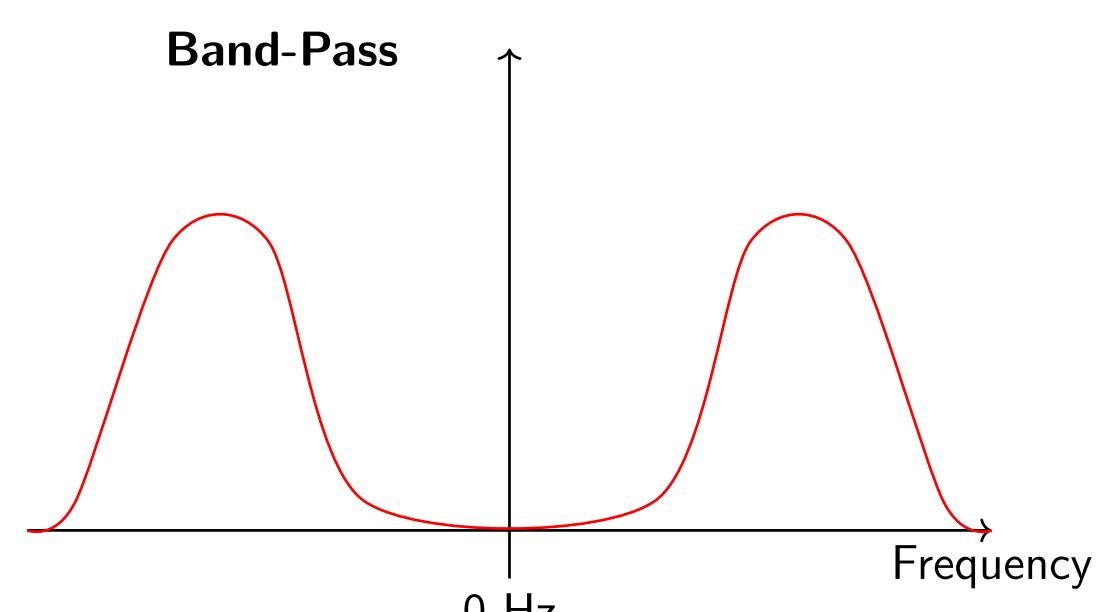
- **Passa-baixa:** Permite a passagem de frequências abaixo de uma determinada frequência de corte, atenuando as partes do sinal recebido acima desse corte.



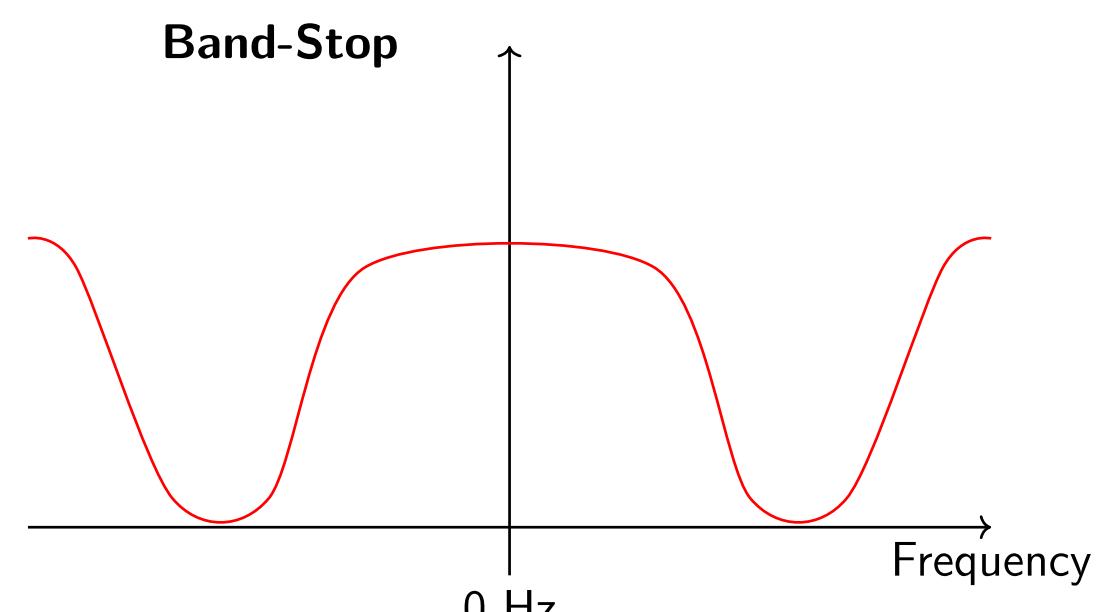
- **Passa-alta:** Aceita a passagem de frequências acima de uma determinada frequência de corte, atenuando parte do sinal abaixo desse corte.



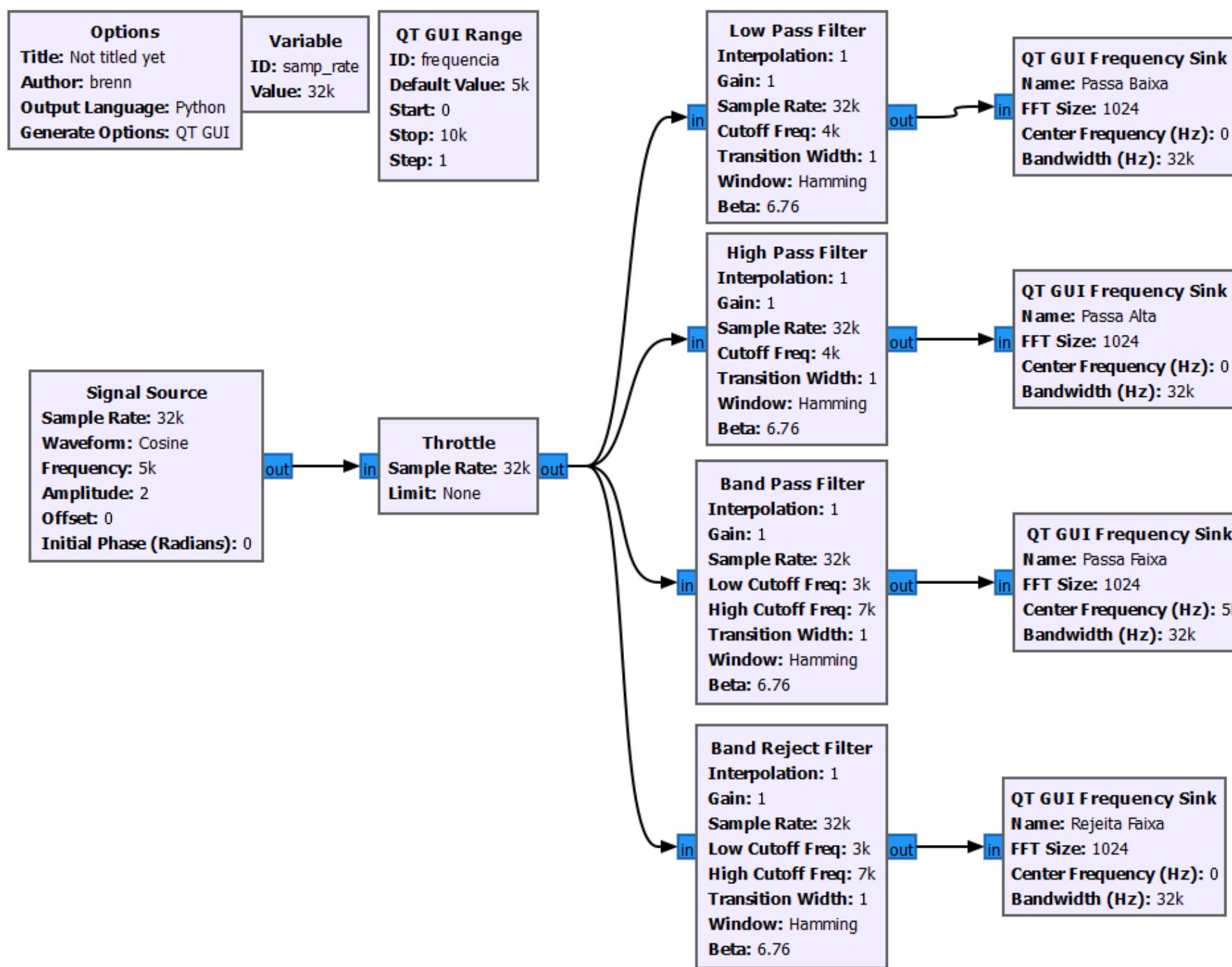
- **Passa-faixa:** Permite a passagem de frequências em uma faixa de frequências específica, atenuando a parte do sinal fora dessa faixa.



- **Rejeita-faixa:** Atenuam as frequências em uma faixa de frequências específica, na qual permite a passagem do sinal fora dessa faixa.

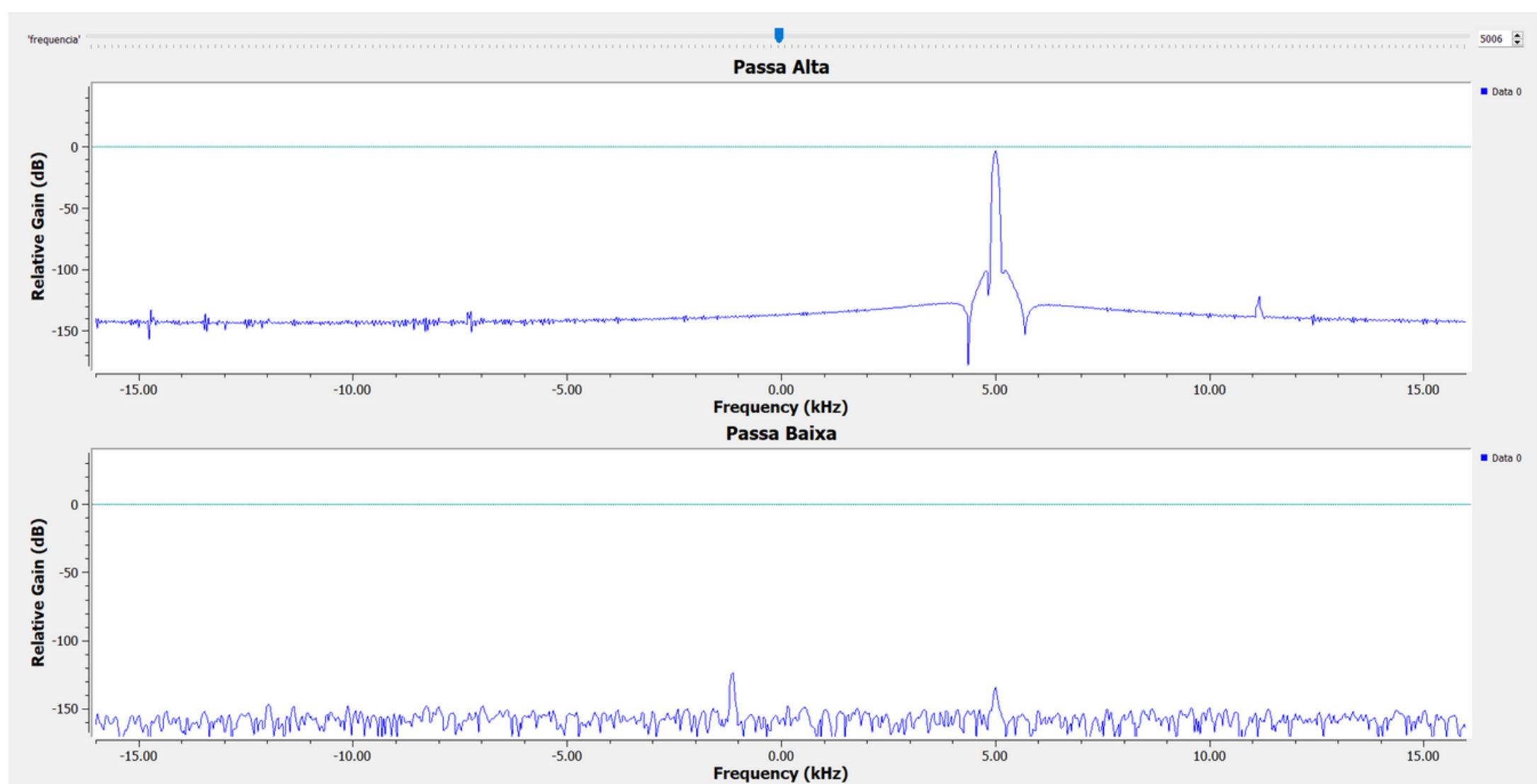


- Para você entender melhor, a segue o fluxograma:

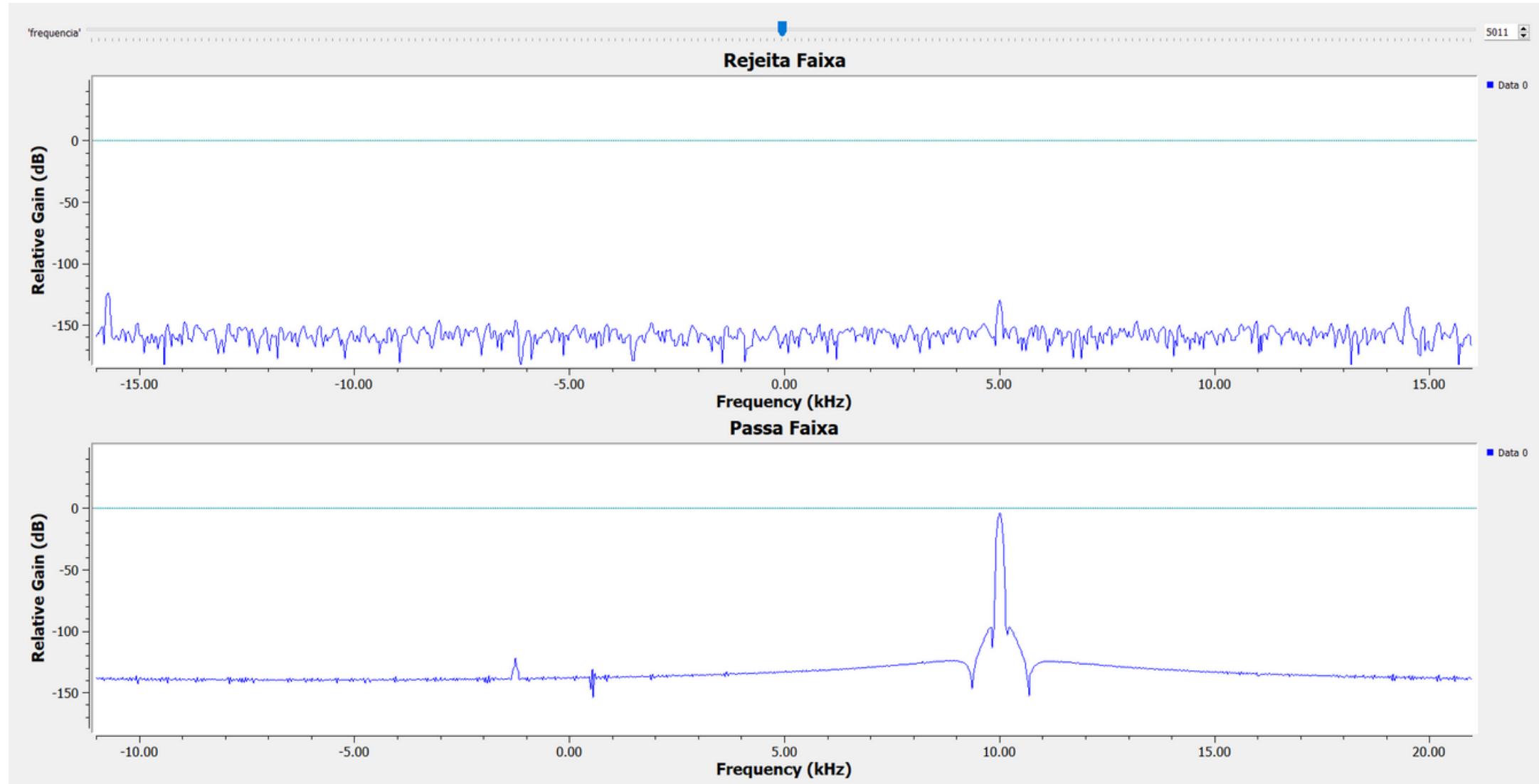


Detalhe : Assim, temos um mesmo sinal, nesse caso uma cossenoide, que passa pelos 4 tipos de filtros apresentados anteriormente.

Como a frequência inicialmente está definida em 5k, isso é o que acontece com o sinal ao ser filtrado.

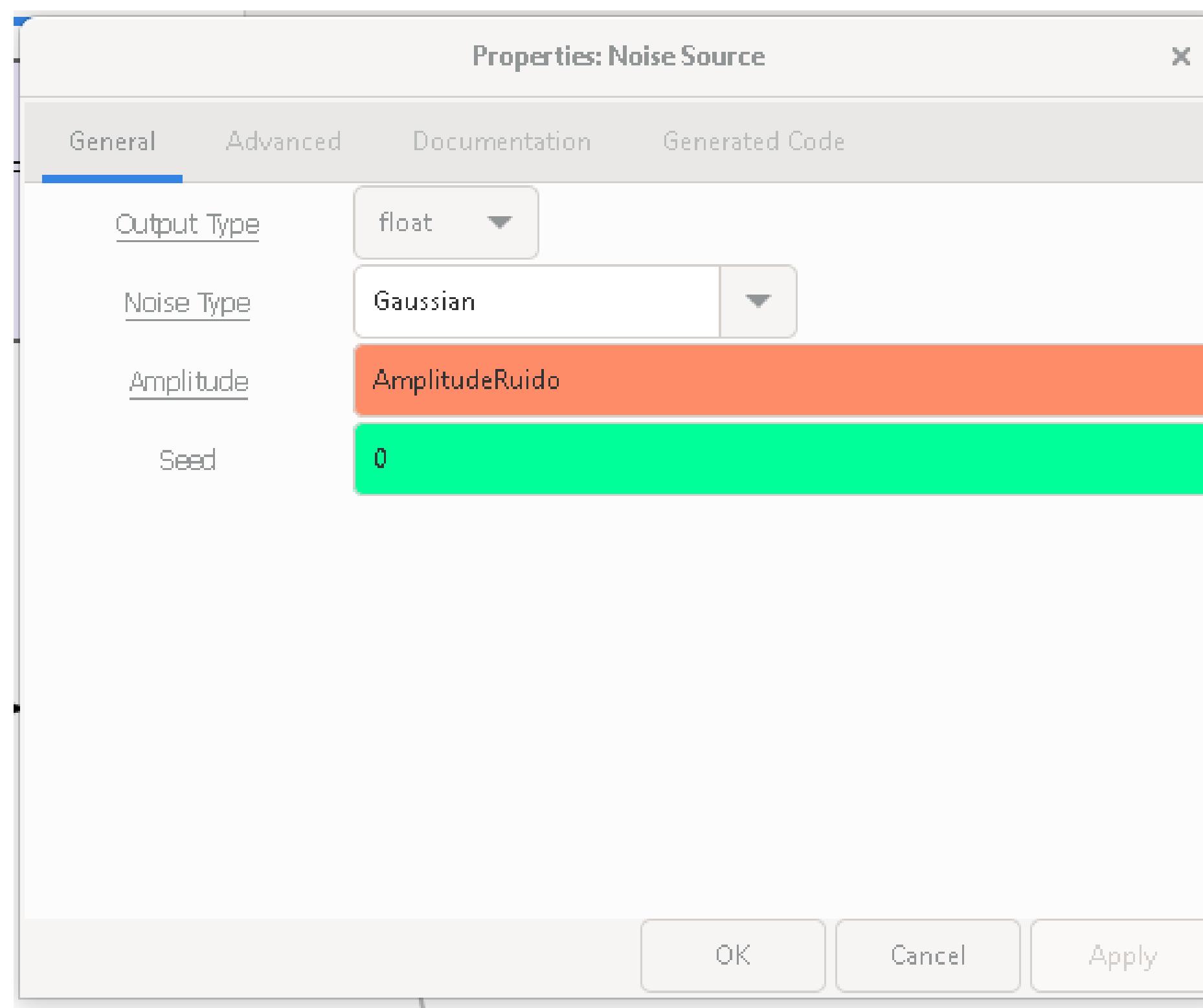


Filtros passa-alta e passa-baixa, ambos com frequência de corte de 4k. Passa-alta não atenua o sinal, pois está acima do corte. Já o filtro passa-baixa atenua o sinal, devido sua frequência não ser menor que o valor definido.



Agora temos os filtros Rejeita-faixa e passa-faixa, cada um com faixa de corte de 3k à 7k. E como nosso sinal está em 5k, observamos pela simulação que esse sinal de frequência de 5k está sendo atenuado pelo filtro que rejeita essa faixa de frequência, enquanto o outro está “deixando passar”.

- Por fim, você pode fazer algumas alterações no tipo de ruído através do bloquinho noise, assim visualizara o comportamento de cada um :



E lá vamos nós, melhor que ver é fazer (<https://abre.ai/filtrosinais>);

5 - Modulação: ({{}})

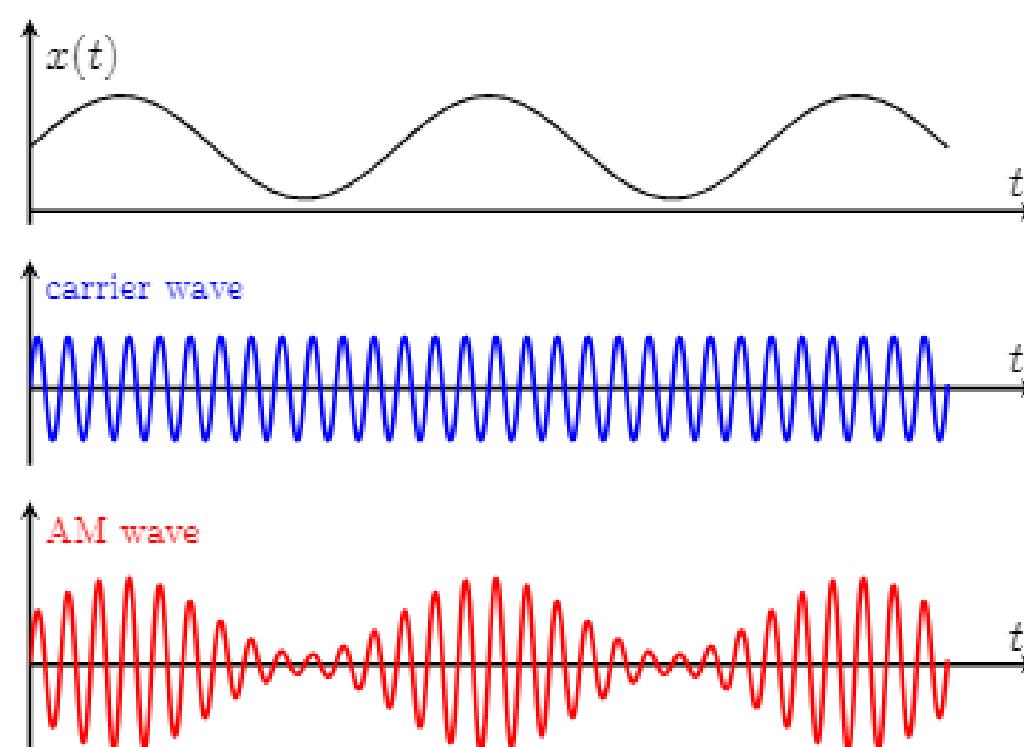
O processo pelo qual dados/informações são convertidos em sinais elétricos/digitais para transferir esse sinal através de um meio é chamado de modulação. Ele aumenta a força para alcance máximo dos sinais. O processo de extração de informações/dados do sinal transmitido é denominado demodulação. Um Modem é um dispositivo que executa processos de modulação e demodulação. As diversas formas de modulação são projetadas para alterar as características das ondas portadoras. As características de modulação mais comumente alteradas incluem amplitude, frequência e fase.

Sinal portadora: Os sinais que não contêm informações, mas possuem uma determinada fase, frequência e amplitude, são chamados de sinais portadores.

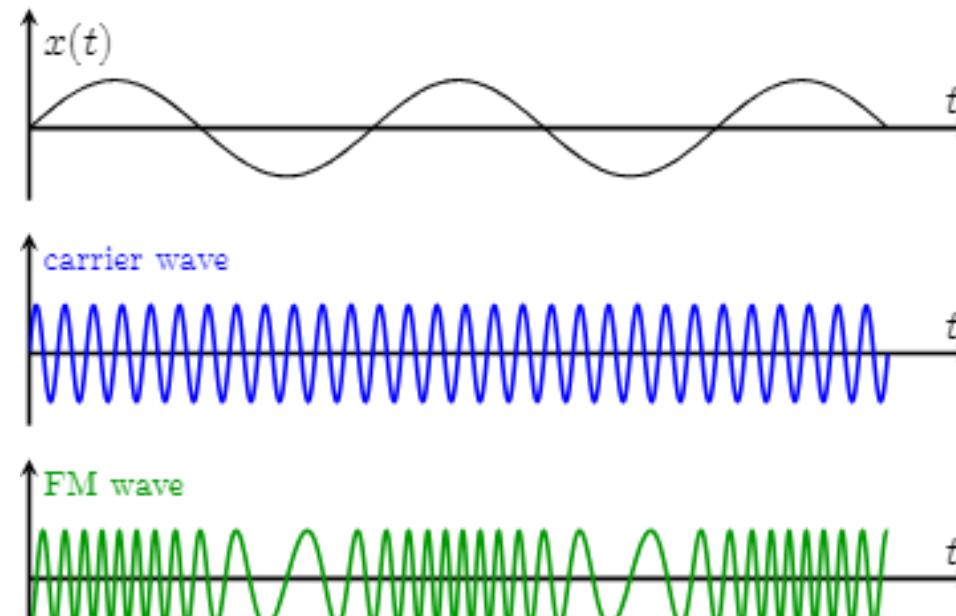
Sinais modulado: Os sinais que são a combinação dos sinais da portadora e dos sinais de modulação são sinais modulados. O sinal modulado é obtido após a modulação dos sinais.

5.1 - Tipos de modulação:

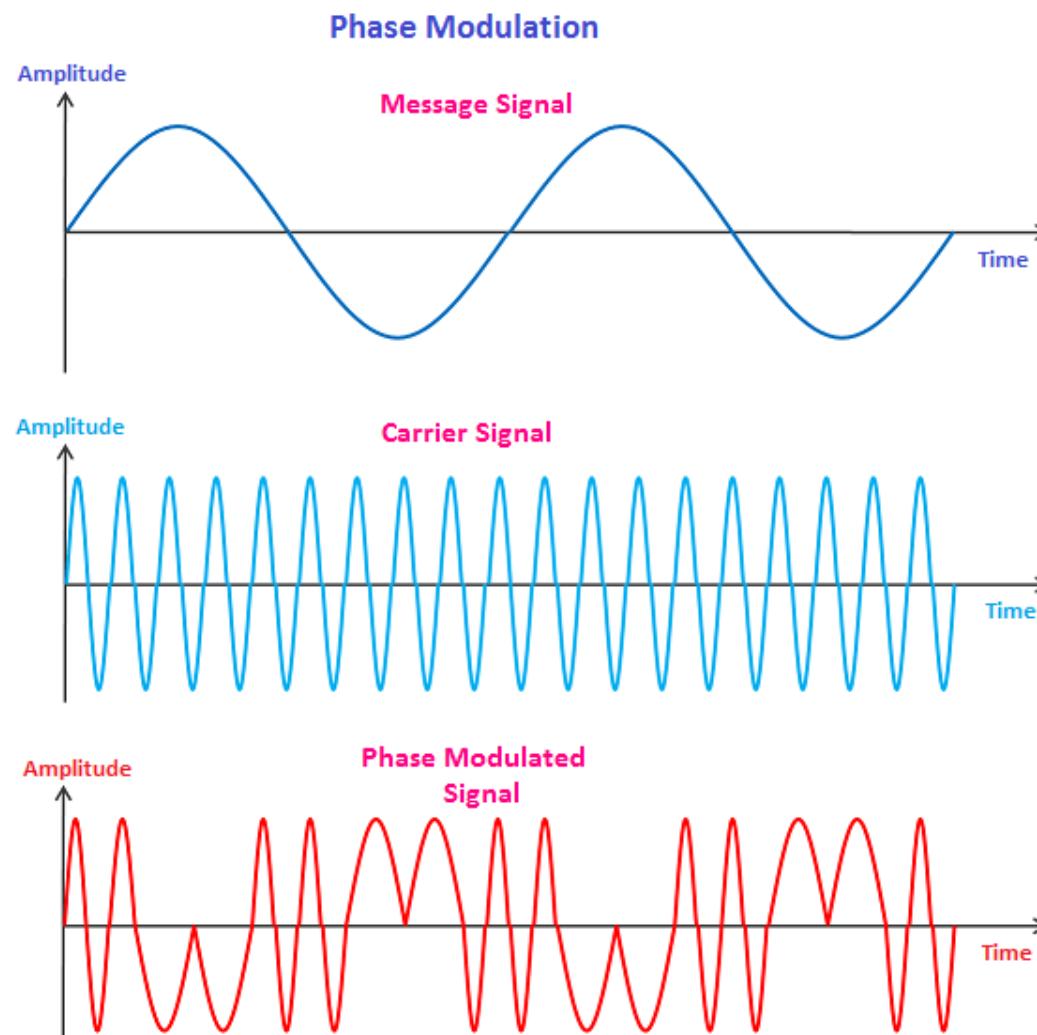
- **Modulação de amplitude:** É um tipo de modulação em que apenas a amplitude do sinal portador é variada para representar os dados que estão sendo adicionados aos sinais, enquanto a fase e a frequência do sinal são mantidas inalteradas.



- **Modulação de frequência:** É um tipo de modulação em que apenas a frequência do sinal portador é variada para representar a frequência dos dados enquanto a fase e a amplitude dos sinais são mantidas inalteradas.

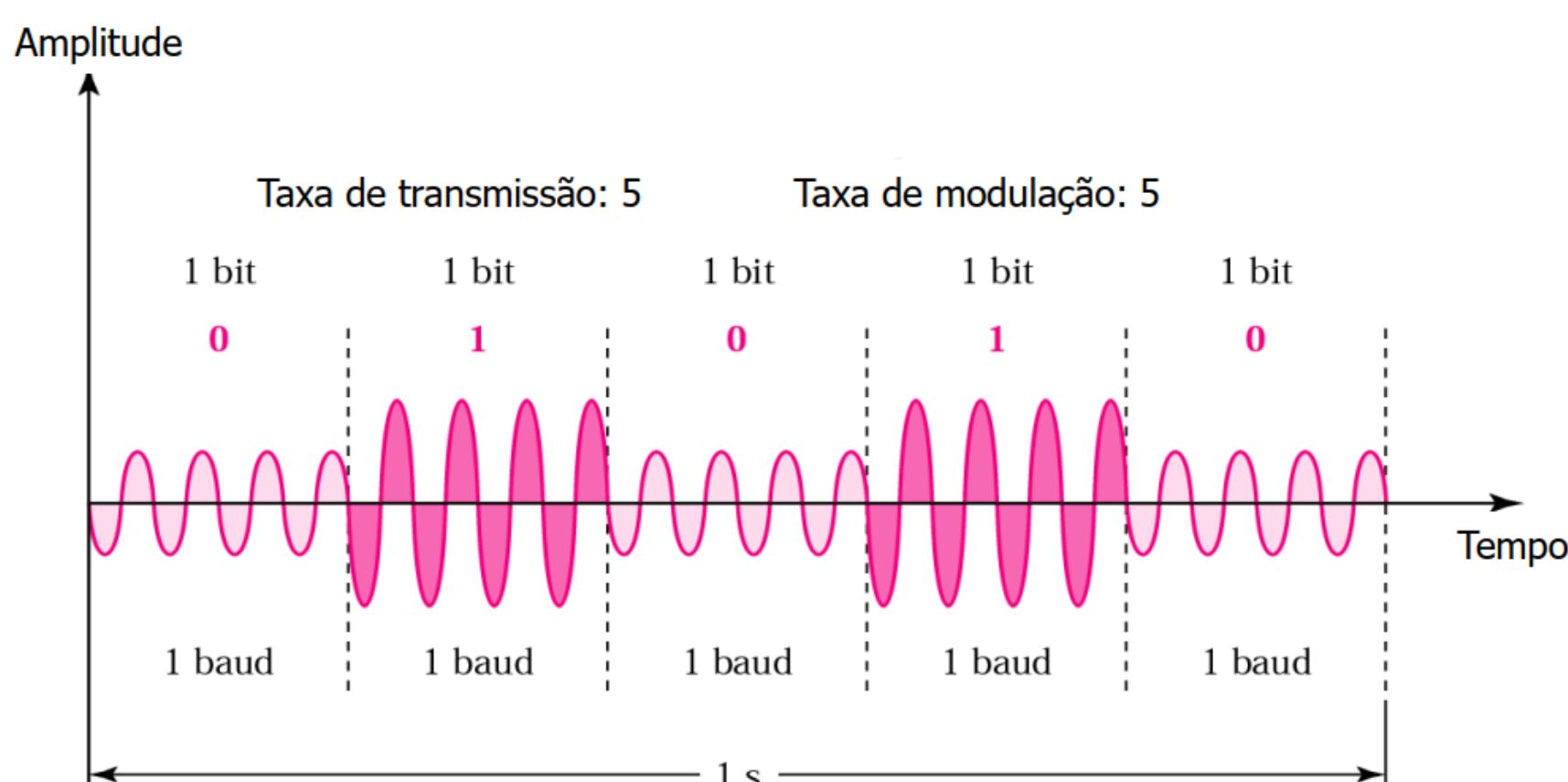


- **Modulação de fase:** É um tipo de modulação em que a fase do sinal da portadora é variada para representar os dados que estão sendo adicionados ao sinal. Diferentes valores de informação são representados por diferentes fases. Por exemplo: '1' pode ser representado por 0° enquanto '0' por 180° .

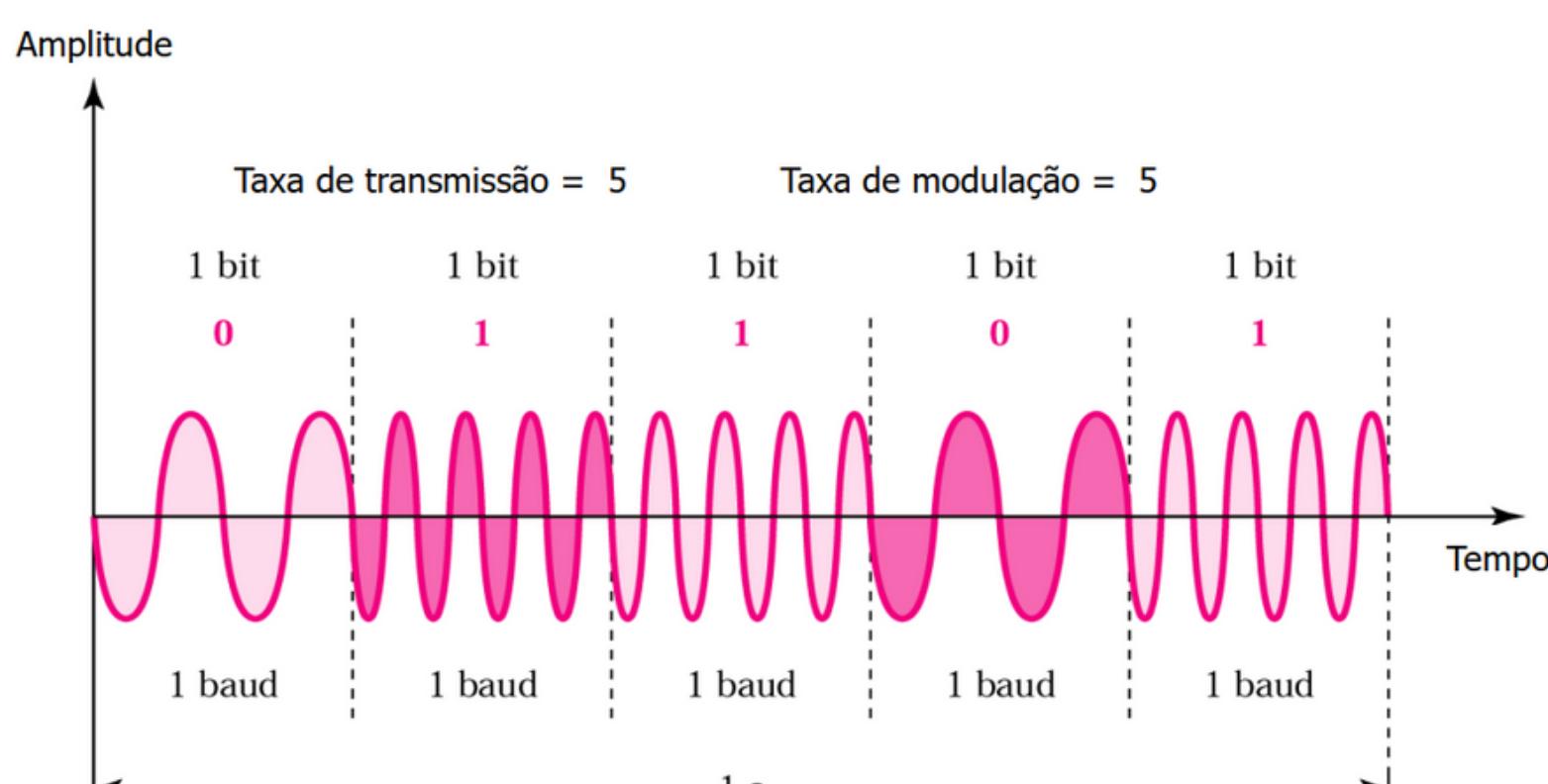


- **Modulação digital com portadoras:**

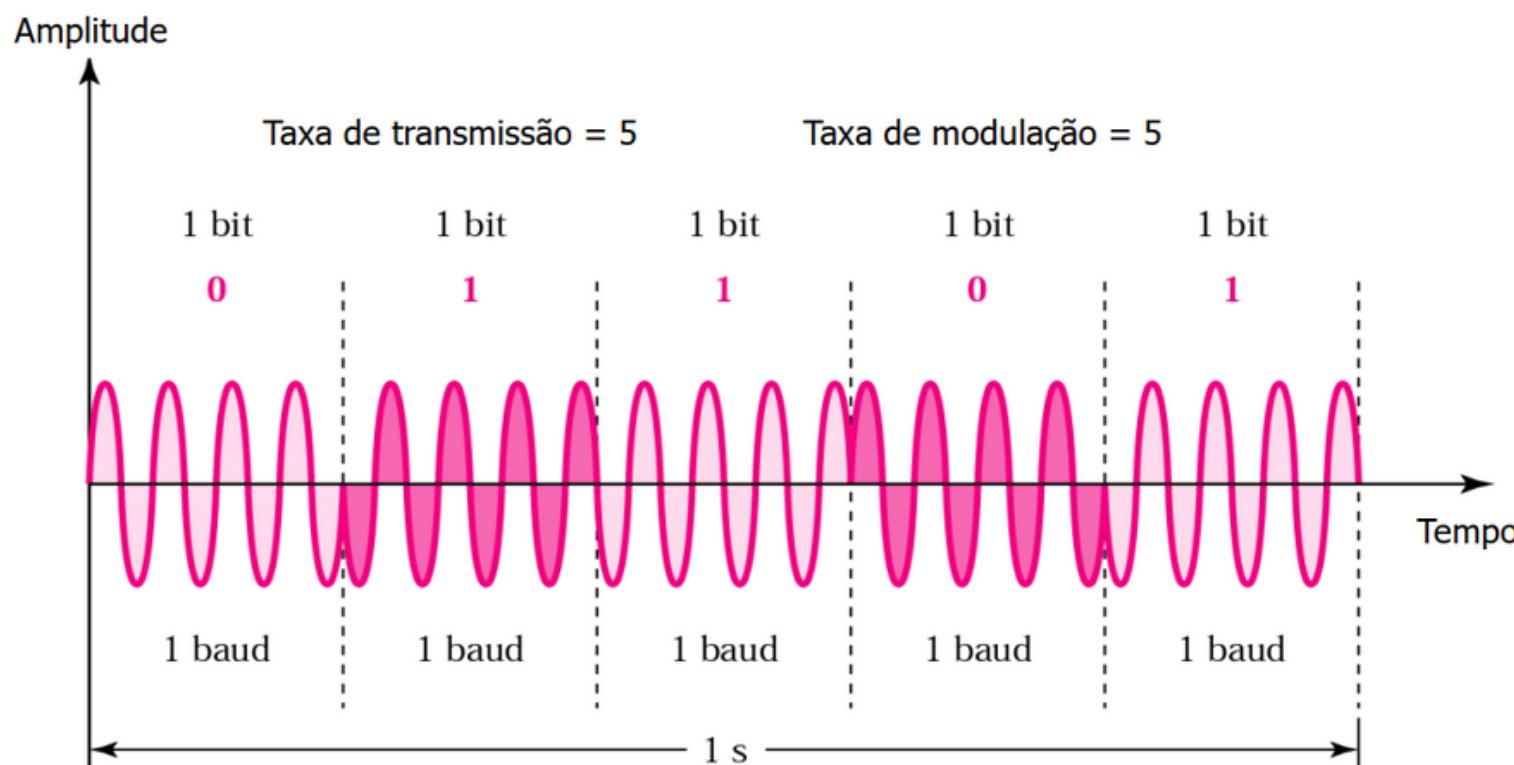
- **ASK - Amplitude Shift Keying :** A técnica ASK é o parente digital da modulação de amplitude (AM). Num sinal ASK é a amplitude de uma portadora que varia no tempo de acordo com os bits a transmitir.



- **FSK - Frequency Shift Keying :** A técnica FSK é o parente digital da modulação de frequência (FM). Num sinal FSK é a frequência de uma portadora que varia no tempo de acordo com os bits a transmitir.

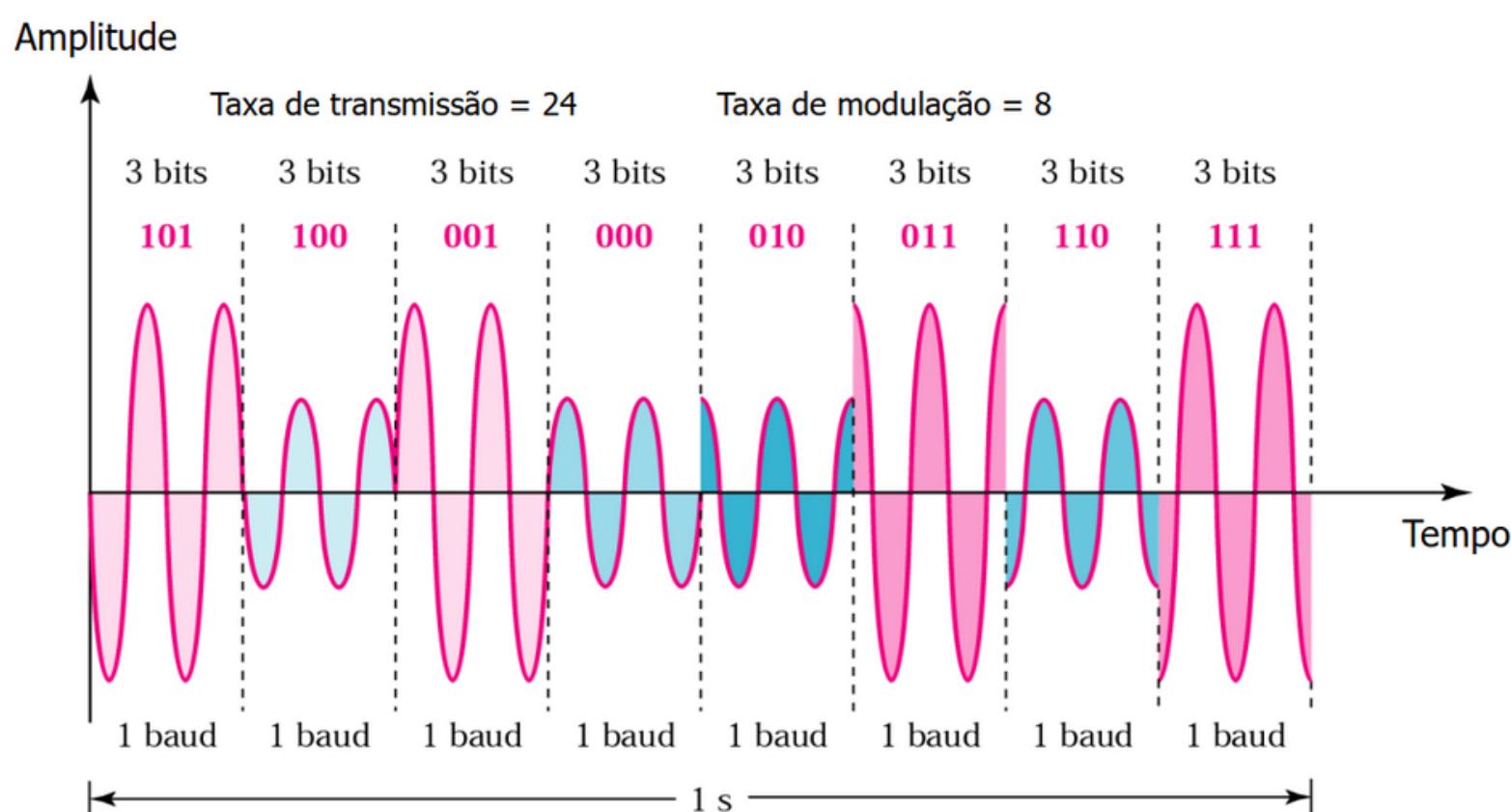


- **PSK - Phase Shift Keying:** Na técnica PSK o sinal transmitido transporta os dados digitais através da variação temporal da fase de uma portadora de acordo com os bits a transmitir.



- **QAM - Modulação por amplitude de quadratura:**

- Essa técnica é uma combinação das técnicas ASK e PSK elaborada de maneira a aumentar o número de bits transmitidos (bit, dígit, tríbit, etc.) para uma dada taxa de modulação .



- O que você acha de entendermos um pouco mais de modulação em amplitude? Cuidado que esse assunto é muito interessante e importante.

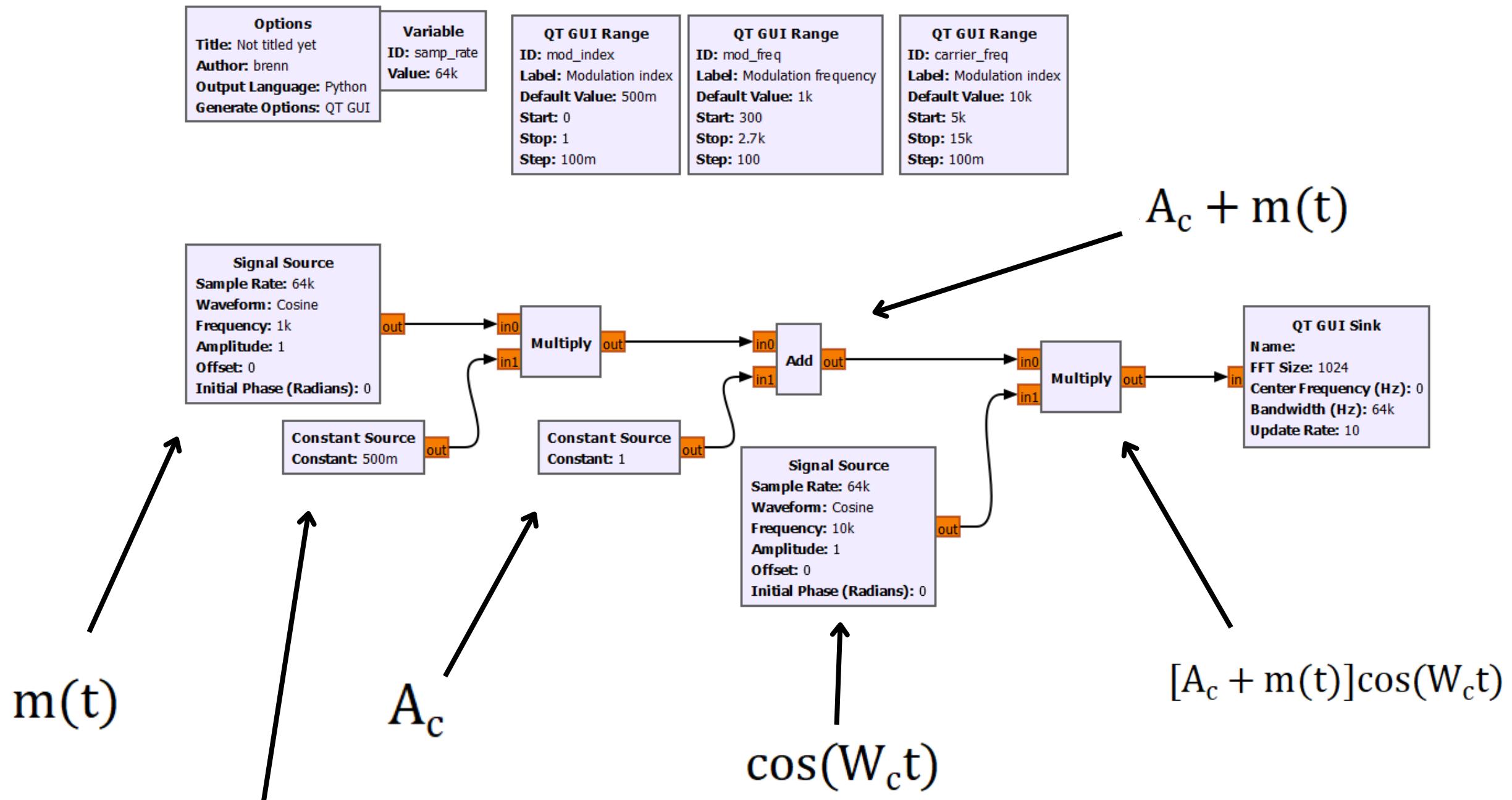
Para entendermos o próximo fluxograma, é preciso se atentar a seguinte equação:

$$[A_c + m(t)]\cos(W_c t)$$

Onde:

- A_c -> é o valor da amplitude do sinal modulante
- $m(t)$ -> é o sinal que será modulado
- $\cos(W_c t)$ -> é o sinal portadora

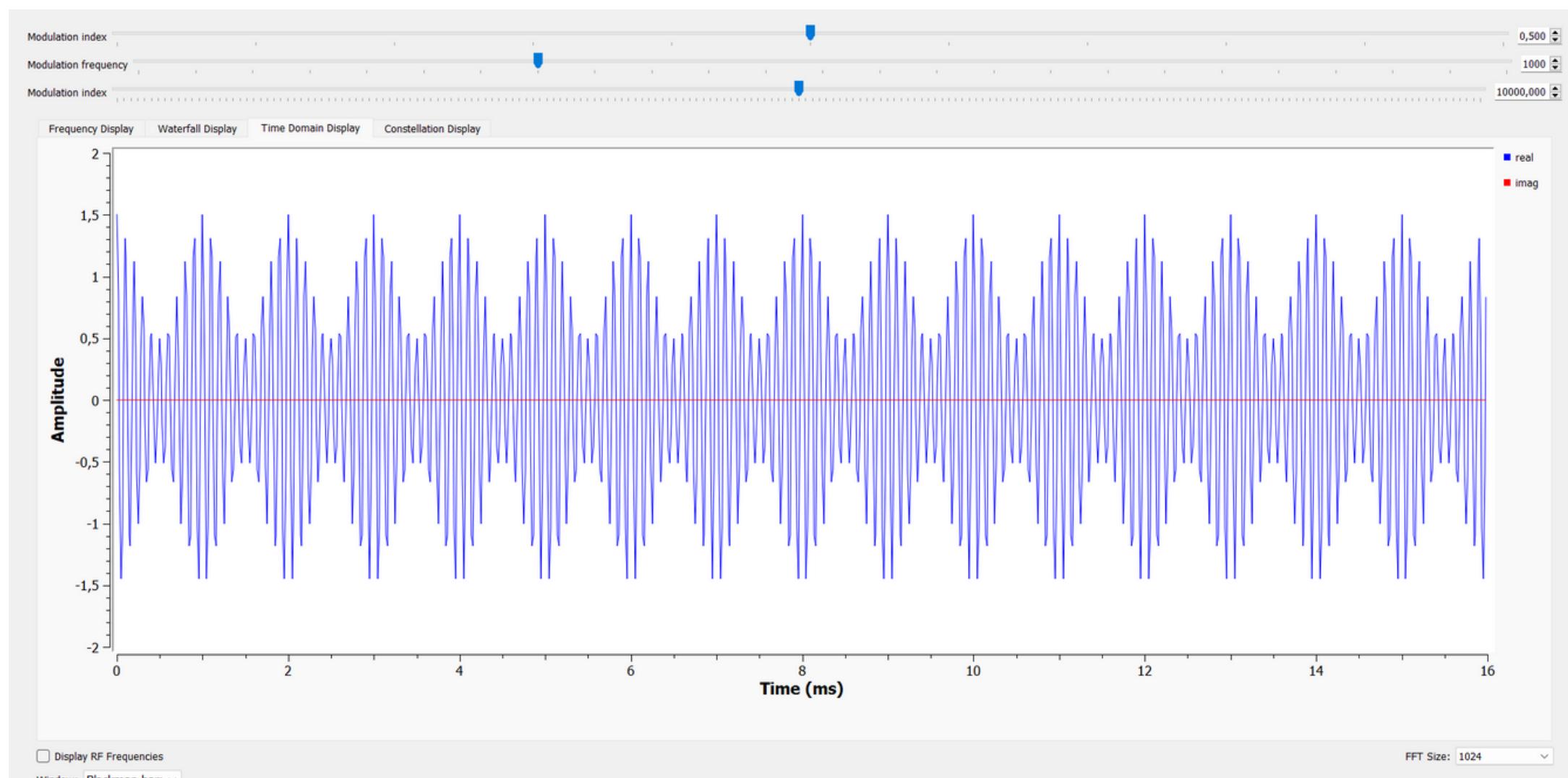
Vejamos como isso fica no GNU Radio:



- Esse bloco é um coeficiente que multiplica $m(t)$ na equação

Detalhando um pouco, temos duas fontes de sinal, uma para o sinal que será modulado e outra para a portadora. Em seguida, os blocos de adição e multiplicação são usados para construir literalmente a equação de modulação que vimos. As variáveis de faixa são configuradas para variar a frequência do sinal de 300 a 2700 Hz, a portadora de 5 KHz a 15 KHz e o o coeficiente de $m(t)$ de 0 a 1,0.

E este é o resultado:



E la vamos nois, melhor que ver é fazer (<https://abre.ai/modulacaoam>);

6 - Conclusão:



Poxa, já chegamos ao final do nosso estudo de sinais com GNU Radio.



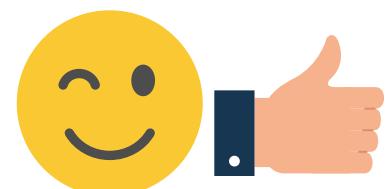
Esperamos que você tenha aprendido bastante, ou se já conhecia esses assuntos, que tenha ajudado a entender e fixar melhor esse conteúdo tão importante para telecomunicações, visto que são um pouco dos fundamentos dessa área.



Se tiver curiosidade em aprender mais sobre esta ferramenta, aqui está uma lista de tutoriais oficiais que lhe ajudará a conhecer melhor como ela funciona e como ela pode lhe auxiliar ainda mais nos seus estudos e pesquisas.

Link dos tutoriais: <https://wiki.gnuradio.org/index.php/Tutorials>

Por fim, parabéns por chegar até aqui, e continue empenhado nos seus estudos. Nos vemos novamente em breve!!



Referências:

PySDR: A Guide to SDR and DSP using Python — PySDR: A Guide to SDR and DSP using Python. Disponível em: <<https://pysdr.org/>>.

GNU Radio - The Free & Open Source Radio Ecosystem · GNU Radio. Disponível em: <<https://www.gnuradio.org/>>.

GNU Radio. Disponível em: <https://en.wikipedia.org/wiki/GNU_Radio>.

Estimativa de ruído gaussiano em imagem ruidosa usando MATLAB - Acervo Lima. Disponível em: <<https://acervolima.com/estimativa-de-ruido-gaussiano-em-imagem-ruidosa-usando-matlab/>>.

Signals and Systems Overview. Conteúdo disponível em: <https://www.tutorialspoint.com/signals_and_systems/signals_and_systems_overview.htm>.

What is Modulation ? Material disponível no endereço : <<https://www.geeksforgeeks.org/what-is-modulation/>>.

Modulation - Definition - Need for Modulation - Types of Modulation - Physics and Radio-Electronics. Disponível em: <<https://www.physics-and-radio-electronics.com/blog/modulation-definition-need-for-modulation-types-of-modulation/>>.

FOROUZAN, B. A.; SOPHIA CHUNG FEGAN; ARIOMALDO GRIESI. Comunicação de dados e redes de computadores. 4o. ed. São Paulo: McGraw Hill, 2008.