
Simulador de Filtros para Análise de Ondas

Objetivos

1. Familiarização com projeto de Filtros Determinísticos em LabVIEW
2. Apresentação de ferramentas de análise espectral de ondas em LabVIEW

Sumário

- 1 Apresentação de Análise Espectral em Ondas
 - 1.1 Representação de Sinais no Domínio da Frequência
 - 1.2 Exemplos
- 2 Análise de Arquitetura
 - 2.1 Análise do Executável
 - 2.2 Inicialização
 - 2.3 Processamento
 - 2.4 Finalização
- 3 Filtros
 - 3.1 Intuição Geral de um Filtro
 - 3.1.1 Projeto Equalizador – PSI3212
 - 3.2 Apresentação visual de um filtro
 - 3.3 Aplicação em um sinal
- 4 FFT
 - 4.1 Aplicação do Filtro em FFT (Comparação)
 - 4.2 Frequências dominantes
- 5 Conclusão do Programa
 - 5.1 Exercícios Complementares
 - 5.1.1 Ver um sinal FFT que seja REAL, função não bem comportada
 - 5.1.2 Filtro RC → Filtro Software

1 Apresentação de Análise Espectral em Ondas

Como visto em laboratório, uma das ferramentas disponíveis quando queremos realizar a caracterização de um sinal é a Série de Fourier. Segundo a teoria vista em aula, sabemos que: **certas séries podem descrever sinais periódicos**. A Série de Fourier, particularmente, possui a característica de representar um sinal qualquer por meio de uma soma de senos, com determinados valores de frequência, magnitude e fase. Neste tutorial veremos de maneira empírica **como a representação por meio de séries por ser útil para uma compreensão mais completa das características de um sinal qualquer**.

1.1 Representação de Sinais no Domínio da Frequência

Para a descrição de sinais periódicos, é utilizado o conceito de séries de Fourier, como visto em aula. Contudo, na maioria das situações nos deparamos com sinais que não possuem essa caracterização, são aperiódicos. Aqui nos deparamos com nossa primeira ressalva: para os sinais periódicos fazemos uso da Série de Fourier para caracterização no Domínio da Frequência, enquanto que nos casos onde não há periodicidade utilizamos a Transformada de Fourier, uma aproximação da Série de Fourier.

Com vista no esboço das ferramentas que serão utilizadas ao longo do curso, é importante ressaltar que todos os conceitos acima descritos serão vistos com mais calma e profundidade em muitas outras matérias. Para os fins deste tutorial, no entanto, basta que seja criada uma intuição sobre o conceito de representação no domínio da frequência: **todo e qualquer sinal possui quantidades determináveis de energia, dadas pelas amplitudes dos senos que o descreve, em certas frequências**.

1.2 Exemplos

Como exemplo da abstração descrita acima, tome um sinal senoidal, como na Fig. 1.1.

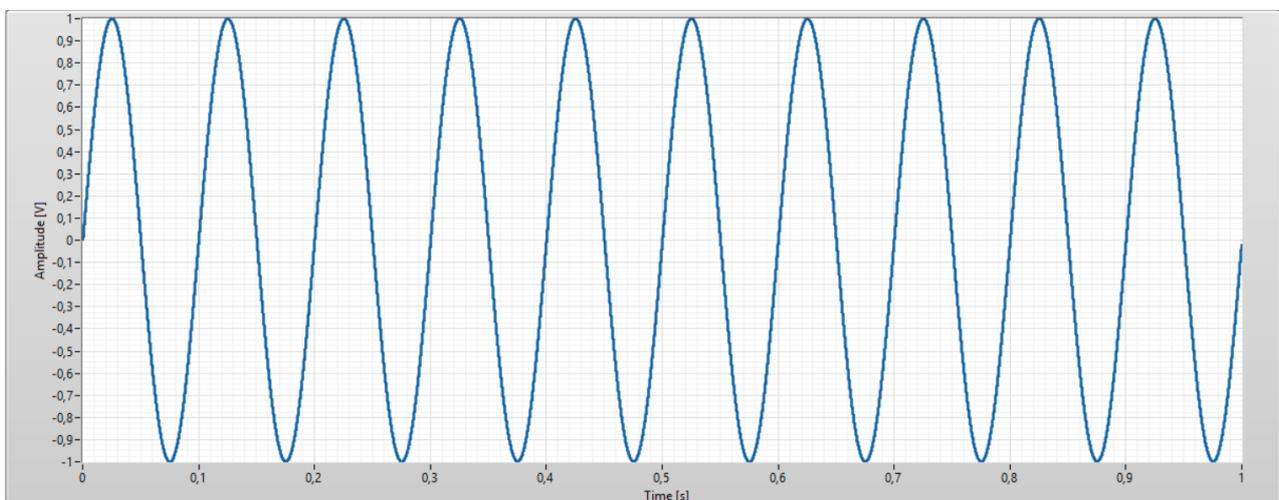


Fig. 1.1 – Sinal senoidal com frequência 10 Hz

Projeto de Análise de Ondas com Filtro

Veja que o sinal é um seno puro, com frequência de 10 Hz e amplitude 1 V_p. Se utilizarmos a teoria dada em aula, é trivial chegarmos à conclusão de que sua representação no domínio da frequência será uma raia de amplitude 1 e frequência 10 Hz, como na Fig. 1.2.

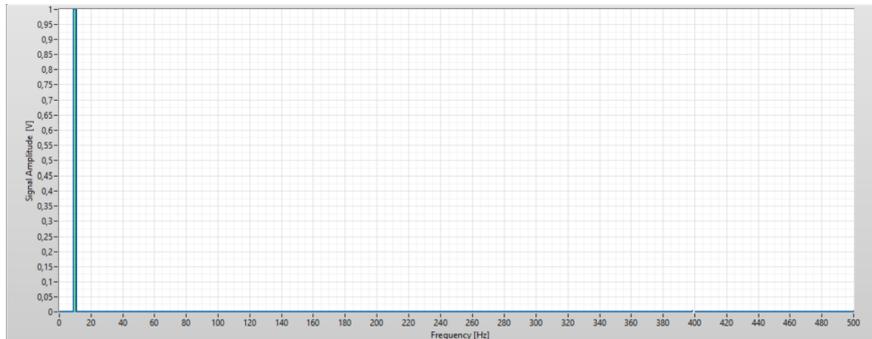


Fig. 1.2 – Representação pela Série de Fourier

O mesmo conceito pode ser aplicado a sinais um pouco mais complexos e menos intuitivos.

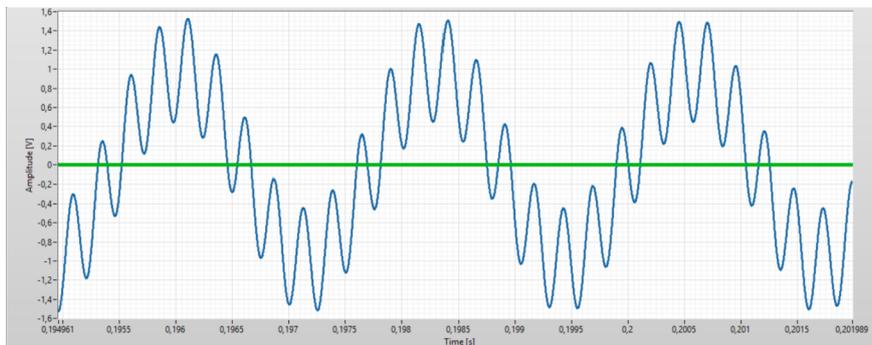


Fig. 1.3 – Soma de sinais senoidais

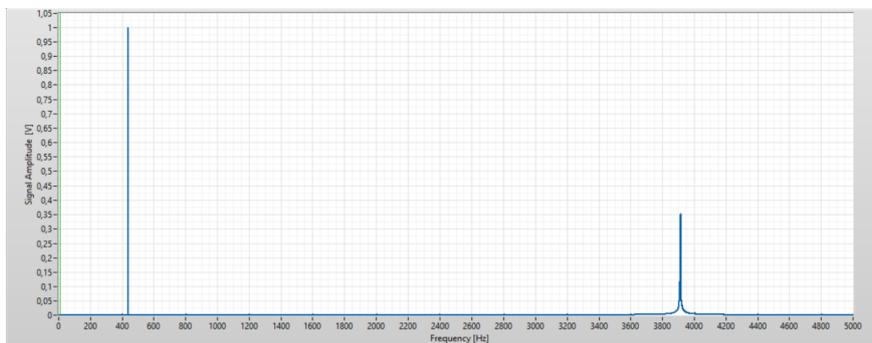


Fig. 1.4 – Representação Espectral da soma de dois sinais senoidais

Note que a utilização da representação no domínio da frequência, Fig. 1.4, deixa óbvia a natureza do sinal analisado, Fig. 1.3. Apesar de o exemplo escolhido ser relativamente simples de ser compreendido mesmo no domínio do tempo, para outros sinais veremos que a utilização da representação no domínio da frequência é imprescindível.

2 Análise de Arquitetura

Destacou-se, até agora, a importância de que seja realizada uma análise tanto no domínio do tempo, quanto no domínio da frequência. Feita essa breve introdução ao assunto de análise espectral, vamos agora entender a funcionalidade do VI que será construído.

2.1 Análise do Executável

Abra o projeto [**TUT3.lvproj**], e abra o VI principal: [**tut3_main.vi**]. Perceba que o VI é semelhante ao utilizado no tutorial passado. Ao lado esquerdo do painel frontal temos a região onde os sinais gerados serão visualizados, enquanto que na região à direita será realizado o controle de todos os parâmetros. Execute o VI e veja que é gerado um sinal de 440 Hz, como no programa anterior.

Neste tutorial, nosso objetivo é que seja criada uma intuição sobre as propriedades da FFT de um sinal, em detrimento de sua Série de Fourier, bem como uma compreensão mais detalhada de como realizar um projeto de filtro em LabVIEW.

Deste modo, para visualizar o sinal no domínio da frequência, clique no menu “Change Graph View” e selecione a opção “FFT View”.

Feito isso, podemos ver com mais clareza as frequências que compõe o sinal.

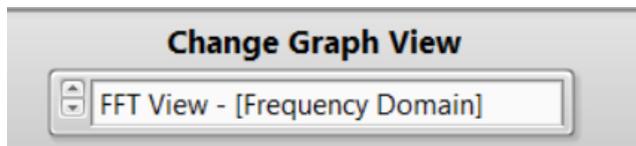


Fig. 2.1 – Menu de mudança de Visualização

Assim como fizemos no tutorial anterior, utilize as ferramentas da paleta do gráfico para aplicar zoom ao gráfico principal e utilize os cursores para identificar com precisão qual a frequência gerada e qual sua amplitude.

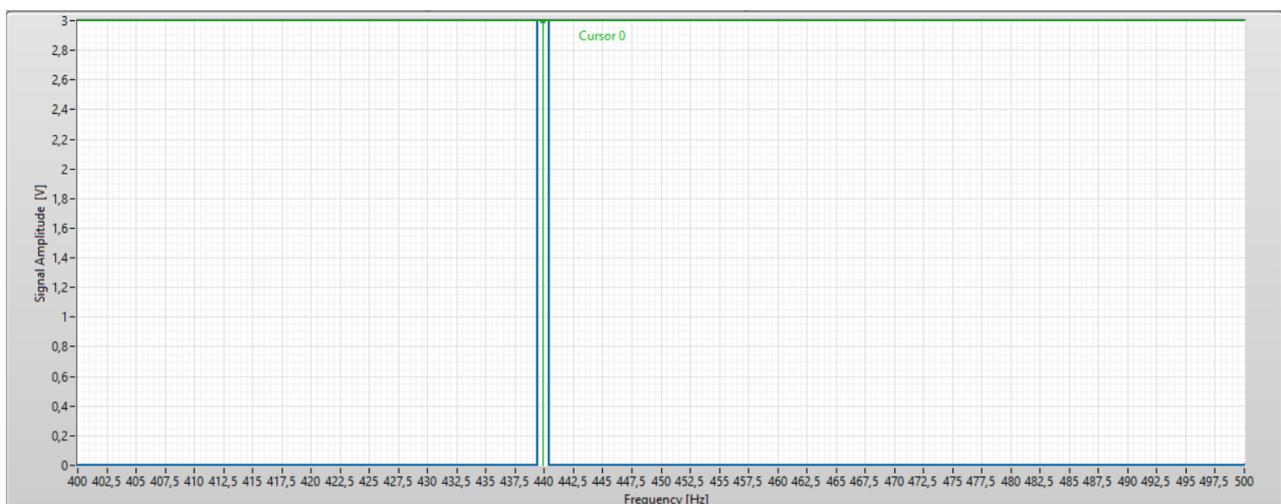


Fig. 2.2 – Representação espectral do sinal gerado

Projeto de Análise de Ondas com Filtro

Como vimos nos exemplos do item 1.2, podemos também gerar ondas mais complexas, como ondas de mais de uma senoide, ruídos e até mesmo senoides somadas a ruídos, os quais serão explorados mais na parte de filtros.

Selecione a opção no menu “Signal Type” a opção “Sine with Noise”, mude para a visualização do sinal no domínio do tempo, no menu “Change Graph View”, e veja que obtemos um sinal senoidal ruidoso, em vermelho, o qual passa por um filtro passa-baixa de primeira ordem, em azul, com frequência de corte 50kHz.

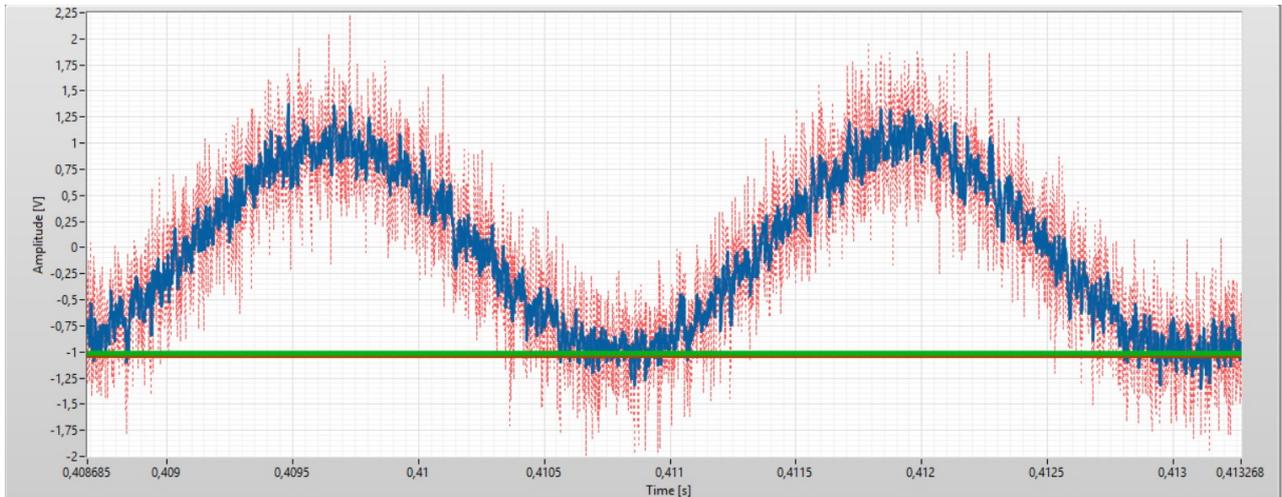
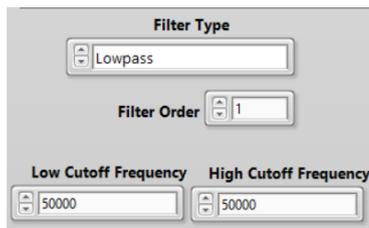
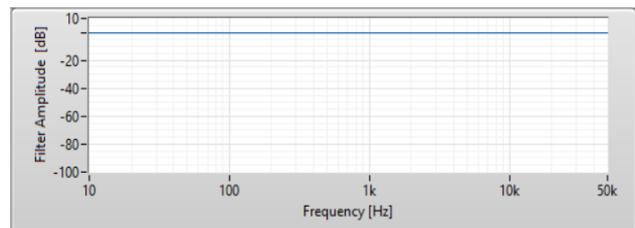


Fig. 2.3 – Onda senoidal com ruído

Podemos visualizar os controles do filtro no canto inferior direito do programa, mostrado na figura (a), e seu efeito no gráfico superior direito, figura (b).



(a)



(b)

Fig. 2.4 – Controles para geração de um filtro

Conhecidas as funcionalidades do programa, vamos agora compreender como foi construído e qual parte que programaremos neste tutorial.

Projeto de Análise de Ondas com Filtro

2.2 Inicialização

Como visto nos tutoriais anteriores, possuímos as três etapas características de um VI que segue uma arquitetura “General Purpose VI”. Deste modo, sabemos que na etapa de inicialização, serão iniciados, antes de qualquer processamento do programa, os valores de gráficos, controles e serão configurados os canais de áudio para que possamos utilizar os alto-falantes do computador.

2.3 Processamento

Na etapa de processamento foi realizada a programação da interface com usuário, assim como a geração de um sinal e seu devido processamento. Como no tutorial anterior, não estamos interessados na geração dos sinais, apenas na aplicação de um filtro sobre eles. Deste modo, focaremos apenas na programação da sessão em verde, denominada “Signal Frequency Domain Analysis”.

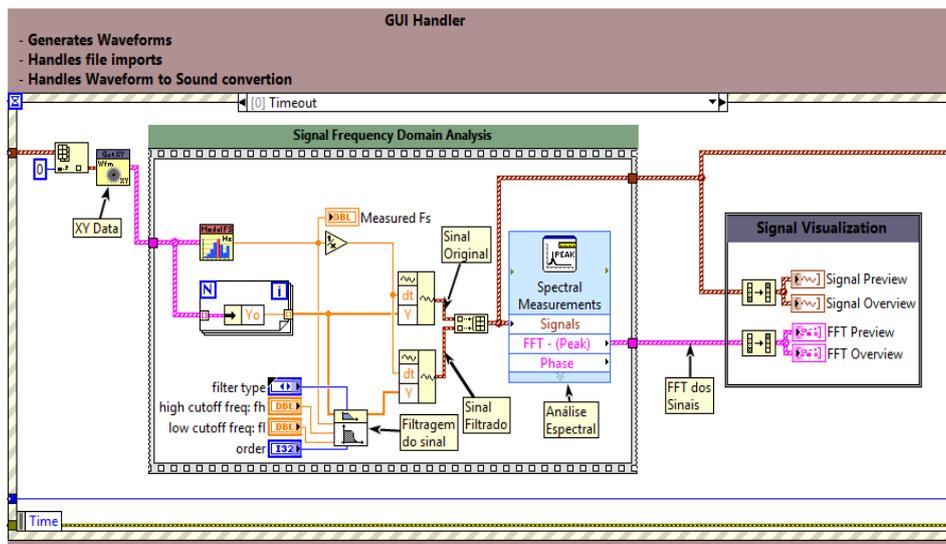


Fig. 2.5 – Etapa de Processamento

A programação que será realizada nesse tutorial é relativamente simples e demanda pouco tempo. Dito isso, é importante ressaltar que o objetivo do tutorial não é a memorização de todos os métodos para se realizar uma análise espectral em LabVIEW, mas sim a compreensão e consolidação dos conceitos de análise no domínio da frequência em seus usos mais casuais.

2.4 Finalização

Por sua vez, na etapa de finalização é realizado o fechamento adequado do dispositivo de áudio e a correção de quaisquer erros gerados ao longo do programa.



Fig. 2.6 – Finalização do VI

3 Filtros

Feche o programa [tut3_main.vi] e agora abra o programa [pr5.vi]. Vistos os conceitos básicos sobre transformações do domínio do tempo para o domínio da frequência, vamos tentar nos aprofundar no uso de filtros.

3.1 Intuição Geral de um Filtro

Em termos gerais, é importante compreendermos primeiramente que **quaisquer sistemas em engenharia tem, dentre outros parâmetros, uma faixa de operação, na qual nos interessamos pelos sinais gerados ou capturados.** Fora dessa região quaisquer sinais são indesejados, denominados ruídos.

O objetivo de um filtro, em síntese, é a retirada de ruídos de modo a preservar os sinais dentro da faixa de interesse. Na disciplina PSI3214 serão estudados filtros determinísticos com vista em sua caracterização no domínio da frequência, ou seja, todos os filtros que veremos, serão construídos a partir de suas atenuações em certos conjuntos de frequências, Fig. 3.1, assim como no projeto da disciplina de Laboratório de Práticas de Eletricidade, PSI3212.

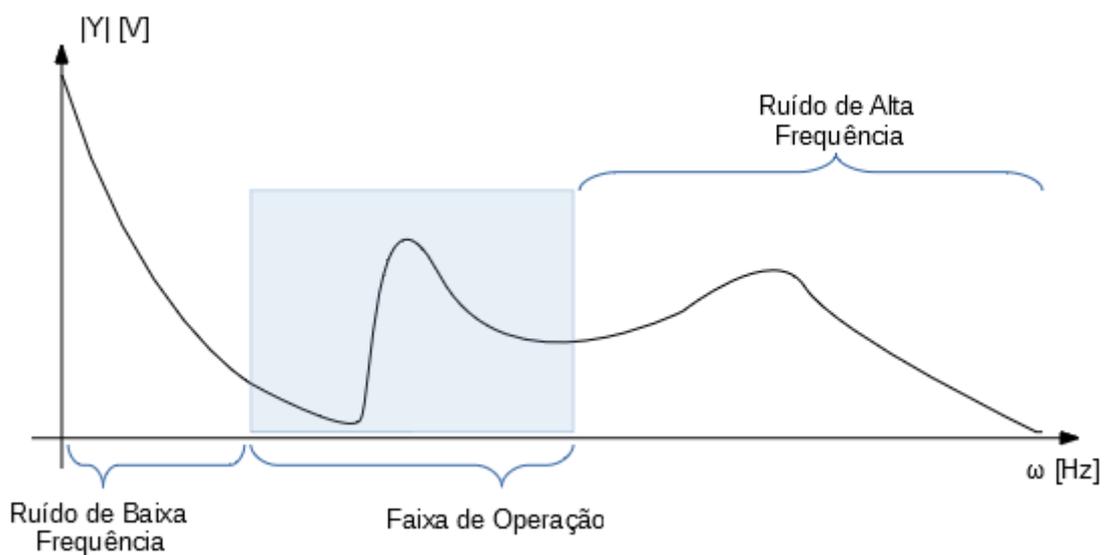


Fig. 3.1 – Faixa de operação de um sistema

3.2 Construção de Filtros em LabVIEW

Vamos agora à etapa de construção de nosso filtro. Note que a área em verde, designada “Signal Frequency Domain Analysis”, se encontra praticamente vazia, contendo apenas os controles necessários para criação do filtro desejado. Será nela em que nossos esforços serão concentrados.

Projeto de Análise de Ondas com Filtro

Para que seja feito qualquer cálculo com nosso vetor de pontos XY, é necessário que antes de tudo ele seja convertido em sinais com uma taxa amostragem determinada. **Vale lembrar que os pontos adquiridos por um conversor AD não necessariamente possuem o mesmo intervalo de tempo entre amostras.** Por isso é necessária essa primeira etapa. Como já realizamos essa tarefa no tutorial passado, utilizaremos um VI que já se encontra em nosso projeto. Para tal:

- (i) – Abra o projeto, [**TUT3.lvproj**]
- (ii) – Expanda a pasta “Auxiliary VI’s”
- (iii) – Clique e arraste o programa [**Get modal Sampling Frequency.vi**] para a área verde no diagrama de blocos de seu VI.

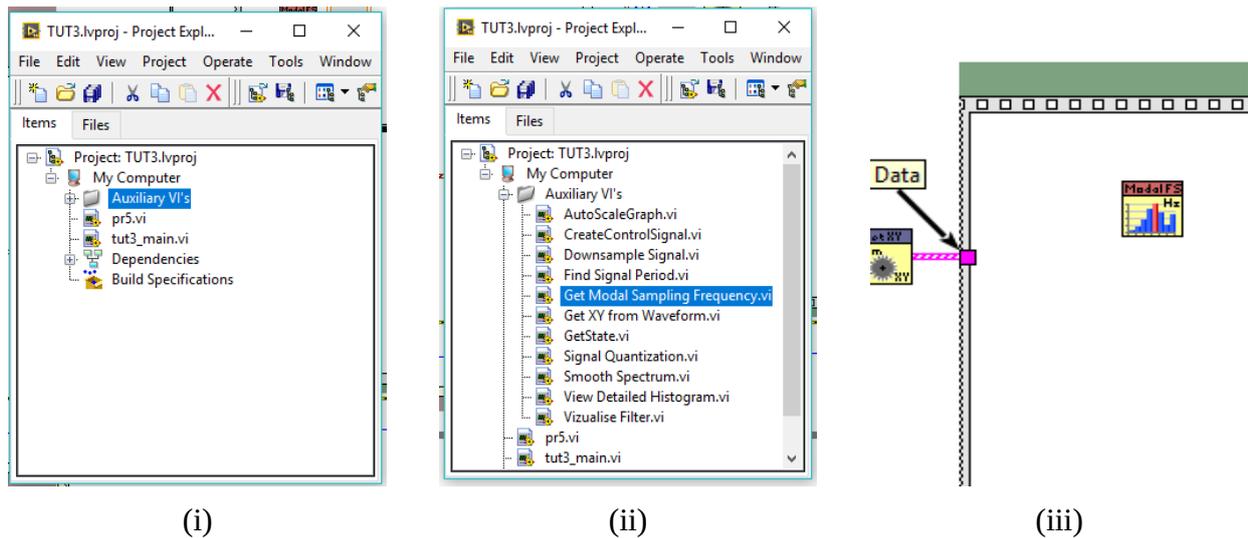


Fig. 3.2 – Procedimento para usar um VI que se encontra no projeto

Feito isso dê um duplo clique no VI arrastado, veremos seu Painel Frontal. Perceba que este VI é como todos os outros até agora trabalhados. Use o atalho Ctrl+E para abrir seu Diagrama de Blocos, Fig. 3.3, e veja seu funcionamento. Note as semelhanças com o programa desenvolvido no tutorial 2.

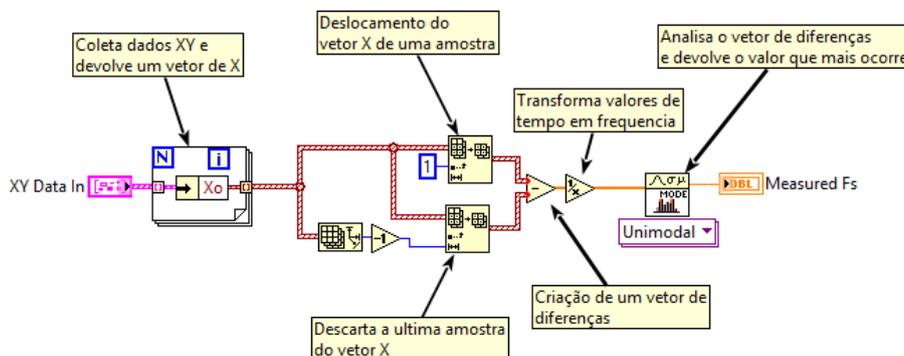


Fig. 3.3 – Programação para determinar frequência de amostragem

Projeto de Análise de Ondas com Filtro

Feche o Diagrama de Blocos e o Painel Frontal desse VI. Por meio do diagrama da Fig. 3.3, sabemos que esse programa tem por entrada dados no formato XY e uma saída que nos indica qual sua taxa de amostragem. Sendo assim:

- (i, ii) – Conecte o vetor de dados “XY Data” à entrada do bloco
- (iii) – Crie um indicador para a taxa de amostragem medida (clique com o botão direito no terminal de saída e selecione create>indicator)

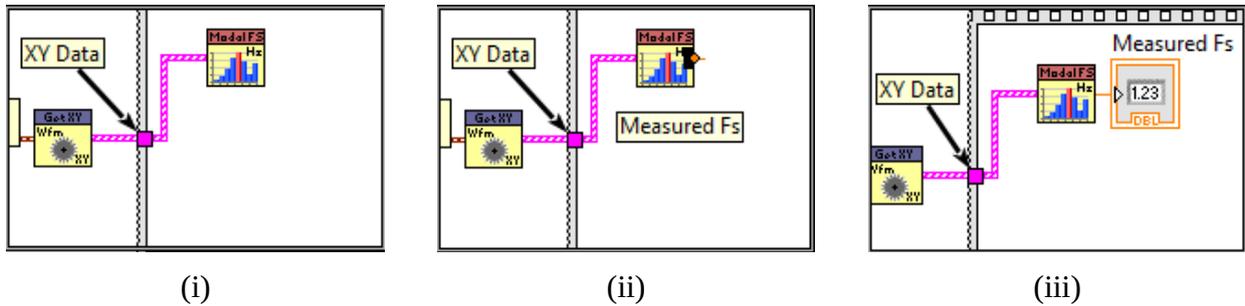


Fig. 3.4 – Conexão dos dados XY ao VI estimador da frequência de amostragem

Para que possamos comparar o sinal filtrado ao sinal original, vamos reconstruir o sinal adquirido de modo a que tenha agora uma taxa de amostragem constante. Para isso:

- (i) – Crie um laço “For”
- (ii) – Crie um bloco “Unbundle by Name” (aba “Cluster, Class & Variant”) e posicione dentro do laço
- (iii) – Conecte o sinal “XY Data” à entrada do laço; o túnel esquerdo do laço à entrada do bloco “Unbundle by Name”; a saída do bloco “Unbundle by Name” à borda direita do laço.
- (iv) – Mude, se necessário o elemento dentro do bloco “Unbundle by Name” para Yo (clique com o botão esquerdo para abrir o menu e selecione).

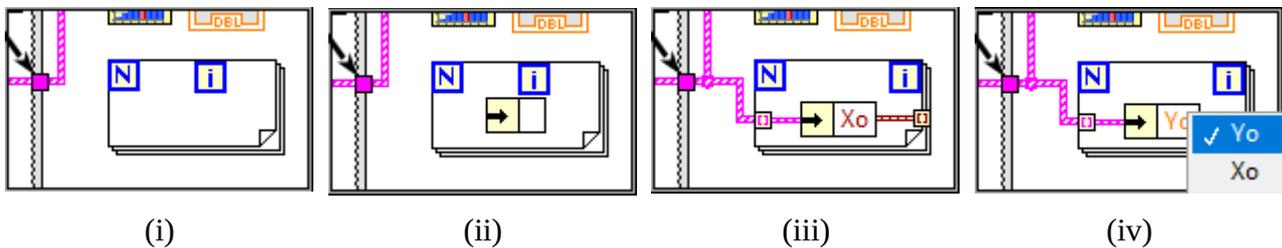


Fig. 3.5 – Seleção das amostras Y do sinal de entrada

Projeto de Análise de Ondas com Filtro

- (i) – Crie um bloco “Build Waveform” (na aba “Waveform”) e um bloco “reciprocal” (na aba “Numeric”)
- (ii, iii) – Expanda o bloco “Build Waveform” arrastando sua parte superior. Clique com o botão esquerdo no elemento recém-criado para abrir o menu e selecione “dt”
- (iv) – Conecte sinal “Measured Fs” à entrada do bloco “reciprocal” e a saída desse bloco ao terminal “dt” do bloco “Build Waveform”. Conecte a saída do laço “For” ao terminal “Y” do bloco “Build Waveform”.

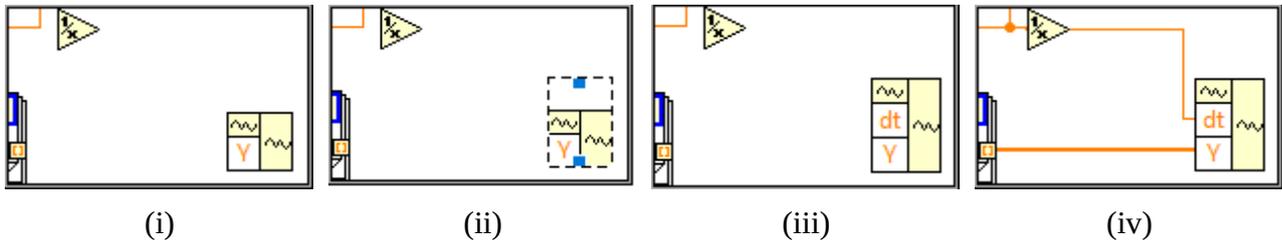


Fig. 3.6 – Reconstrução do sinal original com taxa de amostragem constante

Feito isso coletamos do vetor XY apenas as amostras y, criamos um vetor Y com elas e construímos um sinal do tipo “Waveform” de taxa de amostragem constante.

Vamos agora construir a etapa de filtragem do sinal. Abra a paleta de funções do Diagrama de Blocos, navegue até a seção “Signal Processing>Filters” e repare a vasta gama de filtros que podemos fazer uso, cada qual com sua vantagem e desvantagem. Nesse tutorial utilizaremos o mais simples e intuitivos dos filtros disponíveis: “Butterworth Filter”. Veremos que esse tipo de filtro é ideal para simularmos circuitos RC ou outros de maior ordem. Após a construção do filtro veremos com mais calma cada um de seus aspectos. Por enquanto:

- (i) – Crie um bloco “Butterworth Filter”
- (ii) – Conecte o sinal “Measured Fs” à entrada “sampling freq: fs” e o sinal Y, de saída do laço “For”, à entrada “X” do bloco do filtro.
- (iii) – Para os terminais de entrada: “filter type”, “high cutoff freq: fh”, “low cutoff freq: fl” e “order”, conecte os controles às entradas correspondentes do bloco do filtro.

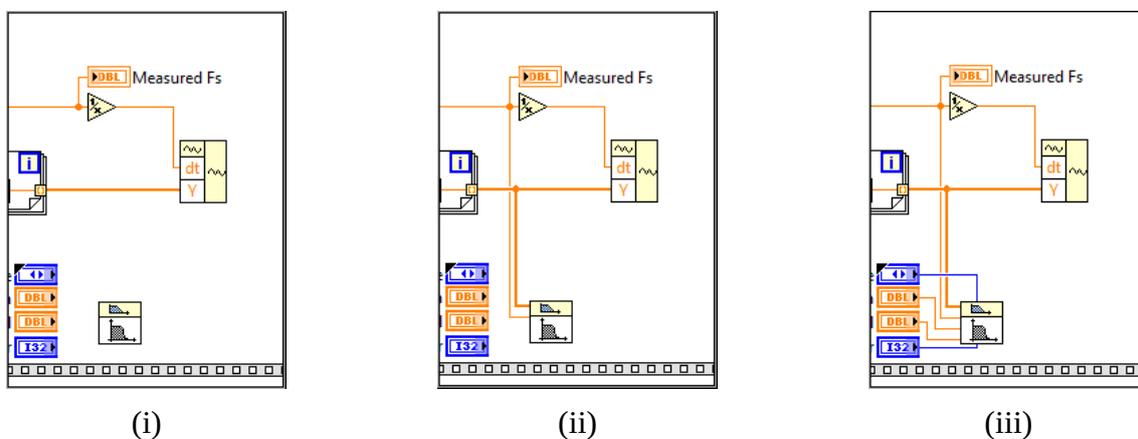


Fig. 3.7 – Criação do filtro Butterworth

Projeto de Análise de Ondas com Filtro

- (i) – Crie um bloco “Build Waveform” e arraste sua borda superior para que possamos modificar mais de um parâmetro ao mesmo tempo.
- (ii, iii) – Conecte o sinal dt usado no bloco acima ao terminal dt desse bloco, e o sinal de saída do bloco “Butterworth Filter”, “Filtered X” ao terminal Y do bloco “Build Waveform”

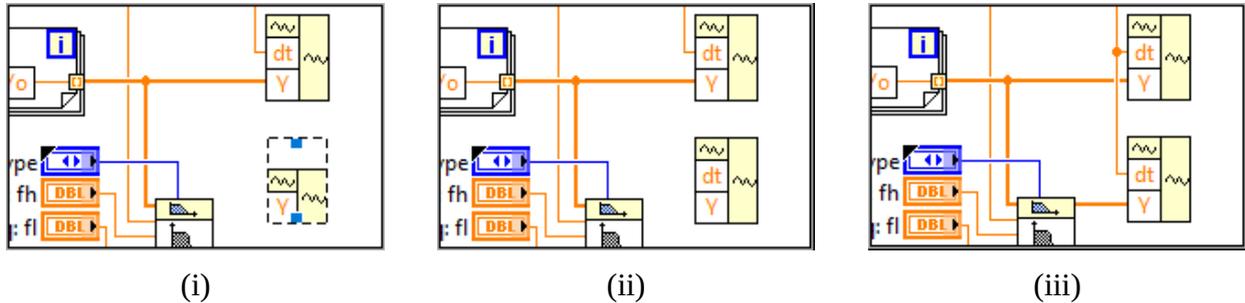


Fig. 3.8 – Construção do sinal filtrado

Terminada a etapa anterior teremos criado dois sinais distintos. No bloco “Build Waveform” superior será gerado o sinal com os valores de Y originais, imutados. O bloco inferior, por sua vez, será gerado um sinal que apresenta valores de Y atenuados de acordo com o filtro criado.

Vamos agora criar um vetor contendo os dois sinais para que possam ser visualizados no mesmo gráfico.

- (i) – Crie um bloco “Build Array” (na aba “Array”) e expanda-o para que possa acomodar duas entradas.
- (ii) – Conecte as saídas dos blocos “Build Waveform” como na figura abaixo. O sinal superior deverá entrar no terminal inferior do bloco “Build Array”, enquanto que a saída do bloco inferior deverá ser conectada ao terminal superior.
- (iii) – Conecte a saída do bloco “Build Array” ao túnel marrom superior na borda direita da região delimitada.

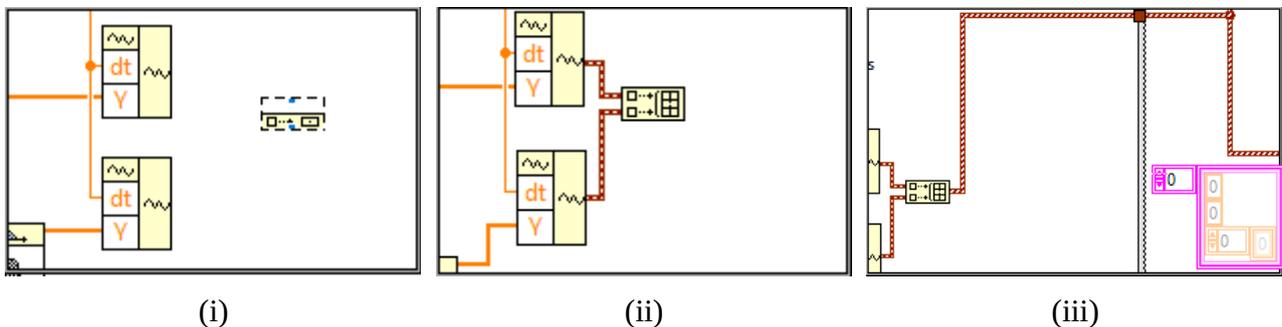


Fig. 3.9 – Junção dos dois sinais em um vetor

Com isso temos um programa funcional pronto para ser testado! Execute o programa e veja se tudo funciona segundo o esperado.

Projeto de Análise de Ondas com Filtro

3.3 Aplicação em um sinal

Com o programa em execução vamos verificar alguns pontos interessantes. Primeiramente, configure os controles para que:

- | | | | |
|-----------------------|--------|---------------------|---------|
| – Signal Type: | Sine | – Filter Type: | Lowpass |
| – Signal Amplitude: | 1 V | – Low Cutoff Freq: | 5E+4 Hz |
| – Signal Frequency: | 440 Hz | – High Cutoff Freq: | 5E+4 Hz |
| – Sampling Frequency: | 1E+6 | – Order: | 1 |

Olhe para o sinal no gráfico principal e veja que a senoide criada tem atenuação desprezível, pois sua amplitude é praticamente 3 V.

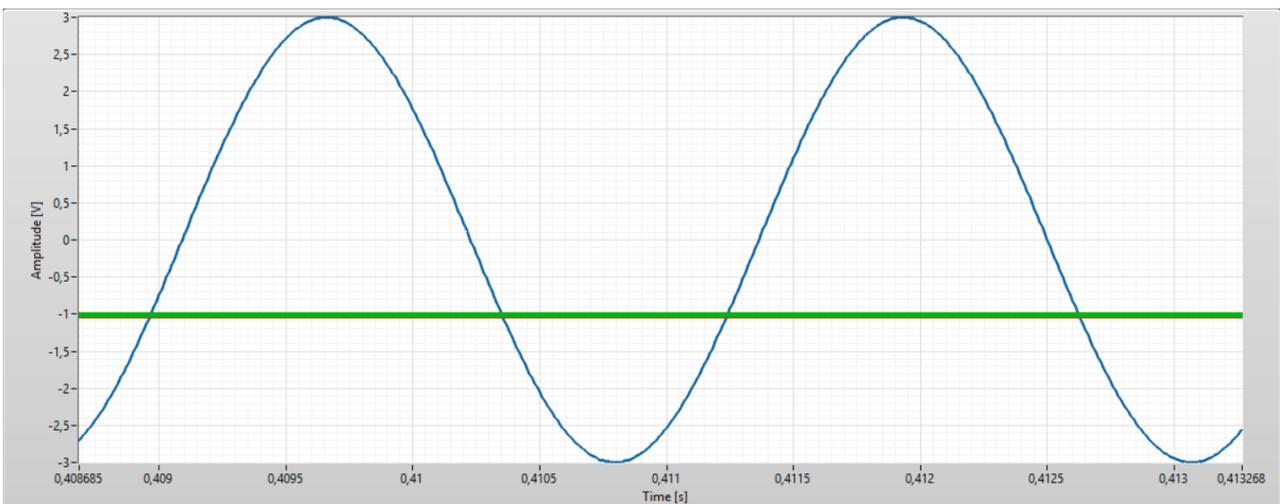


Fig. 3.10 – Senoide 440 Hz com filtro passa-baixa ($F_C = 50000$ Hz)

Mude o valor do controle “Low Cutoff Freq” para 440 Hz e utilize os cursores para verificar que o ganho entre o sinal original, em vermelho, e o sinal filtrado, em azul, é de -3 dB.

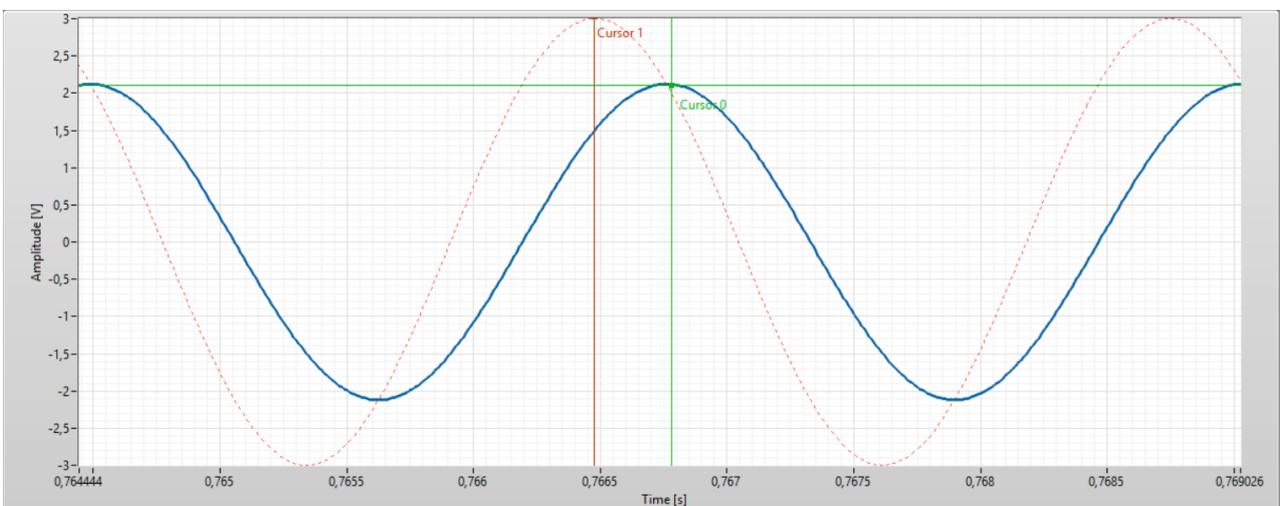


Fig. 3.11 – Senoide 440 Hz com filtro passa-baixa ($F_C = 440$ Hz)

Projeto de Análise de Ondas com Filtro

Note também que utilizando uma simples regra de proporções concluímos que há um atraso de fase de aproximadamente 45° , como esperado de um filtro passa-baixa de primeira ordem. Mude agora o controle “Signal Type” para “Square” e veja o resultado.

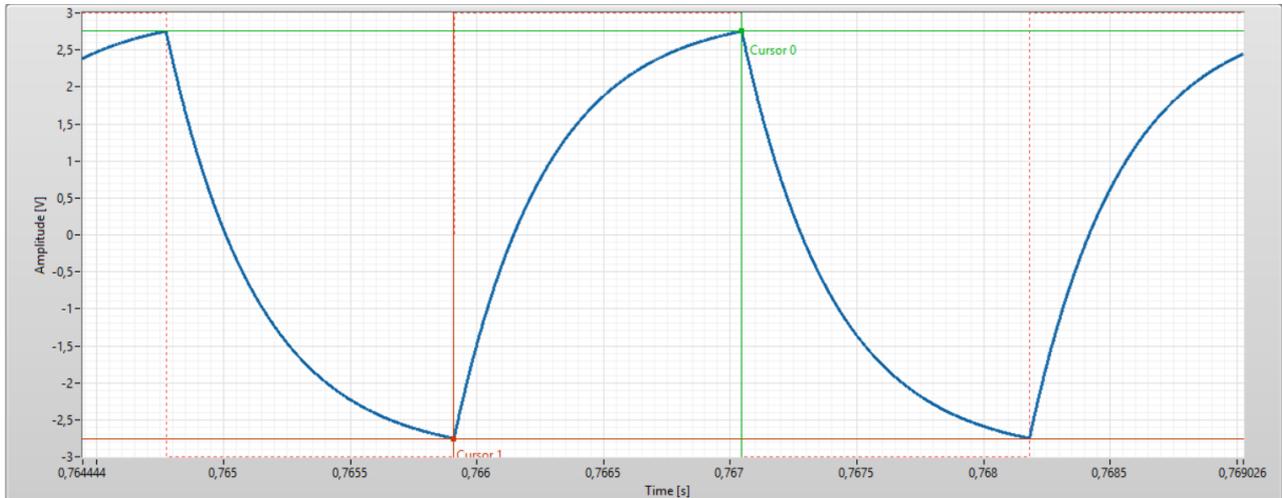


Fig. 3.12 – Onda quadrada 440 Hz com filtro passa-baixa ($F_C = 440$ Hz)

Perceba que temos uma resposta idêntica àquela de um circuito RC, como realizado na simulação do tutorial 1. Aqui, no entanto, utilizamos apenas um parâmetro, a frequência de corte do filtro, para determinar uma resposta que foi obtida no tutorial 1 com grande sacrifício de compreensibilidade.

Vamos agora iniciar a parte que mais nos interessa nesse tutorial. Mude o valor do controle “Signal Type” para “Sine with Noise” e veja os resultados.

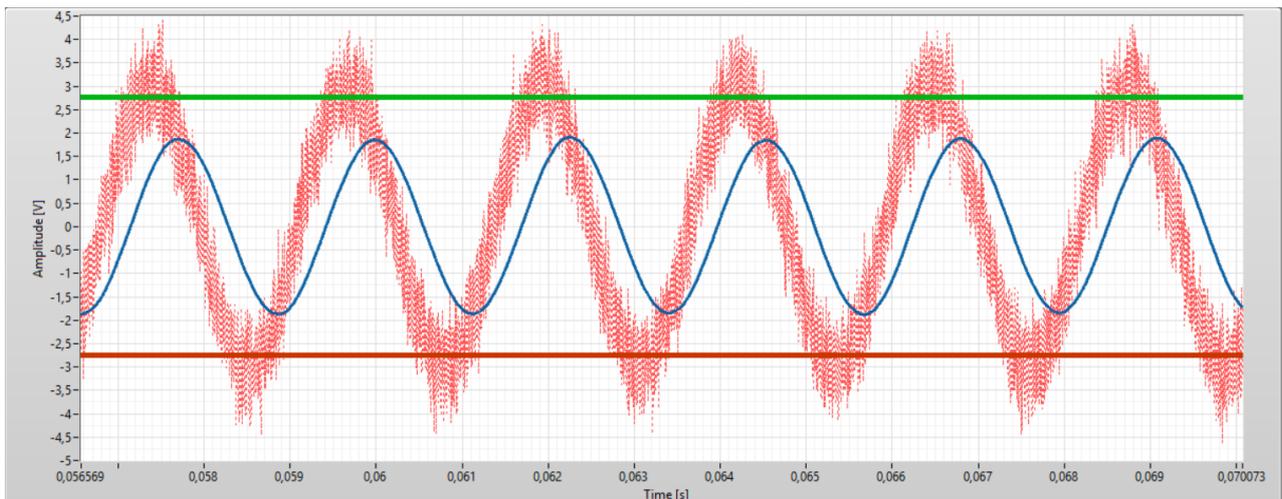


Fig. 3.13 – Senoide 440 Hz + ruído gaussiano com filtro passa-baixa ($F_C = 440$ Hz)

Finalmente os frutos de nossos esforços estão sendo recompensados! Fomos capazes de obter uma estimativa, em azul, de qual seria o sinal sem ruído medido em nosso sensor.

4 FFT

Pronto! Temos finalmente um filtro funcional correto? De fato, o filtro construído atenua frequências indesejadas para o sinal testado. Contudo, note que para outras frequências nosso sinal será atenuado muito mais do que o desejado. Tente mudar as frequências do controle “Signal Frequency” para averiguar esse comportamento.

Para compreender melhor como nosso sinal é atenuado em relação ao original, vamos implementar um bloco de Transformada de Fourier para que possamos visualizá-lo no domínio da frequência. Pare a execução do programa e abra seu Diagrama de Blocos. A implementação de medidas como FFT em LabVIEW é simples e direta:

- (i) – Delete a constante “FFT Constant” selecionando-a e apertando o botão “delete”. Crie um bloco “Spectral Measurements” (na aba “Signal Processing > Waveform Measurements”). Feita sua criação, deverá surgir uma janela de configuração, na qual podemos selecionar parâmetros pertinentes.
- (ii) – Configure o bloco para que apresente as medidas em Magnitude Pico, Linear e selecione a opção de nenhuma janela (“None”). Feito isso, clique em OK.
- (iii) – Conecte o sinal de saída do bloco “Build Array” ao terminal “Signals” do bloco de FFT e em seguida conecte o terminal de saída “FFT – (Peak)” aos dois indicadores “FFT Preview” e “FFT Overview”, localizados na área delimitada “Signal Visualization”

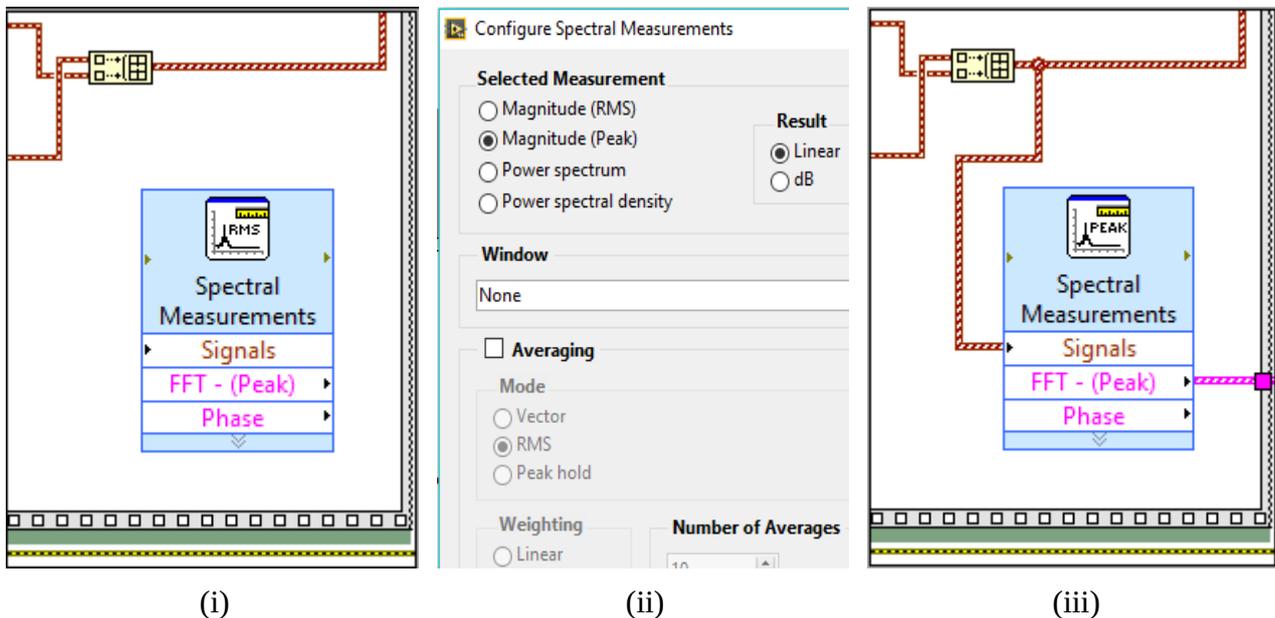


Fig. 4.1 – Construção das medidas espectrais

Feito isso temos nosso programa de LabVIEW pronto para avaliar tanto no domínio do tempo quanto no domínio da frequência as propriedades de nosso sinal de entrada e do filtro projetado.

Projeto de Análise de Ondas com Filtro

4.1 Aplicação do Filtro em FFT (Comparação)

Execute o programa e insira os valores abaixo nos controles:

- | | | | |
|-----------------------|--------|---------------------|---------|
| – Signal Type: | Square | – Filter Type: | Lowpass |
| – Signal Amplitude: | 1 V | – Low Cutoff Freq: | 5E+5 Hz |
| – Signal Frequency: | 440 Hz | – High Cutoff Freq: | 5E+5 Hz |
| – Sampling Frequency: | 1E+6 | – Order: | 1 |

Note que obteremos uma onda quadrada como na figura abaixo:

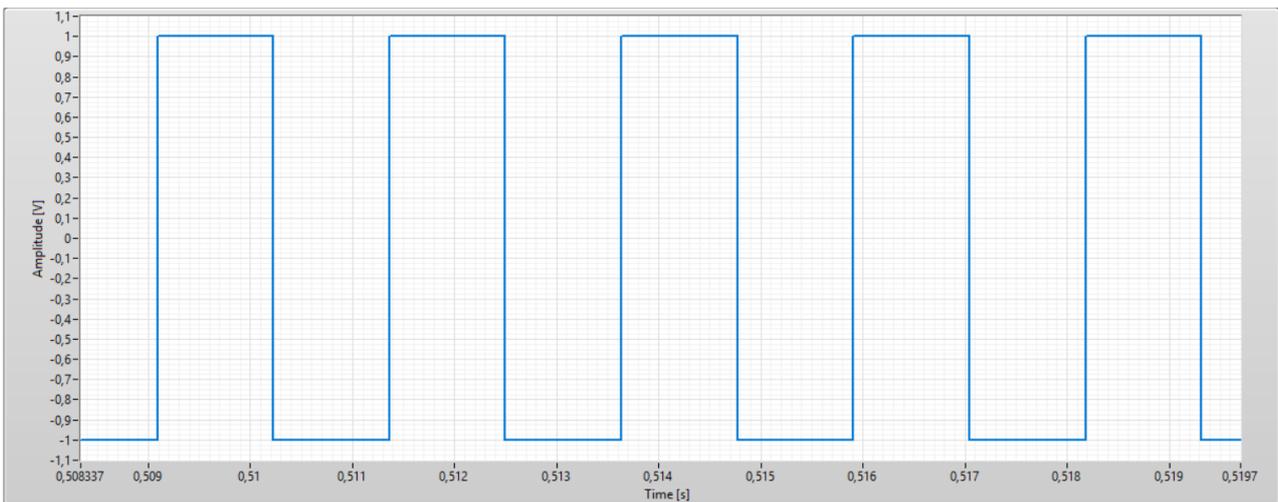


Fig. 4.2 – Onda quadrada 440 Hz com filtro passa-baixa ($F_C = 50000$ Hz)

Agora que podemos visualizar a FFT do sinal, clique no menu “Change Graph View” e selecione a opção “FFT View”. Nela deveremos ver a seguinte figura:

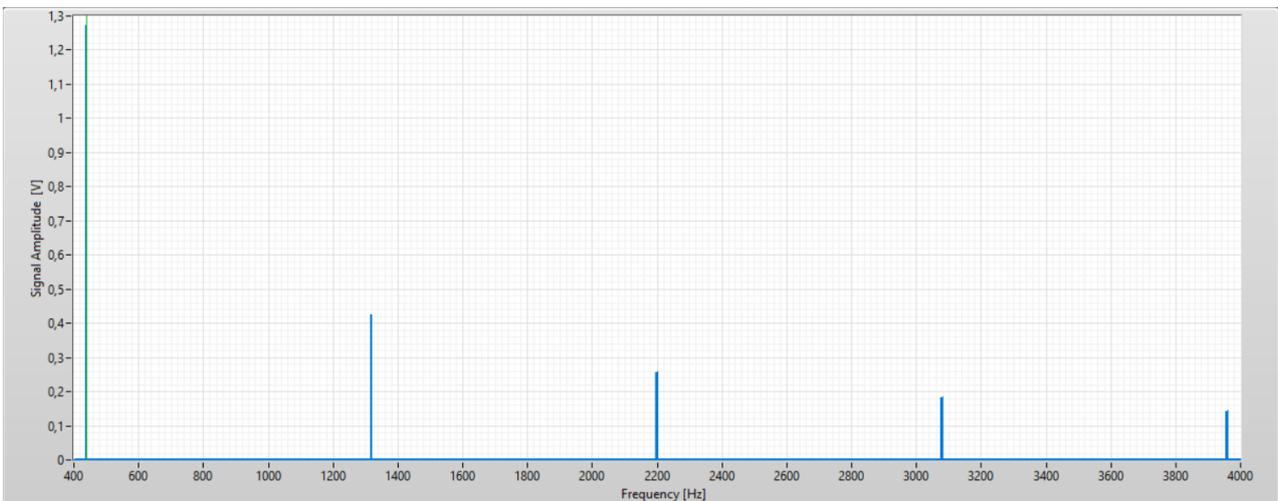


Fig. 4.3 – Espectro de uma onda quadrada de 440 Hz

Projeto de Análise de Ondas com Filtro

Note que a figura mostra as raiais correspondentes à Série de Fourier do sinal, como esperado da FFT, uma aproximação da Série de Fourier para sinais não periódicos.

Modifique agora o valor da frequência de corte do filtro passa baixa para 440 Hz e veja como os picos da FFT se comportam.

Mude o gráfico para domínio do tempo e veja o sinal correspondente. Note que o sinal se deforma por conta da atenuação dos harmônicos ímpares de ordem maior que, apesar de possuírem frequências maiores do que a frequência de base, são imprescindíveis para caracterização do sinal quadrado.

Do mesmo modo, outros sinais mais complexos como notas musicais também possuem caracterizações no domínio da frequência mais complexos. Utilizando o botão de importação do VI, importe o arquivo de nota musical utilizado no tutorial anterior e veja como é caracterizado por suas frequências.

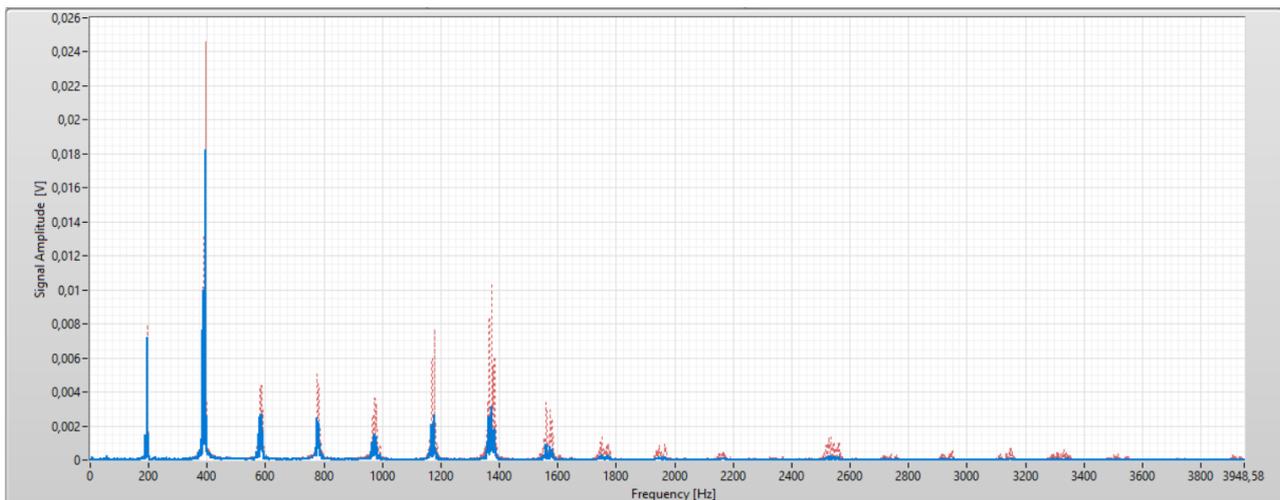


Fig. 4.4 – Distribuição espectral de uma nota de violino

Veja que temos frequências de importância até 2 kHz. Modifique a frequência de corte do filtro para que inclua esse harmônico. Veja se seu sinal filtrado no domínio do tempo aproxima adequadamente o original.

4.2 Frequências dominantes

Vimos no item anterior que um sinal pode conter diversas frequências que o caracterizam, e que dependendo dessas frequências, devemos projetar nosso filtro adequadamente. Perceba que para um sinal desconhecido adquirido por um ADC, tais frequências, denominadas “Frequências Dominantes”, indicam quais as frequências de nosso sinal que contém mais energia.

Lembre-se de que o problema de identificação de frequência de um sinal desconhecido foi atacado no tutorial anterior e que, mediante sinais mais complexos ou ruidosos o algoritmo desenvolvido não foi muito preciso. Para o sinal da Fig. 4.4 nosso algoritmo no domínio do tempo apresentava medições da ordem de 350 Hz, 50 Hz abaixo da frequência dominante de 400 Hz, a mais predominante no sinal.

5 Conclusão do Programa

Ao longo desse tutorial vimos como sinais podem ser filtrados em LabVIEW e como obter uma representação de um sinal no domínio da frequência. Por meio dessas ferramentas, pudemos compreender melhor a natureza de alguns sinais como de ondas quadradas e notas musicais, de modo a que dado um sinal de entrada dentro de certas condições de contorno, sabemos qual o filtro ótimo para ele. Vamos agora colocar todos esses conceitos e ferramentas em prática.

No projeto da disciplina utilizaremos um microcontrolador com determinadas características para adquirir uma gama de sinais de entrada. Gostaríamos, a princípio de que sejam capturados valores de tensão advindos de um microfone para que possamos identificar qual nota, ou som está sendo produzido. Deste modo, devemos levar em conta também os aspectos que caracterizam nosso ADC.

5.1 Exercícios

Abra o programa [**ex3.vi**] e complete-o. Ao final você deverá ser capaz de projetar um sistema primitivo de aquisição.

5.2 Exercícios Complementares