
Simulador de Conversor AD para Análise de Ondas

Objetivos

1. Familiarização com estruturas de arquivos de som processados pelo LabVIEW
2. Apresentação de ferramentas de análise temporal de ondas.

Sumário

- 1 Reforçando Arquiteturas em LabVIEW: “General Purpose VI”
 - 1.1 Análise do Executável
 - 1.2 Inicialização
 - 1.3 Processamento
 - 1.4 Finalização
- 2 Análise do Sinal por Inspeção Visual
 - 2.1 Taxa de Amostragem
 - 2.2 Quantização
 - 2.3 Período
 - 2.4 Limitações da Inspeção Visual
- 3 Medições Automatizadas
 - 3.1 Taxa de Amostragem
 - 3.2 Quantização
- 4 Conclusão do Programa
 - 4.1 Exercícios Complementares

1 Reforçando Arquiteturas em LabVIEW: “General Purpose VI”

No tutorial passado (Tutorial 1) construímos um VI capaz de simular um circuito RL ou RC por meio de suas fórmulas. Tal construção foi baseada na arquitetura “General Purpose VI”, na qual se divide o problema a ser resolvido em 3 etapas: (1) Inicialização de Parâmetros, (2) Processamento e Simulação, (3) Finalização do Programa. Do mesmo modo como foi realizado no tutorial passado, vamos verificar como é aplicada a mesma metodologia para um problema de simulação e análise de Conversão AD.

O objetivo deste tutorial é explicar as diferentes partes que compõe um processo de aquisição de dados, criar uma intuição sobre os conceitos mais básicos de conversão AD e, por fim, demonstrar como é possível transformar tal intuição matemática em blocos simples de programação em LabVIEW.

Apesar das etapas de “Geração de Sinal” estarem presentes no VI a ser estudado neste tutorial, estas não entram no escopo desse estudo, e são apresentadas como caixas pretas, nas quais iremos apenas indicar quais as entradas e saídas.

1.1 Análise do Executável

Antes de analisar as etapas do VI que correspondem à arquitetura “General Purpose VI”, vamos averiguar quais as funcionalidades presentes no programa. Abra o programa [tut2_main.vi] dentro do projeto [Tutorial 2.lvproj] e execute o VI.

Perceba que o VI é dividido em diversas secções. Na esquerda temos dois gráficos: o gráfico na região superior mostra uma visão geral do sinal em questão, enquanto que o gráfico inferior mostra uma visão mais detalhada do sinal, ou seja, com uma escala ampliada de tensão.

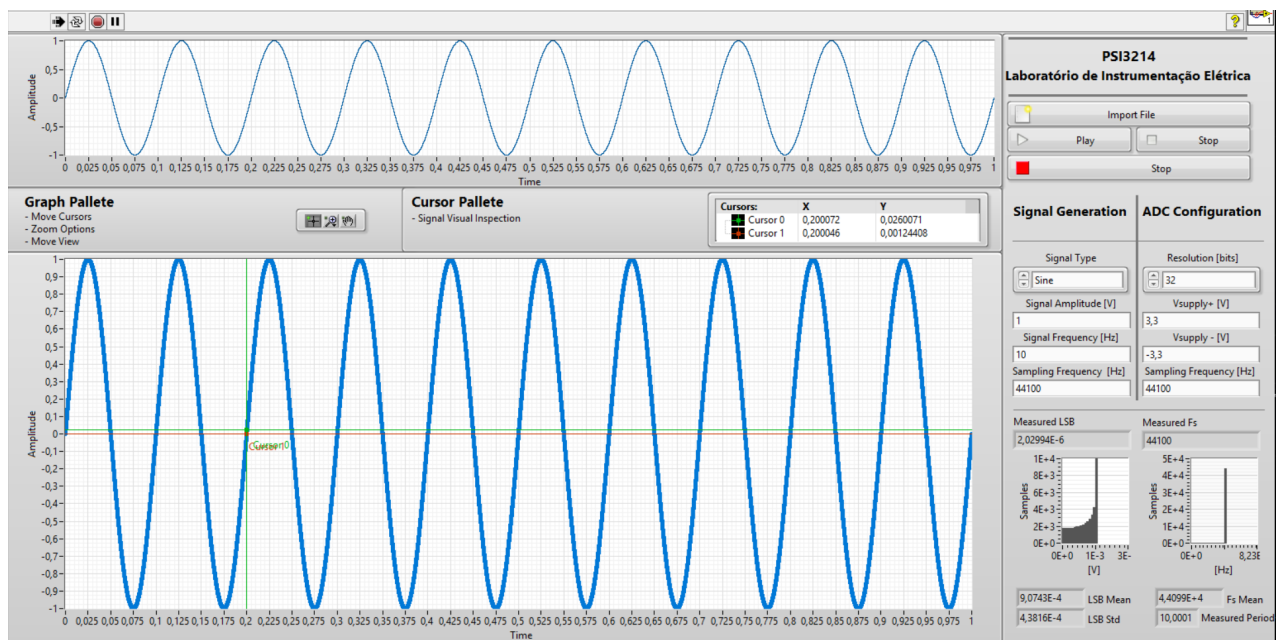


Fig. 1.1 – Apresentação do sinal gerado pelo VI em dois gráficos

Análise de Ondas com Conversor AD

Do lado direito do programa, por sua vez, temos os controles gerais do VI (Fig. 1.2), nos quais podemos escolher os parâmetros de simulação do sinal gerado através do gerador de ondas e as configurações do conversor AD.

Na parte inferior dessa seção podemos visualizar as medições mais pertinentes ao programa, como o valor do LSB (*Measured LSB*, em Volts) e o valor da Taxa de Aquisição de Dados (*Measured Fs*, em Hertz).

Para testar este VI, na seção de geração de sinais mude os valores para:

- Signal Type: Sine
- Signal Amplitude: 4 V
- Signal Frequency: 10 Hz
- Sampling Frequency: 44100 Hz

Perceba que conforme os parâmetros são modificados, a forma da onda muda, como esperado.

Agora, na seção de configuração do ADC mude os valores para:

- Resolution: 6 bits
- $V_{\text{supply}+}$: +3.3 V
- $V_{\text{supply}-}$: -3.3 V
- Sampling Frequency: 44100 Hz

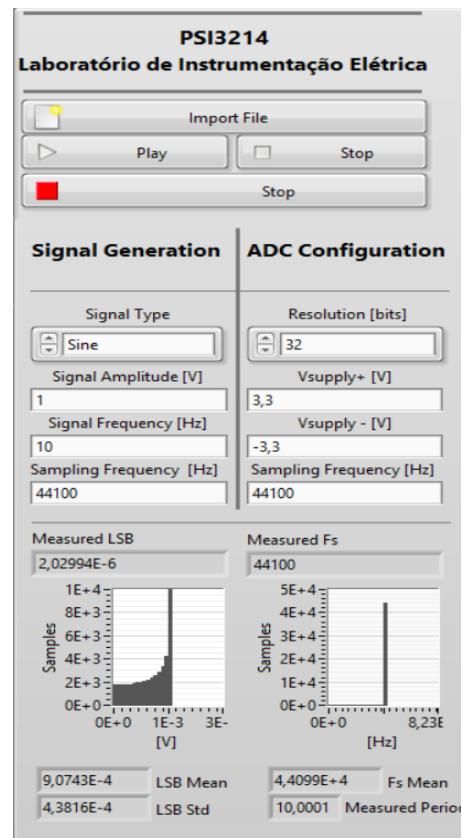


Fig. 1.2 – Controles do VI

Note que o gráfico nos mostra duas ondas diferentes conforme os parâmetros acima são modificados. A onda em vermelho corresponde ao sinal adquirido por um ADC ideal, capaz de reproduzir perfeitamente o sinal que sai do gerador, enquanto que a onda azul mostra a simulação de um sinal adquirido por um ADC não ideal, com as características indicadas no VI (selecionadas por você). Você consegue entender porque o sinal amostrado fica com formato saturado?

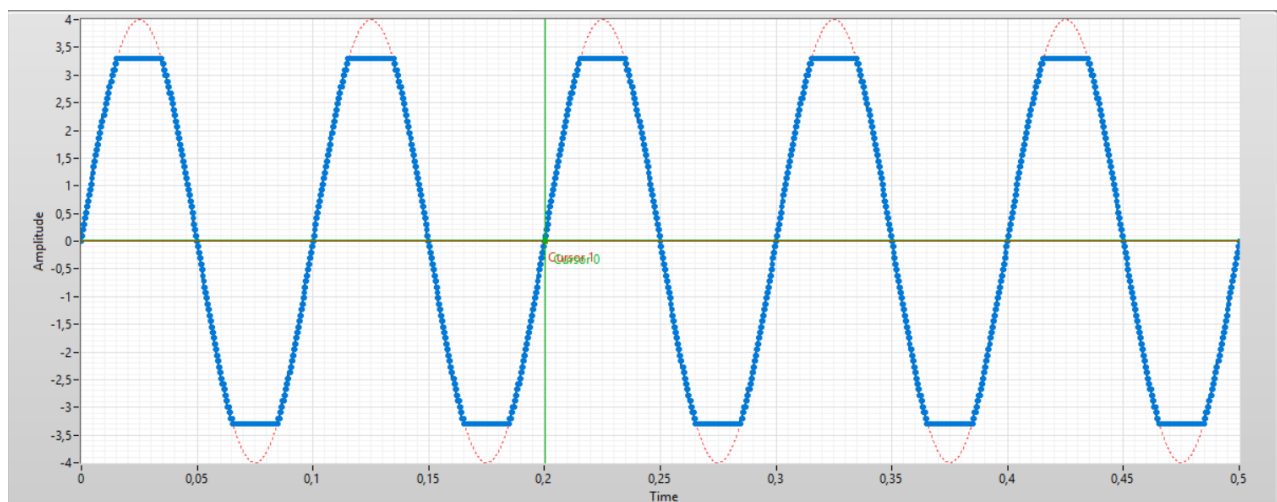


Fig. 1.3 – curvas do sinal do gerador (em vermelho) e do sinal amostrado (em azul)

1.2 Inicialização

Análise de Ondas com Conversor AD

Criada uma noção geral das funcionalidades do VI, vamos agora, de maneira breve, explicar as etapas gerais de funcionamento do VI. Visualize o Diagrama de Blocos deste VI (Ctrl+E). Na etapa de Inicialização do VI, são realizadas três tarefas:

- Inicialização do Sinal: Criação do sinal inicial que popula o VI quando aberto.
- Configuração do Pannel Frontal: Configura o Pannel frontal para que não mostre as barras de rolagem vertical e horizontal quando o VI estiver sendo executado.
- Inicialização do Dispositivo de Áudio: Configura os drivers de áudio para que as saídas e entradas do computador possam ser utilizadas pelo programa em LabVIEW.

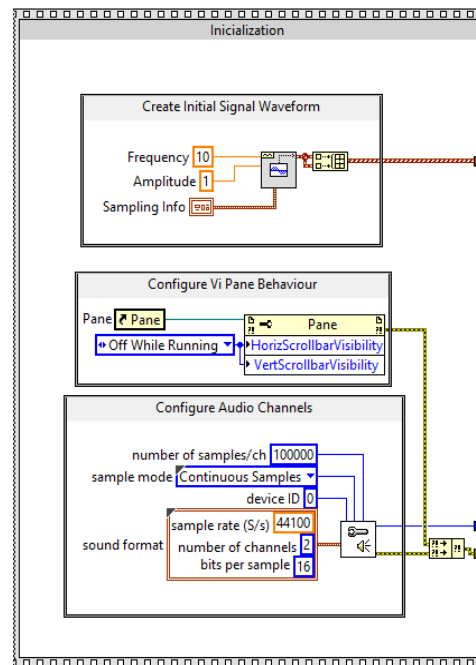


Fig. 1.4 – Etapas de Inicialização do VI

1.3 Processamento

Assim como um laço principal (“Main”) na linguagem C, nesta etapa são executadas as tarefas principais: “GUI Handler”, “Signal Visualization”, “Signal Analysis”, descritas abaixo:

- **GUI Handler**: Processa todas as ações realizadas pelo usuário no VI. Nela, são gerados os sinais que simulam um gerador de funções e um conversor AD, bem como são efetuadas todas as ações para importar arquivos e reproduzir os sinais em análise pelo computador.

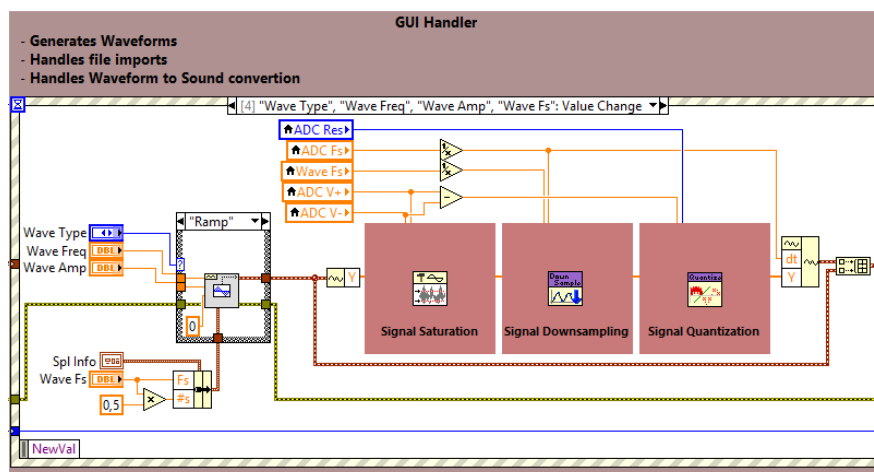


Fig. 1.5 – Etapas de Processamento do VI

– **Signal Visualization:** Colhe os sinais gerados pelo usuário e exibe-os no display principal. O gráfico “Signal Preview” mostra uma visão geral do sinal em questão, enquanto que o gráfico “Signal Overview” mostra o gráfico detalhado. Utilizaremos este gráfico para realizar as análises de inspeção visual.

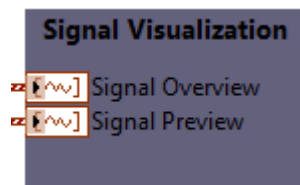


Fig. 1.6 – Signal Visualization

– **Signal Measurement & Analysis:** Esta tarefa será a qual teremos mais interesse nesse tutorial e será a qual desenvolveremos ao longo deste documento. Nela, os sinais gerados são colhidos e é realizada uma análise a fim de que sejam calculados parâmetros importantes para a caracterização de sinais de um ADC, tais como: período do sinal, taxa de aquisição e LSB.

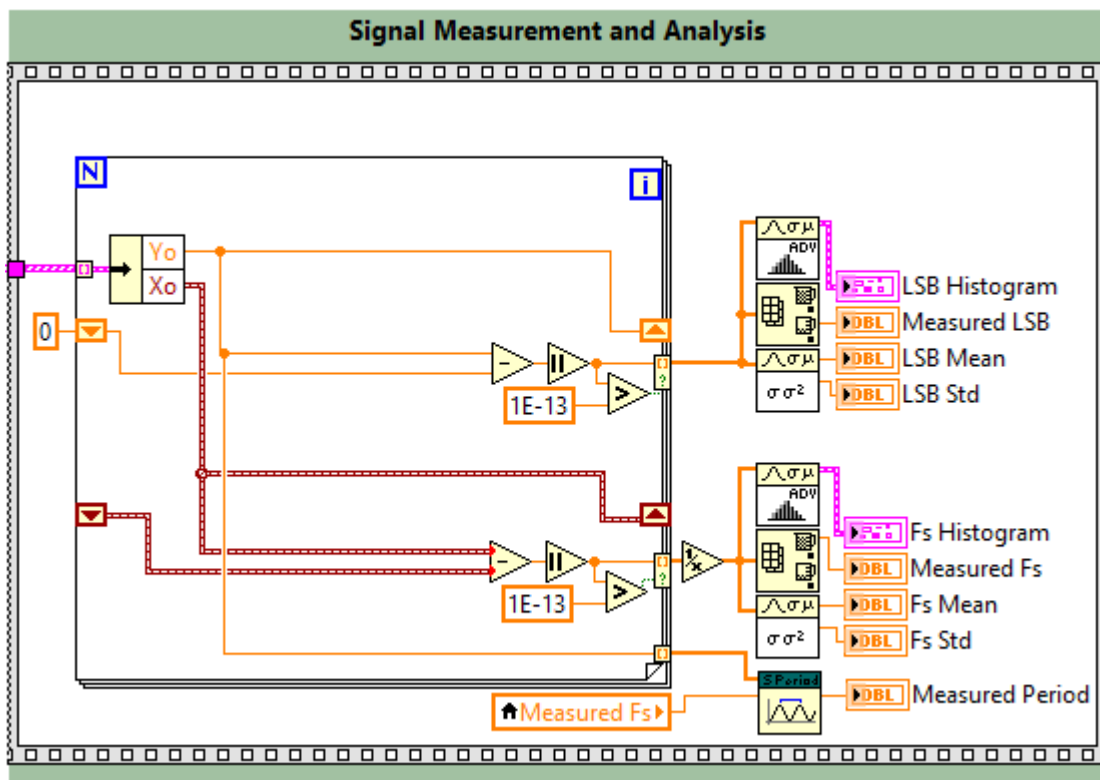


Fig. 1.7 – Etapa de Aquisição e Análise dos Sinais

É importante deixar claro que o intuito deste tutorial é a compreensão dos conceitos que caracterizam o sinal adquirido por um conversor AD, não os métodos numéricos utilizados. Esta discussão será aprofundada quando o projeto da disciplina for realizado.

1.4 Finalização

Por fim, ao final do programa os dispositivos de áudio previamente configurados são fechados e quaisquer erros que ocorram ao longo do VI são exibidos ao usuário.



Fig. 1.8 – Finalização do VI

2 Análise do Sinal por Inspeção Visual

Análise de Ondas com Conversor AD

Após ter sido realizada uma análise superficial do funcionamento do programa focaremos agora em como efetuar medições de um sinal. Inicialmente realizaremos uma análise por inspeção visual, isto é, utilizaremos ferramentas visuais como cursores e o zoom dos gráficos para que seja criada uma intuição dos conceitos que caracterizam um sinal no tempo.

Abra o VI **pr4.vi**, e execute o programa. Mude os controles para que tenhamos a seguinte configuração:

- | | | | |
|-----------------------|----------|--------------------------|----------|
| – Signal Type: | Sine | – Resolution: | 8 bits |
| – Signal Amplitude: | 4 V | – $V_{\text{supply}+}$: | +3.3 V |
| – Signal Frequency: | 10 Hz | – $V_{\text{supply}-}$: | -3.3 V |
| – Sampling Frequency: | 44100 Hz | – Sampling Frequency: | 44100 Hz |

A seguir, use as ferramentas da paleta do gráfico (como o da Fig. 2.1) para aplicarmos um zoom no gráfico. Gostaríamos de analisar uma faixa de valores que abranja cerca de 10 pontos adquiridos pelo ADC, como na Fig. 2.2:

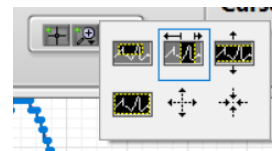


Fig. 2.1 – Paleta do Gráfico

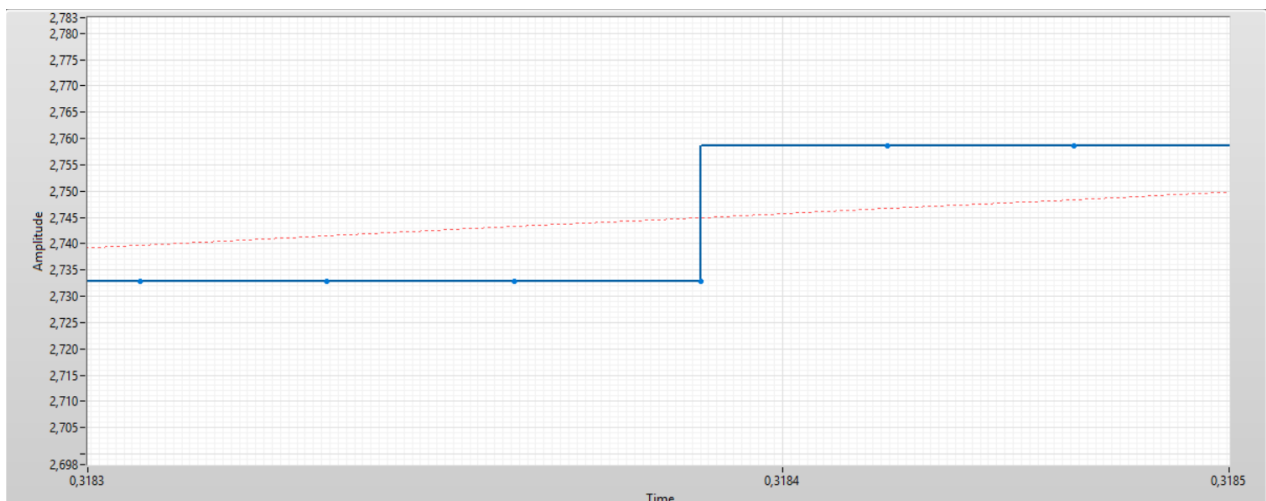


Fig. 2.2 – Zoom do sinal amostrado

Para realizarmos medições, faremos uso dos cursores. Clique com o botão direito do mouse em cada um dos cursores situados na paleta de cursores X Y (situada acima do seu gráfico) e selecione “Bring to Center” (Fig. 2.3), para trazer os dois cursores até o meio do gráfico.

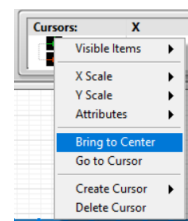


Fig. 2.3 – Funções da paleta de Cursores

Em seguida, selecione a opção de mover cursor na paleta do gráfico (Fig. 2.4). Clique com o botão esquerdo no centro de cada cursor e posicione-os em duas amostras adjacentes, como



Fig. 2.4 – Paleta do Gráfico, “Mover Cursor”

indicado na Fig. 2.5:

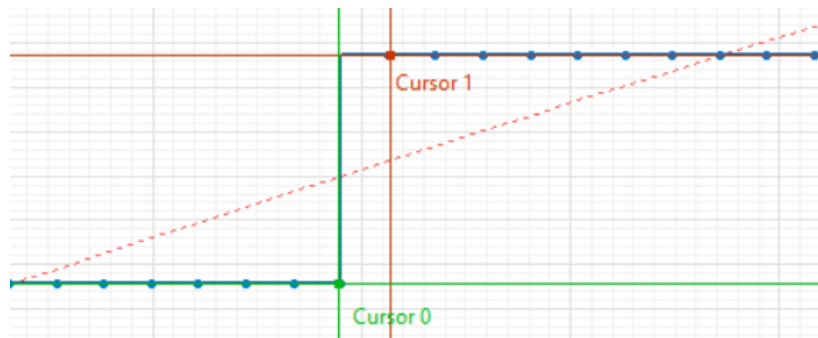


Fig. 2.5 – Posicionamento dos cursores para medir grandezas entre duas amostras adjacentes

2.1 Taxa de Amostragem

Feito o posicionamento correto dos cursores, vamos analisar a primeira das características temporais de um sinal, sua **taxa de amostragem**. Como foi visto em laboratório, um ADC é um circuito integrado cuja função é converter sinais reais analógicos em sinais digitais, que podem ser interpretados por um computador. A taxa de amostragem é a taxa na qual essa conversão é realizada, de modo a que corresponde ao intervalo entre as amostras digitais. Em nosso programa, as amostras são os pontos localizados na linha azul e a taxa de amostragem é a distância no eixo X entre os pontos adjacentes. Sendo assim, seu cálculo é realizado simplesmente pela subtração dos valores no eixo X das duas amostras.

No exemplo da Fig. 2.6 temos os valores de X e Y ao lado. Sua subtração corresponde a um intervalo de tempo de aproximadamente 22.705µs, uma taxa de aquisição de 44.044kHz.

Cursors:		
	X	Y
Cursor 0	0,3183899521	2,7330812474
Cursor 1	0,3184126566	2,7588449705

Fig. 2.6 – Destaque para dois pontos adjacentes

Quando comparado ao valor selecionado de 44.100 kHz, vemos que nossa medida possui um erro de aproximadamente 0.13%, associado majoritariamente à nossa habilidade (ou não) de posicionar o cursor corretamente na amostra.

Apesar de não ser o caso da simulação que estamos realizando, é importante comentar que quando for realizada a implementação de um sistema de aquisição de dados em um sistema real, como no sistema com microcontrolador que utilizarão no projeto extraclasse da disciplina, haverá erros associados à taxa de amostragem que não discutiremos aqui. Tais erros serão comentados brevemente em tutoriais auxiliares ao projeto.

2.2 Quantização

Outro parâmetro intrínseco a um sinal digital são seus erros de quantização. Apesar dos inúmeros erros relacionados ao processo de quantização de um conversor AD, vamos tratar apenas do erro relacionado à menor medida quantizável pelo conversor, ou LSB. Esta é uma característica que informa ao projetista qual a sensibilidade do ADC em relação a variações do sinal, ou seja, qual a menor variação de tensão que o conversor é capaz de detectar, e **depende apenas de dois fatores**: o fundo de escala e a resolução do ADC. A tensão relacionada ao LSB é obtida através da equação a seguir:

$$LSB[V] = \frac{\text{fundo de escala}}{2^{\text{resolução}}}$$

$$V_{LSB} = \frac{(V_{+supply} - V_{-supply})}{2^{res}} = \frac{(3.3 - (-3.3))}{2^8} \Rightarrow V_{LSB} = \frac{6.6}{256} \simeq 25.781 \text{ mV}$$

Apesar de o LSB ser um atributo teórico, que indica qual a **menor variação de tensão detectável** por um ADC, em termos práticos a **menor variação de tensão medida** pelo conversor, nem sempre corresponde ao valor do LSB.

As explicações das causas desse fenômeno e sua importância serão vistos com mais profundidade à frente, por hora, nossa análise por inspeção visual visa determinar apenas qual a menor variação de tensão medida em um sinal. Para isso, basta determinar qual a distância no eixo Y de duas amostras adjacentes. Tomando o exemplo da figura 1.13 e 1.14, temos uma variação de 25.764mV, que apresenta um erro de 0.65%.

2.3 Período

Por fim, o último aspecto temporal que será tratado neste tutorial será o período de um sinal. Esta medida, em essência, quantifica a repetitividade de um sinal, a menor parcela de tempo correspondente a um ciclo. Primeiro, vamos alterar a amplitude do sinal para 3 V e, para obtermos uma visão geral do sinal, vamos diminuir o zoom do gráfico (Fig. 2.7). Devemos ter um gráfico equivalente ao da Figura 2.8.



Fig. 2.7 – Paleta do Gráfico, “Restaurar Zoom”

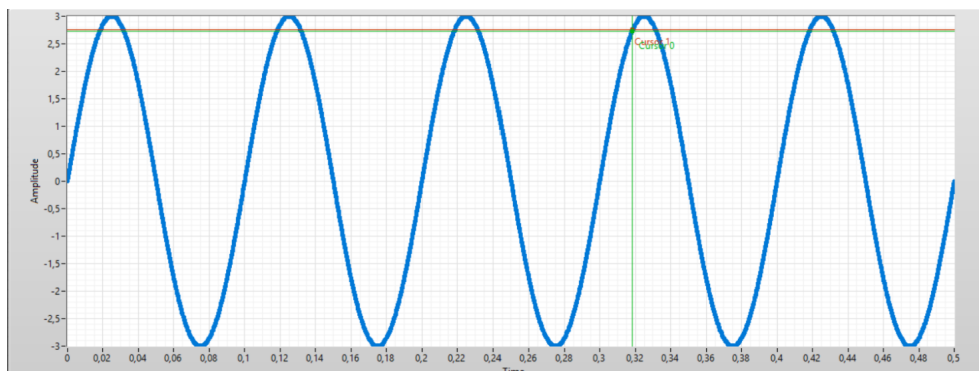


Fig. 2.8 – Visão geral do sinal sem zoom

Análise de Ondas com Conversor AD

Feito isso, basta medirmos dois pares de pontos em ciclos adjacentes que tenham os mesmos valores de tensão e derivadas para encontrarmos numericamente um valor para o período. Novamente, utilizaremos os cursores para tal. Tomando-se o exemplo indicado nas Fig. 2.9 e 2.10 como referência, percebe-se que um ciclo possui por volta de 100.28ms ou aproximadamente 9.97Hz, um valor com apenas 0.3% de erro em relação ao esperado.

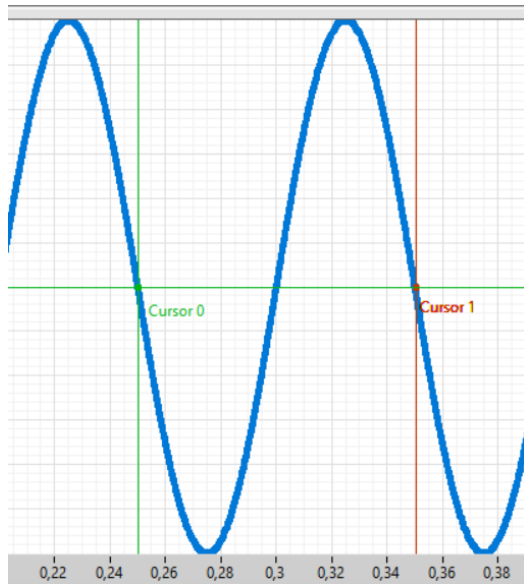


Fig. 2.9 – Cursores posicionados para medir o período de um ciclo de senoide

Fig. 2.10 – Destaque para os dois pontos selecionados

Cursors:	X	Y
Cursor 0	0,2502299908	-0,0071599045
Cursor 1	0,3505059798	-0,0071599045

Fig. 2.10 – Destaque para os dois pontos selecionados

2.4 Limitações da Inspeção Visual

Para sinais ideais o problema analisado é de certo modo trivial. Contudo, quando surgem problemas com o sinal analisado, como ruídos, ou quando o sinal é mais complexo, a análise pode se tornar uma tarefa cansativa e até mesmo impossível de ser realizada pelo olho humano.

Para tal será desenvolvido um programa em LabVIEW que possa automatizar as medições do sinal. Ao longo desse desenvolvimento, veremos que o problema de aquisição de dados torna-se uma tarefa delicada conforme o olhamos com um pouco mais de profundidade, as vezes contradizendo o senso comum.

3 Medições Automatizadas

Volte ao programa [pr4.vi] e abra seu diagrama de blocos. Na área designada “Signal Measurement & Analysis” construiremos os blocos necessários para análise do sinal.

Primeiramente crie um laço “For” na área (Functions Palette > Structures > For Loop) e conecte o sinal de saída “XY Data” para o laço como na figura abaixo:

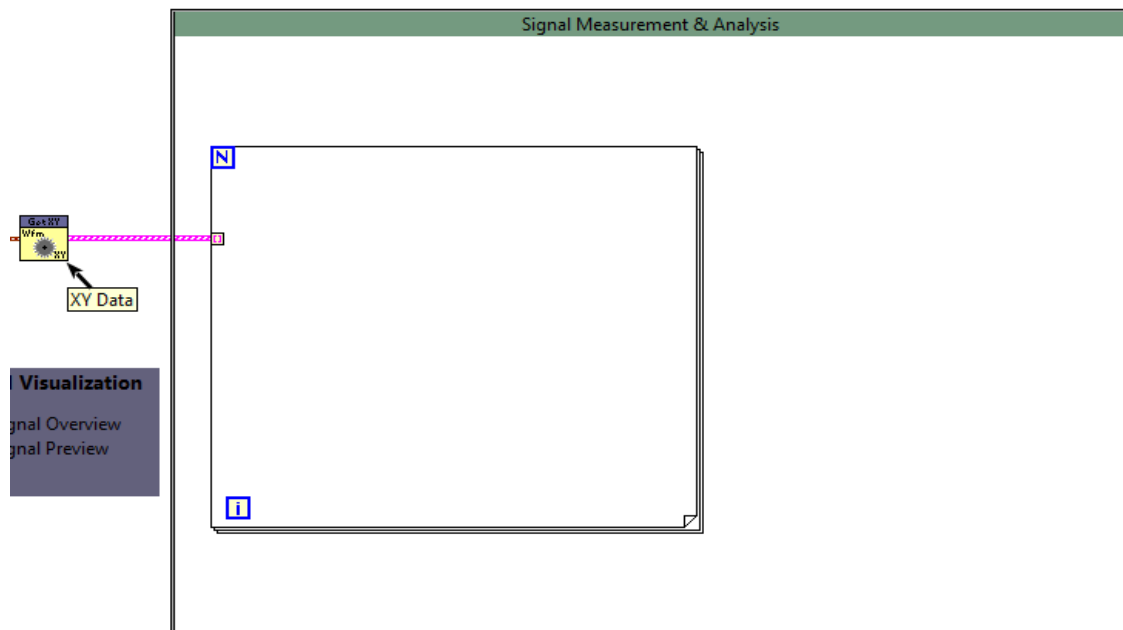


Fig. 3.1 – Laço “For” para análise dos elementos do sinal

Com isso temos acesso dentro do laço acesso a cada elemento do vetor XY individualmente, com os quais trabalharemos numericamente.

Para tal, devemos separar os pontos do vetor XY de modo a obter acesso separadamente a ambas parcelas X e Y. Para tal siga os passos abaixo:

- (i) – Dentro do laço, crie um bloco denominado “Unbundle by Name” (Functions Palette > Programming > Cluster Class & Variant > Unbundle by Name).
- (ii, iii) – Com o apontador do *Tools Palette*, clique e arraste na borda inferior do bloco criado para modificar o número de saídas para duas saídas
- (iv) – Conecte o sinal que entra no laço ao bloco criado.
- (v, vi) – Clique com o botão esquerdo na região Yo inferior para abrir o menu de opções. Selecione Xo.

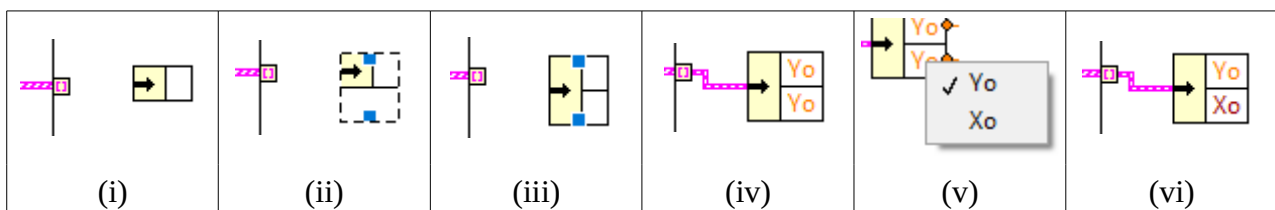


Fig. 3.2 – Passo a passo para separar os pontos do vetor XY

3.1 Taxa de Amostragem

Assim como realizado na etapa de análise por inspeção visual, a taxa de amostragem de um vetor é dada pelo menor intervalo temporal médio entre as amostras. Assim, para obter esta grandeza, basta subtrair as posições de elementos adjacentes do vetor X_o . Para fazer isso devemos realizar os seguintes passos, como ilustrado na Fig. 3.3:

- (i) – Conecte o sinal X_o na outra borda do laço
- (ii) – Clique no túnel com o botão direito e selecione “Replace with Shift Register”
- (iii) – Clique com o botão esquerdo na borda esquerda do laço para completar a criação do “Shift Register”

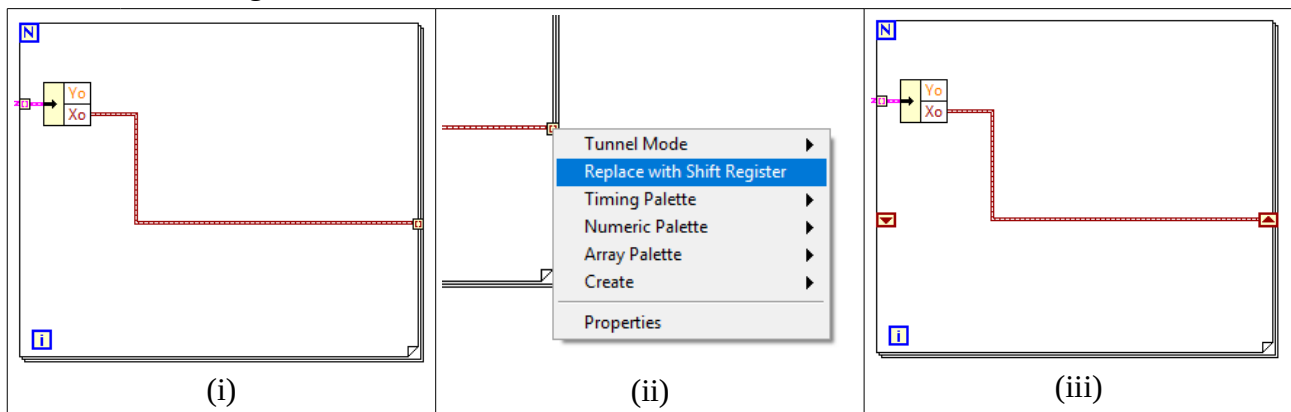


Fig. 3.3 – Conexão do sinal X_o na saída do bloco e criação do Shift Register

Terminada essa operação, teremos acesso ao elemento X_o do instante K , como também no instante $K-1$, como ilustrado na Fig. 3.4. Como K deve ser sempre > 0 , devemos também criar uma constante que inicializa o valor de $X_o[K-1]$ quando $K=0$. Além disso, segundo o que vimos no item 2.1, devemos subtrair $X_o[K]$ de $X_o[K-1]$. Para isso, efetue:

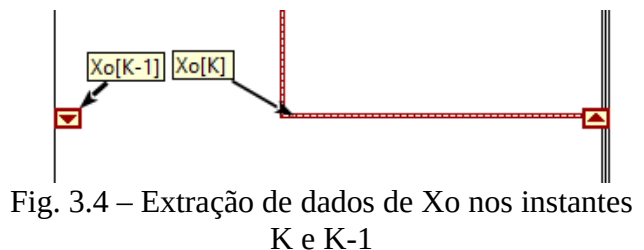


Fig. 3.4 – Extração de dados de X_o nos instantes K e $K-1$

- (i) – Com a ferramenta *wiring* do *Tools Palette*, clique com o botão direito do mouse no “Shift Register” esquerdo e selecione `create>constant`
- (ii) – Crie um bloco de subtração (*Functions Palette* > *Programming* > *Numeric* > *Subtract*)
- (iii) – Conecte o sinal X_o na função subtração

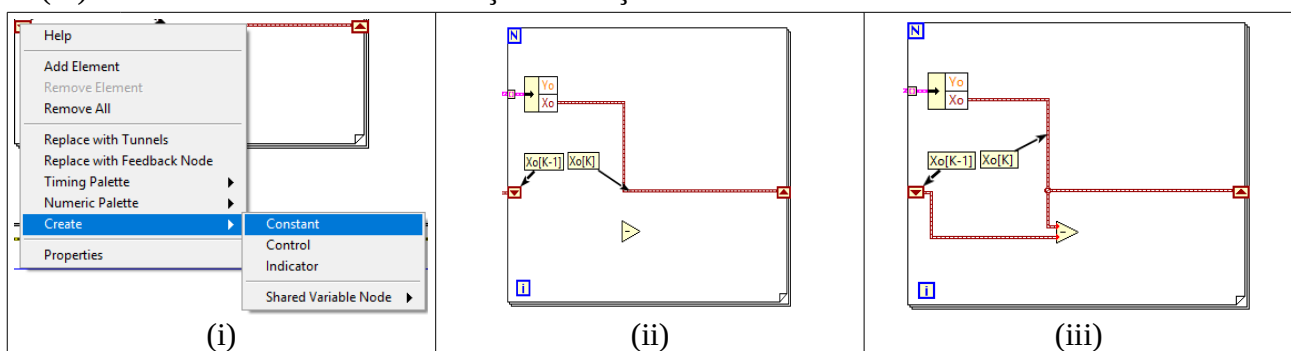


Fig. 3.5 – Passos para criar uma constante e efetuar a subtração entre $X_o[K]$ e $X_o[K-1]$

Para visualizarmos os resultados calculados da operação de subtração, devemos criar um vetor de saída do laço, coletar os dados desta operação e identificar qual é seu valor mínimo, o qual corresponderá a nossa taxa de amostragem. Como é mais compreensível a leitura e o dimensionamento dos valores em frequência, aplicaremos a fórmula $f=1/\Delta t$ para traduzir os valores do domínio do tempo para domínio da frequência.

Siga o procedimento a seguir para efetuar esta parte da programação, como ilustrado na Fig. 3.6:

- (i) – Conecte a saída do subtrator à borda direita do laço. Note que o túnel de saída estará em modo de indexação, de modo a que a saída do laço será também um vetor (Fig. 3.6.a).
- (ii) – Crie o bloco “Array Max & Min” (Function Palette > Programming > Array > Array Max & Min) e o bloco “Reciprocal” (Function Palette > Programming > Numeric > Reciprocal)
- (iii) – Conecte o vetor de saída do laço à entrada do primeiro bloco; a saída “min value” do bloco “Array Max & Min” ao bloco “1/x”. Clique na saída do bloco “Reciprocal” com o botão direito e selecione create>indicator.

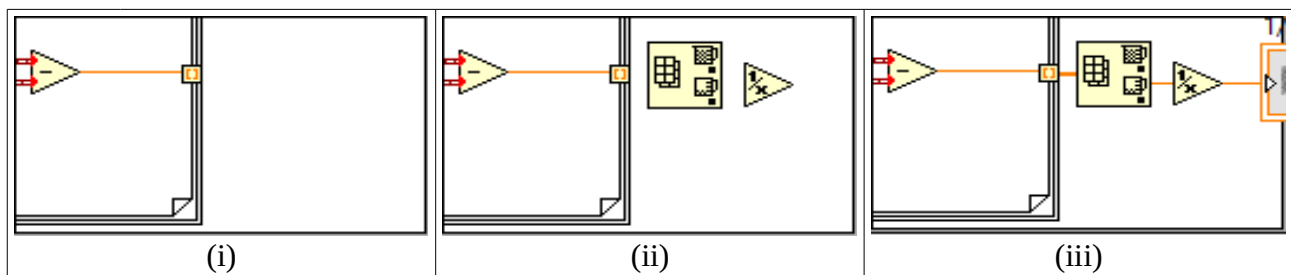


Fig. 3.6 – Passos para calcular a taxa de amostragem da conversão AD

Com isso, arraste o indicador criado no painel frontal (Ctrl+E) a um lugar adequado e execute o programa.

Note que o valor obtido não é exatamente o esperado (resultado equivalente ao da Fig. 3.7). Você sabe por que isso ocorreu?

Análise novamente o programa: perceba que na primeira amostra do vetor utilizado, subtraímos dois valores nulos, o que resulta em “0” como a resposta mínima dentro do vetor!

Para corrigir o programa e garantir que o comportamento do VI seja o esperado, deveremos adicionar ao vetor de saída do laço somente os valores não nulos. Para tal, efetue:

- (i) – Clique com o botão direito sobre o túnel de saída e selecione Tunnel Mode > Conditional
- (i) – Crie um bloco de comparação “Greater?” (Function Palette > Programming > Comparison > Greater?).
- (ii) – Conecte a subtração ao terminal superior do comparador e a saída do comparador ao terminal condicional do túnel de saída.
- (ii) – Crie uma constante no terminal inferior do bloco de comparação e mude seu valor para 1E-13

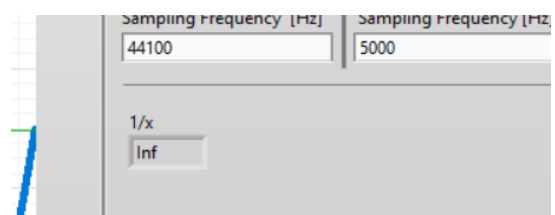


Fig. 3.7 – Resultado do cálculo da taxa de amostragem (1/x)

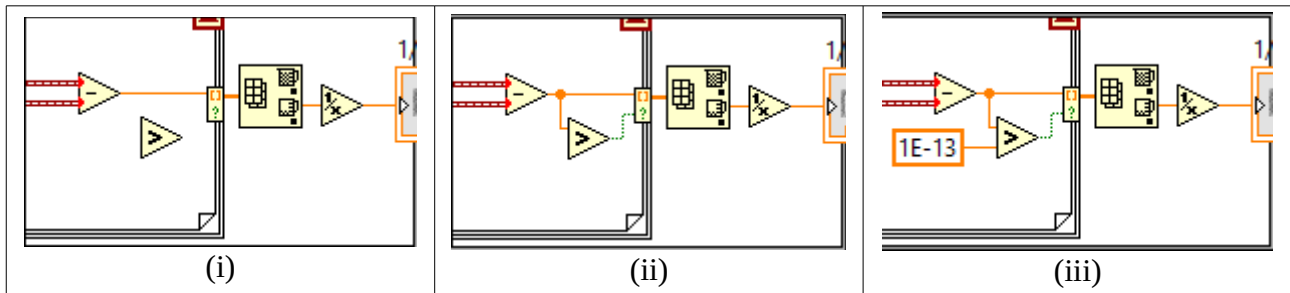


Fig. 3.8 – Criação de um túnel condicional para que somente os valores diferentes de zero sejam enviados para o bloco “Max & Min”

Pronto! Execute o programa e veja que o valor é o esperado, exatamente aquele selecionado na configuração do ADC. Tente mudar os valores e verifique que o programa funciona.

3.2 Quantização

Para que o programa também calcule o valor da menor medida quantizada ou **MMQ**, devemos seguir o mesmo raciocínio da programação do cálculo da taxa de amostragem. Inicialmente criaremos um par de “Shift Registers”, indicaremos um valor inicial, e efetuaremos a subtração dos valores de $Y_o[K]$ e $Y_o[K-1]$, de modo a que ao final tenhamos o diagrama de blocos equivalente ao da Fig. 3.9:

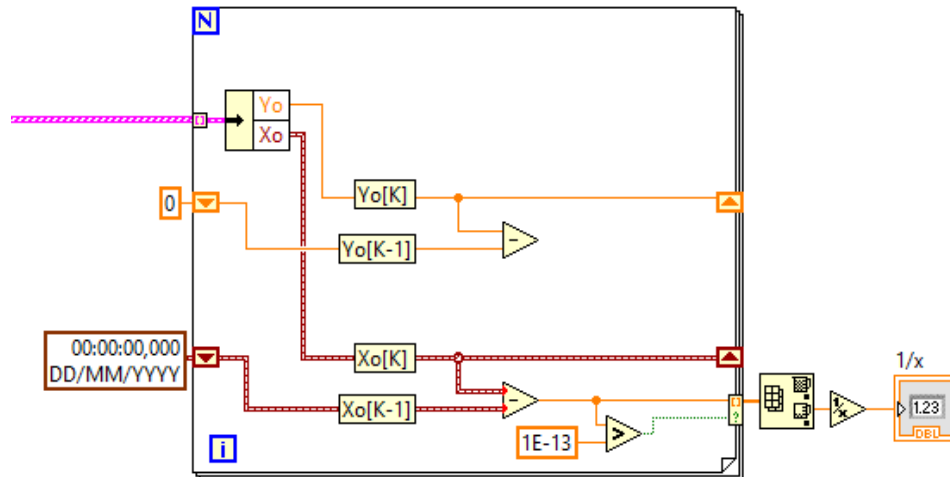


Fig. 3.9 – Programação parcial para o cálculo do valor do MMQ

Seguindo os passos mencionados, temos a subtração da posição em Y de dois elementos adjacentes do vetor de dados XY. Contudo, diferentemente dos elementos em X, os elementos de Y não estarão necessariamente ordenados, de modo a que a subtração poderá ser positiva ou negativa. No entanto, para o cálculo da MMQ estaremos apenas interessados no módulo da diferença entre dois elementos adjacentes. Sendo assim, efetue, como ilustrado na Fig. 3.10:

- (i) – Crie um bloco “Absolute Value” (Function Palette > Numeric > Absolute Value) e conecte a saída do subtrator à entrada desse bloco.
- (ii) – Conecte a saída do bloco “Absolute Value” à borda do laço e configure o túnel para que seja “condicional”, como efetuado anteriormente.
- (iii) – Crie um bloco comparador, como foi realizado no item anterior (para garantir que valores iguais a zero sejam desprezados), e conecte a saída do bloco “Absolute Value” ao terminal superior do comparador. A saída do comparador deve ser conectada ao terminal condicional e, finalmente, crie uma constante no terminal inferior do comparador com valor $1E-13$.

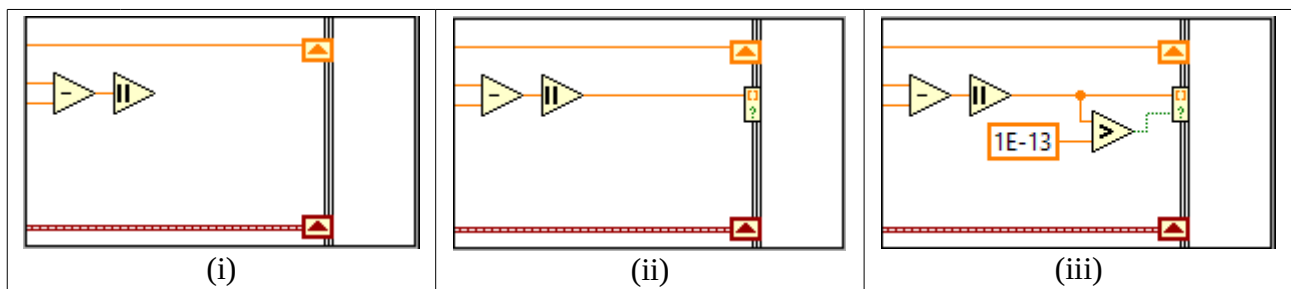


Fig. 3.10 – Programação para detecção de valor absoluto

Análise de Ondas com Conversor AD

Feito isso, basta encontrarmos qual o valor mínimo entre os elementos do vetor de saída Y, e este valor deverá corresponder ao MMQ esperado. Note que é necessário que seja encontrado o menor valor entre os valores do vetor Y. Essa ação é necessária pois dependendo das quedas ou subidas de tensão do sinal, podem ocorrer saltos de mais de 1 LSB dentro do intervalo de aquisição do ADC, como mostra a Fig. 3.11:

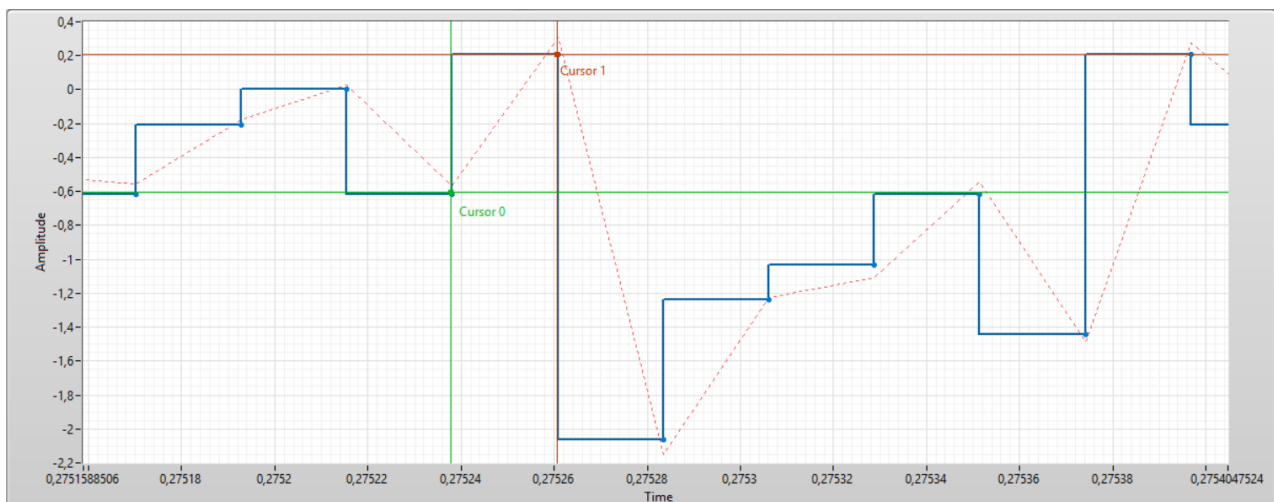


Fig. 3.11 – Distância entre dois pontos amostrados em tensão podem ser superiores a 1 LSB

Esse detalhe, aparentemente trivial, é muito importante e será tópico de discussão mais detalhado no final desse tutorial. Por enquanto realizaremos as seguintes ações, como ilustrado na Fig. 3.12:

- (i) – Crie, como realizado para a taxa de aquisição, um bloco “Array Max & Min” e conecte o vetor de saída do laço à sua entrada.
- (ii) – Clique com o botão direito no terminal “min value” e selecione create>indicator para criar um indicador próprio para o MMQ.
- (iii) – Para que possamos diferenciar o indicador do MMQ e o da Taxa de Amostragem, dê um duplo clique no nome do indicador e mude-o para um nome apropriado.

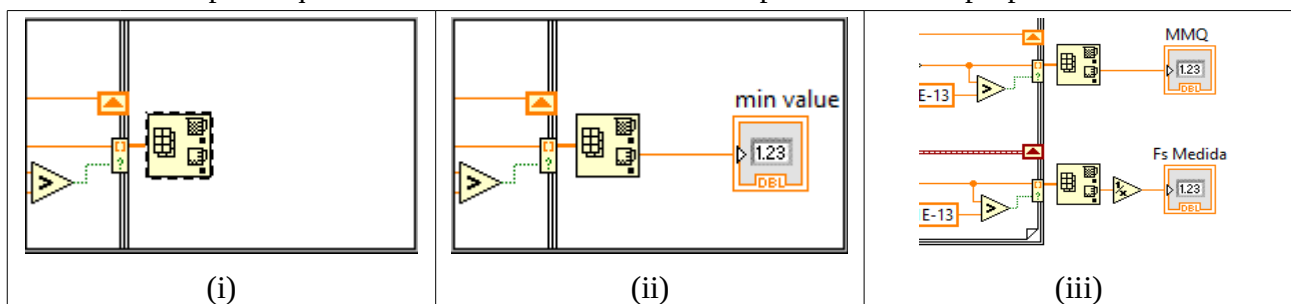


Fig. 3.12 – Programação para popular o vetor “Max & Min” que identificará o valor do MMQ

Pronto! Execute o programa e veja que a menor variação de tensão medida muda de acordo com a variação tanto do fundo de escala quanto da resolução do ADC. Teste o programa com algumas das ondas que podem ser geradas e perceba o comportamento do programa!

3.3 Período

Como último passo para completarmos nosso programa atacaremos o problema do período. O modo como desenvolveremos esse problema é quase idêntico ao modo como a função “Trigger” dos Osciloscópios funcionam. Do mesmo modo que fizemos na inspeção visual, nosso objetivo é adquirir dois pontos cujos valores sejam similares a um patamar pré-determinado, “Trigger Value”, e cujas derivadas sejam de mesmo sinal.

Para desenvolvermos a primeira parte deste algoritmo, é necessário que comparemos um par de pontos adjacentes a um patamar determinado. Para tal:

- (i) – Crie uma constante “DBL Numeric” (Function Palette > Numeric > DBL Numeric) e conecte-a à borda esquerda do laço “For”.
- (ii) – Conecte, como feito no item anterior, o sinal Y_0 à borda direita do laço, clique com o botão direito no túnel recém-criado, mude-o para um “Shift Register”. Por fim, crie uma constante no “Shift Register” da borda esquerda do laço. Para tal, clique com o botão direito nele e selecione create > constant.
- (iii) – Crie os blocos de comparação “Greater?” e “Less Or Equal?” (Function Palette > Comparison > ...). Conecte a constante “0” criada no passo (i) ao terminal superior de ambos os blocos criados. Conecte o sinal $Y_0[K]$ na entrada inferior do bloco “Greater?” e o “Shift Register” da borda esquerda à entrada inferior do bloco “Less Or Equal?”.

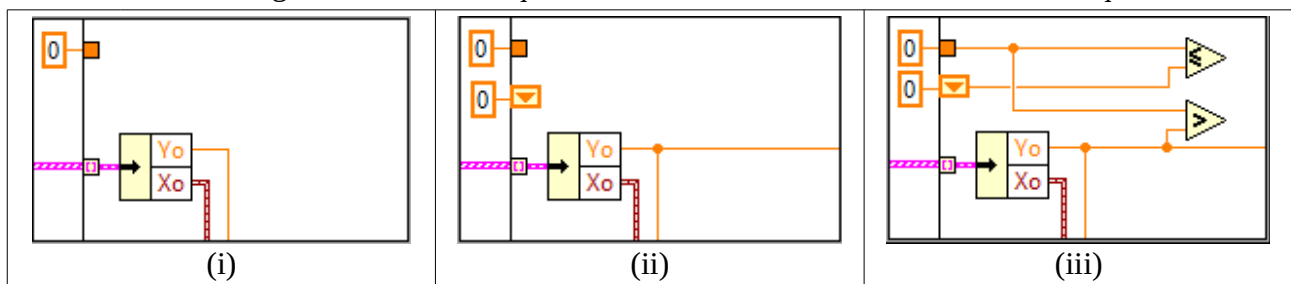


Fig. 3.13 – Programação da Função “Trigger”

Feito isso, nosso programa compara a amostra atual e a anterior a uma constante, no caso “0”, verificando se a amostra atual é maior do que “0” e se a amostra anterior é menor ou igual a “0”. Satisfeitas as duas condições temos um ponto que cruza o valor 0 de derivada é positiva. Gostaríamos que fossem adicionadas ao vetor de saída somente amostras que cumpram ambos condicionais acima. Deste modo:

- (i) - Conecte o sinal X_0 à borda direita do laço e modifique o túnel para que seja condicional.
- (ii) - Crie um bloco “And” (na aba “Boolean”), conecte os dois blocos de comparação previamente criados ao bloco “And” e, por fim, conecte a saída do bloco “And” ao terminal condicional

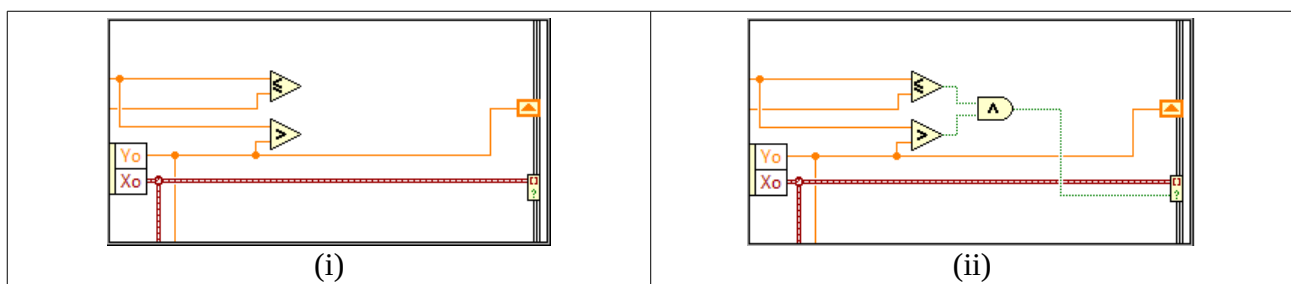


Fig. 3.14 – Criação do vetor de amostras “Rising Edge”

O vetor de saída será composto apenas por amostras que cruzam o eixo Y no valor desejado (no nosso caso é o valor “0”), e que possuam derivada positiva, “Rising Edge”. Agora, para obtermos o período do sinal, gostaríamos de determinar o intervalo entre as amostras do vetor recém-criado. Para esse fim, criaremos agora dois vetores de dados deslocados de uma amostra, e aplicaremos a operação de subtração neles para obter um vetor de intervalos de tempo. Do vetor resultante será extraído o intervalo de tempo de maior valor, que corresponderá ao período do sinal.

Para realizarmos essa operação efetuaremos os seguintes passos, como na Fig. 3.15:

- (i) – Crie o bloco “Array Size” (na aba “Array”) e o bloco “Decrement” (na aba “Numeric”), conecte a saída do laço ao primeiro bloco e a saída deste ao bloco “Decrement”. Com isso, dado um vetor de tamanho N, o valor na saída do bloco “Decrement” corresponderá ao valor N-1.
- (ii) – Crie dois blocos “Array Subset” e um bloco “Subtract” (na aba “Numeric”). Conecte o vetor de saída do laço às entradas “array” dos dois blocos “Array Subset”. Conecte a saída do bloco “Decrement” à entrada “length” do bloco inferior para que sejam selecionadas as amostras no intervalo [0, N-1]. Crie uma constante de valor 1 no terminal “index” do bloco superior para que sejam selecionadas as amostras no intervalo [1, N]. Por fim, conecte as saídas dos blocos “Array Subset” ao bloco de subtração.
- (iii) – Crie um bloco “Array Max & Min” (na aba “Array”) e um bloco “Reciprocal” (na aba “Numeric”). Conecte a saída do subtrator ao bloco “Array Max & Min”.

Conecte a saída “max value” desse bloco ao bloco “Reciprocal” e, por fim, crie um indicador para que possamos visualizar o valor obtido para o período do sinal.

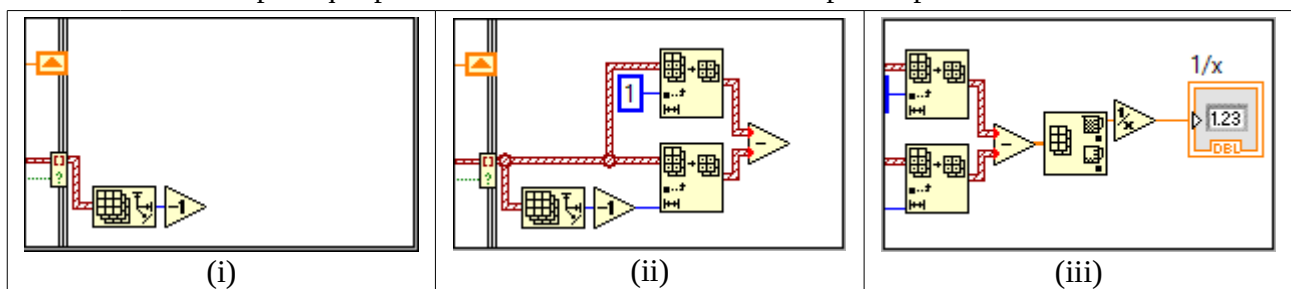


Fig. 3.15 – Programação para identificação do período

Perceba que no bloco superior é criado um vetor que abrange os valores [1, N], contendo (N-1) amostras, enquanto que o bloco inferior contém os valores [0, N-1] de mesmo tamanho.

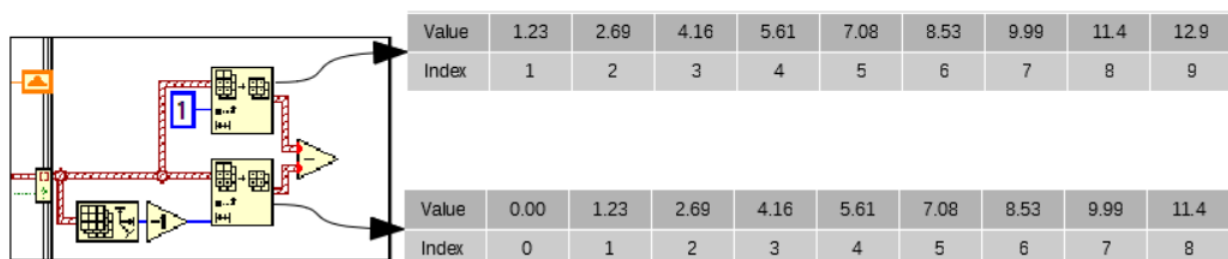


Fig. 3.16 – Diagrama dos blocos “Array Subset” e exemplo de saída

Análise de Ondas com Conversor AD

Com isso, todas as operações necessárias para o cálculo do período de um sinal foram completadas e nosso programa está pronto para ser testado. Mude o nome do indicador e no painel frontal arraste-o para uma posição adequada. Execute o programa e veja seu desempenho!

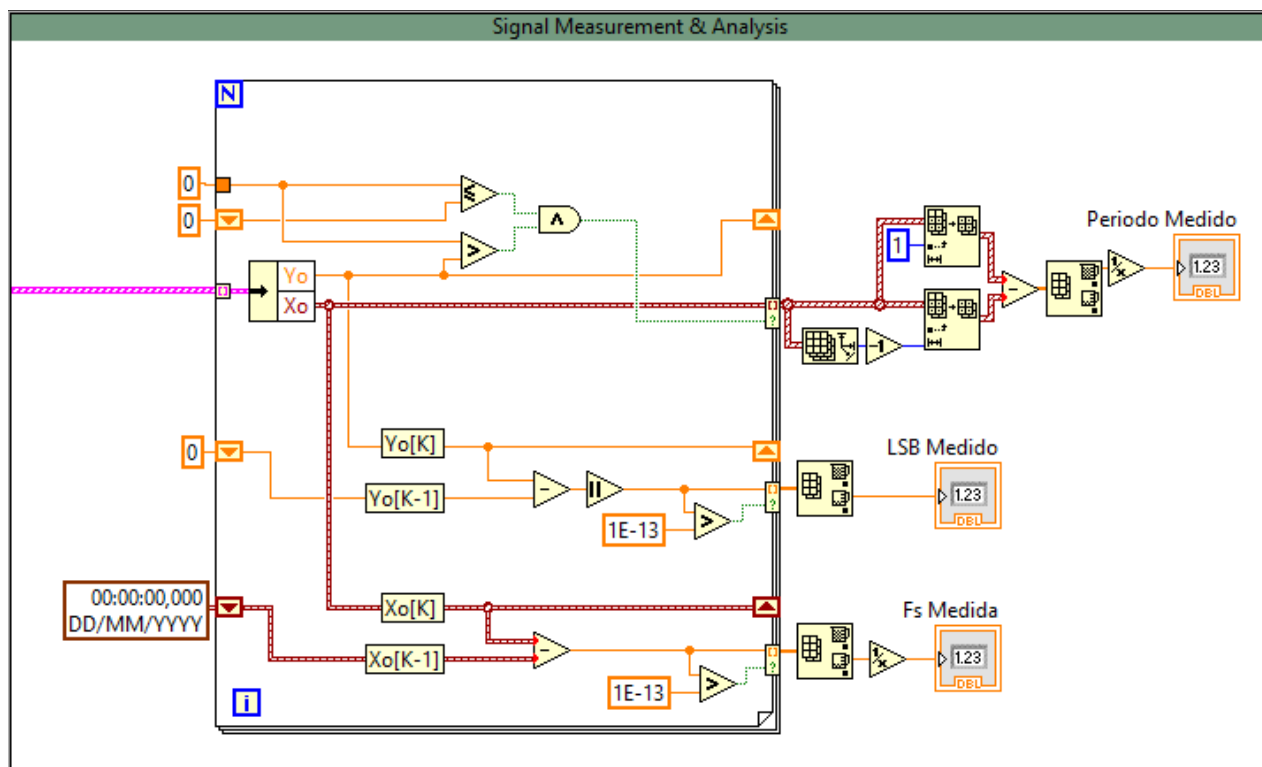


Fig. 3.17 – Programação completa

4 Conclusão do Programa

Até agora todos os sinais analisados consistiam em modelos matemáticos ideais, ondas senoidais, quadradas, trapezoidais e triangulares perfeitas. Tal direcionamento foi dado a fim de se introduzir certas noções e conceitos.

No mundo real, os sinais adquiridos serão dotados de ruídos e distorções das mais diversas. Na conclusão do tutorial vamos verificar qual a aparência de um sinal advindo de uma nota musical e qual o comportamento de nosso programa frente a um sinal mais complexo.

Execute o VI até agora construído e certifique-se de está funcionando normalmente. Configure o sinal criado e o ADC para que tenham os seguintes valores:

– Signal Type:	Sine	– Resolution:	8 bits
– Signal Amplitude:	0.1 V	– $V_{\text{supply}+}$:	+0.1 V
– Signal Frequency:	440 Hz	– $V_{\text{supply}-}$:	-0.1 V
– Sampling Frequency:	44100 Hz	– Sampling Frequency:	44100 Hz

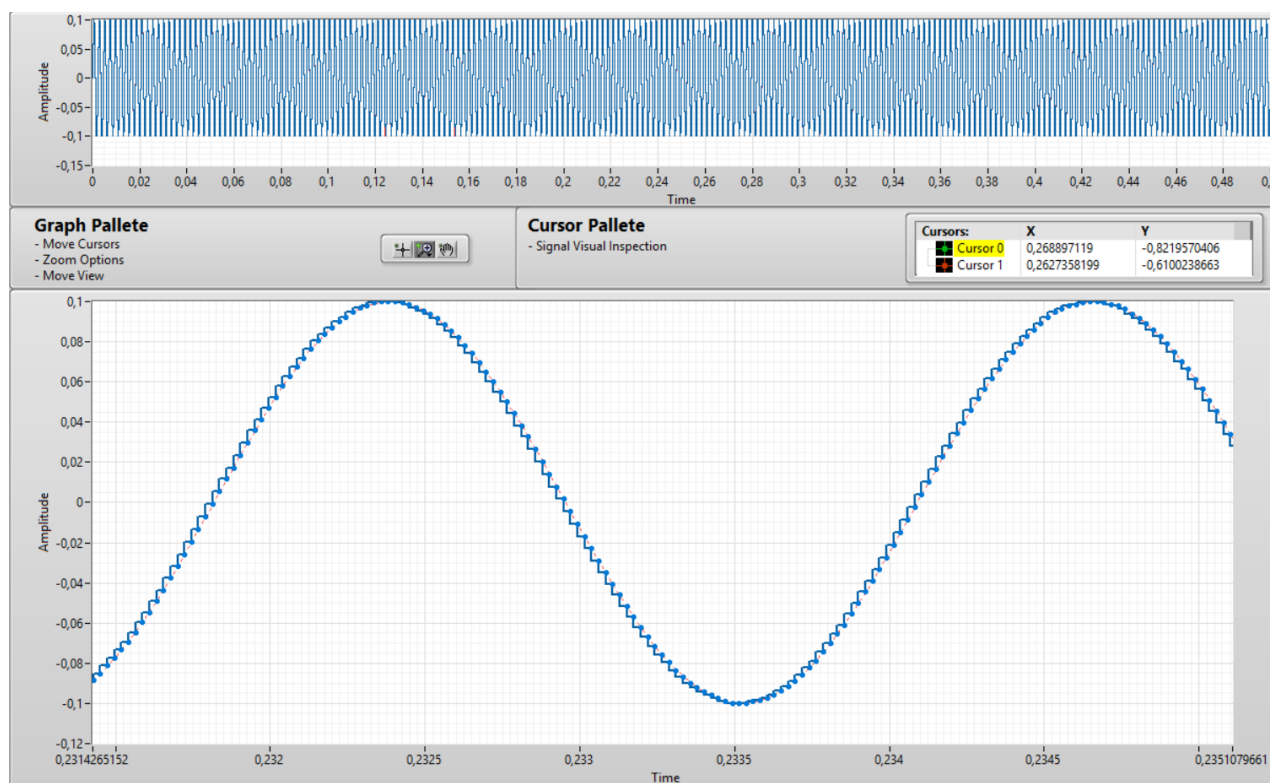


Fig. 4.1 – Verificação de que o programa funciona corretamente

Feito isso, vamos testar se o dispositivo de áudio foi inicializado corretamente. Dê um clique com o botão esquerdo do mouse no botão “Play” e tente ouvir a nota produzida. Mude individualmente parâmetros como a resolução do ADC, sua taxa de amostragem e a amplitude do sinal para ver quais os efeitos no som resultante.

Análise de Ondas com Conversor AD

Volte os valores dos controles aos mencionados previamente e aperte o botão “Import File” e selecione o arquivo “Note Violin 1”, nele temos um trecho de uma nota produzida por um violino.

Selecione um intervalo do sinal importado que tenha mais ou menos 20ms. Nessa faixa de valores deveremos ter um gráfico como o seguinte:

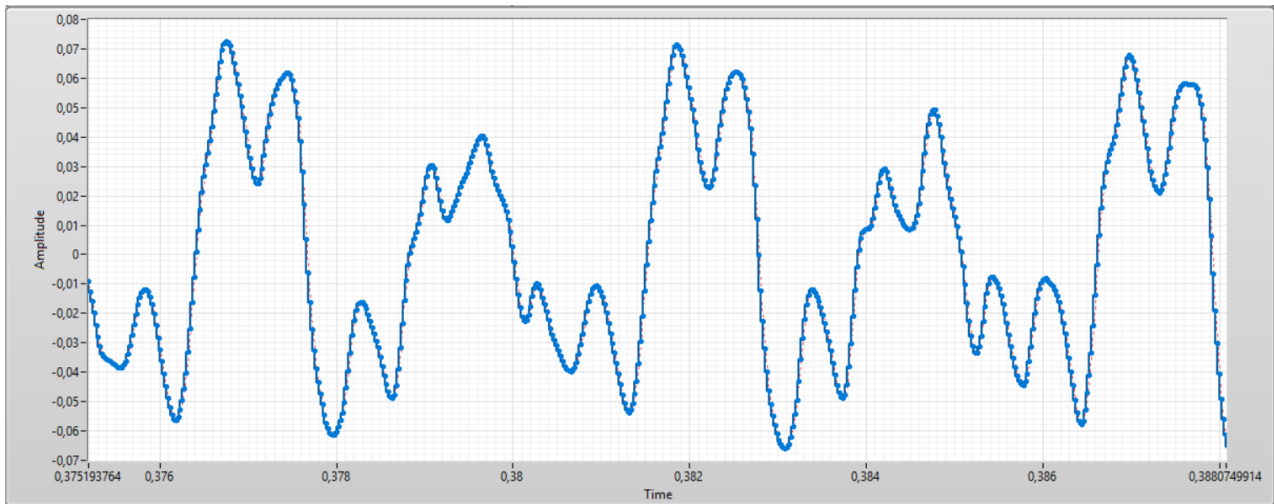


Fig. 4.2 – Sinal adquirido de um violino

Perceba que nosso ADC tem a capacidade de adquirir o sinal com destreza. Primeiramente note que o sinal não é uma senoide, mas sim um **conjunto de senoides somadas**. Veremos com mais calma essa noção na **experiência 3 do laboratório da disciplina PSI3214**. Por enquanto, basta que seja reconhecida a limitação clara do nosso algoritmo para identificação do período.

Em um segundo momento, aperte o botão “Play” novamente e preste atenção no som produzido. Repare que de fato, o conjunto aparentemente arbitrário de pontos de tensão formam um som de um violino.

Como visto anteriormente, o botão “Play” reproduz o sinal em azul, o qual nós podemos modificar por meio dos parâmetros do ADC.

A seguir faremos algumas modificações nos parâmetros que definem nosso ADC a fim de averiguar seus impactos em um sinal real:

1. Em uma situação de mundo real, **não é plausível fornecer a um ADC tensões de alimentação da ordem de 100mV**. Perceba que a tensão de fundo de escala é apenas uma faixa de valores a qual o conversor AD usa para comparar com o sinal de entrada. Perceba que essa dependência implica em que quaisquer variações da tensão do fundo de escala são propagadas para a leitura do sinal, resultando em um valor digital que não corresponde ao valor analógico na porta de entrada do ADC. Dito isso, quanto menor for o valor da tensão de alimentação, menor fica a razão entre tensão de alimentação e ruído, ou seja, maiores serão os erros causados por ruídos de tensões parasitárias. Sendo assim, mude-o para valores mais adequados, da ordem de 5 ~ 12 V.

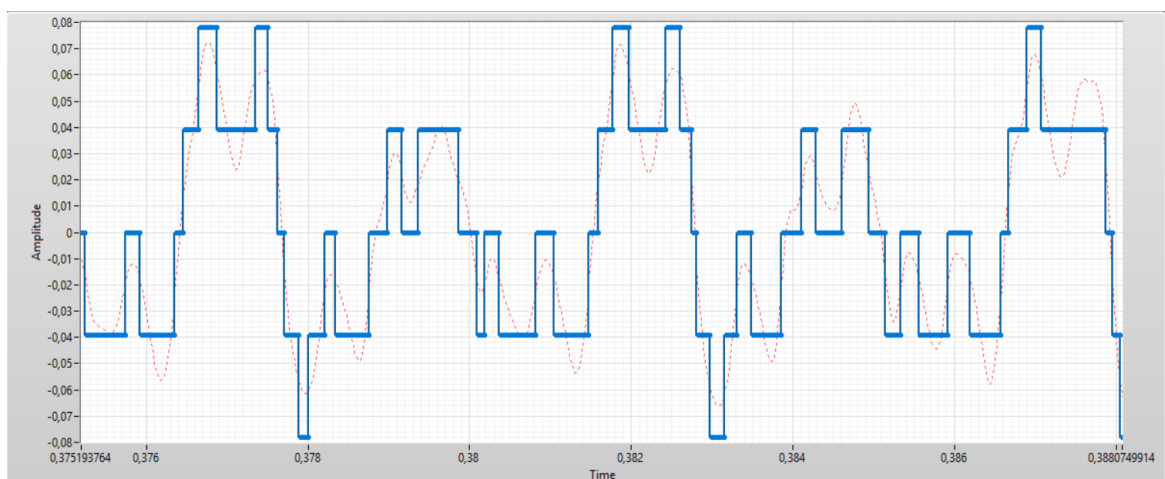


Fig. 4.3 – Sinal distorcido por conta de LSB muito grande

Perceba que ao mudarmos a tensão do fundo de escala, nosso LSB aumenta de forma considerável. De modo a que o sinal, antes representado de forma adequada, mostra agora muitas distorções, Fig. 4.3.

Feito o procedimento acima, percebemos porque são necessários amplificadores para sinais de baixas tensões. **É preferível que um sinal seja amplificado e depois analisado por um ADC cujas tensões do fundo de escala são adequadas do que realizar uma conversão AD com tensões de alimentação da ordem de ruídos parasitários.**

2. Em seguida, vamos verificar qual o comportamento de nosso algoritmo para medição de MMQ em relação aos valores teóricos do LSB. Note que o LSB é uma medida fixa, com equação definida. A depender do sinal medido pelo ADC, a menor variação percebida pelo ADC, o MMQ, nem sempre corresponde ao LSB, e essa informação pode vir a ser útil ao projetista do sistema de aquisição. Levando esse fato em consideração, temos duas situações em que o MMQ não tem mesmo valor do LSB.

(a) **Resoluções Baixas:** Nesta situação, das duas a mais fácil de se compreender, o LSB é maior do que a amplitude do sinal. De modo a que o sinal adquirido não é “percebido” pelo ADC, como na Fig. 4.4.

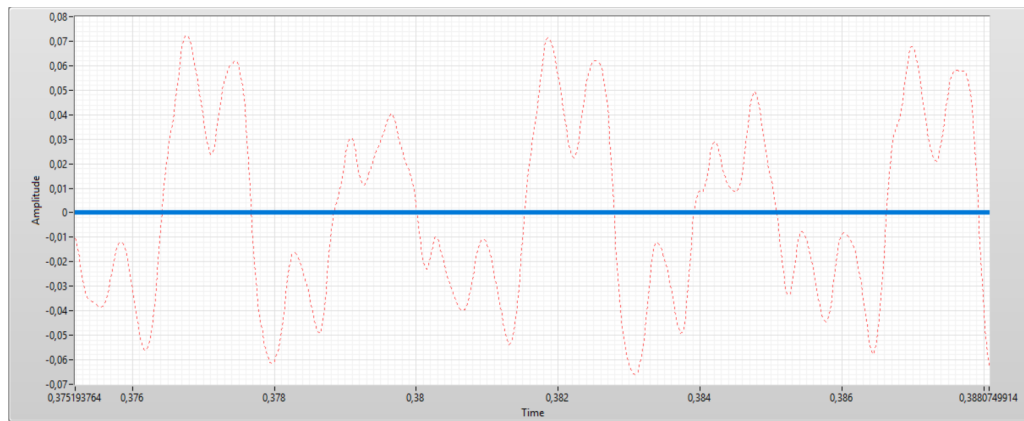


Fig. 4.4 – LSB maior do que amplitude do sinal

Nessa situação, o MMQ é nulo, pois o ADC não foi capaz de identificar nenhuma variação do sinal.

- (b) **Resoluções Altas:** Até o momento, apenas foi destacada a importância do LSB para a quantização do sinal. Contudo, esta é apenas uma simplificação de um processo muito mais complicado. Em realidade, a habilidade de quantização de um ADC, ou seja, sua capacidade de identificar patamares definidos no sinal é impactada não só pelo LSB, como também pela taxa de aquisição do ADC, pela frequência do sinal em questão e de sua fase.

De modo mais “intuitivo”, podemos imaginar a seguinte situação: “Imagine um sinal cujo nível de tensão no tempo t_A é V_A , e no tempo t_B é V_B . Caso a taxa de aquisição seja suficientemente maior do que a frequência do sinal, a distância entre V_B e V_A deverá ser 0 ou 1xLSB (Fig. 4.5 a). Contudo, caso a taxa de variação local do sinal seja muito próxima, ou maior, do que a taxa de aquisição do ADC, a tensão percebida pelo conversor será maior do que 1xLSB (Fig. 4.5 b). Do mesmo modo, a fase do sinal adquirido pode levar às mesmas consequências (Fig. 4.5 c)”

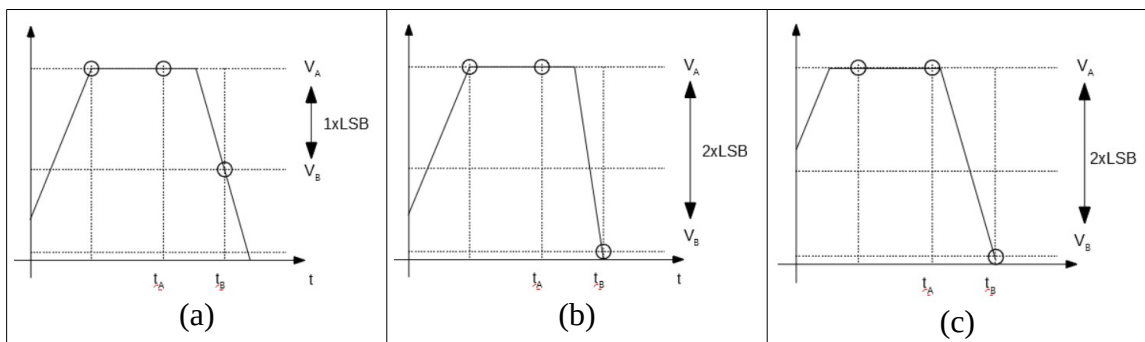


Fig. 4.5 – Influência da frequência e fase do sinal nos pontos de aquisição

Nesta situação, o aumento da resolução não gera resultados expressivos na aquisição do sinal, de modo a que a variação mínima percebida pelo ADC torna-se um múltiplo inteiro do LSB, Fig 4.6.

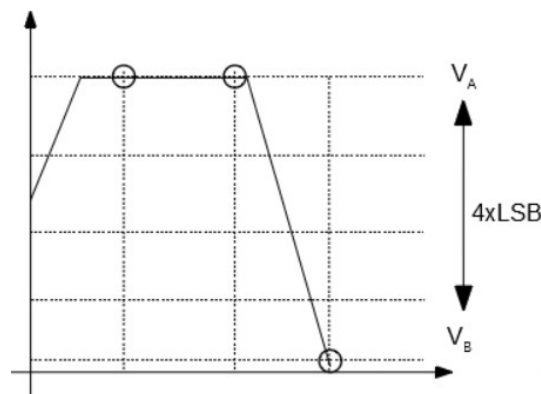


Fig. 4.6 – Aumento da resolução e sua consequência

Para verificar essa situação, aumente a resolução até que o LSB medido não mude de valor. No caso do sinal importado, essa saturação deverá ocorrer de 23bits para 24bits. Conforme aumentamos a resolução a razão entre variação mínima do sinal e LSB cresce exponencialmente.

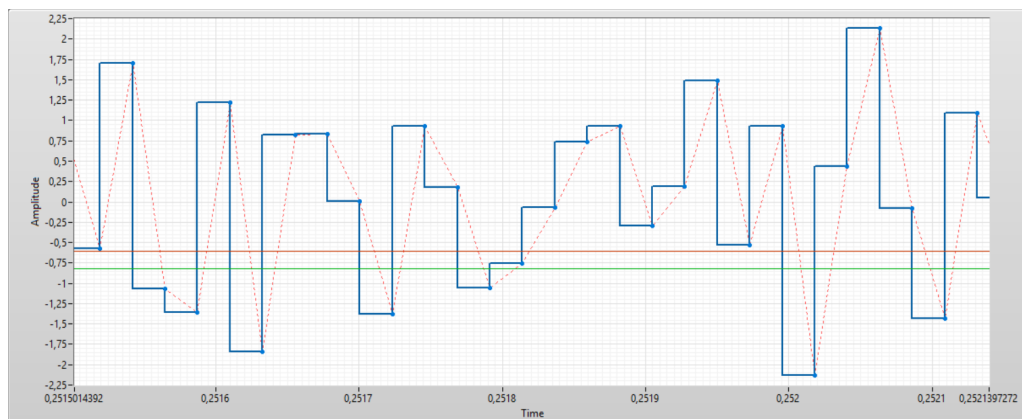


Fig. 4.5 – Sinal cuja menor variação é maior do que o LSB

A explicação das causas desse fenômeno, bem como a consideração de suas influências no valor dos dados adquiridos fogem ao escopo do tutorial. Novamente, este conceito foi introduzido, assim como outros neste tutorial como contextualizador. Neste caso, nos importamos em deixar claro que a escolha de um conversor AD com a maior resolução possível nem sempre é a escolha mais sábia.

Como um experimento para que essa última consideração fique mais clara, realize o seguinte procedimento:

– Mude os valores dos controles para os seguintes:

– Signal Type:	White Noise	– Resolution:	8 bits
– Signal Amplitude:	1 V	– $V_{\text{supply}+}$:	+5 V
– Signal Frequency:	250 Hz	– $V_{\text{supply}-}$:	-5 V
– Sampling Frequency:	44100 Hz	– Sampling Frequency:	1000 Hz

– Perceba que o MMQ, ou seja, a menor variação percebida no sinal discretizado coincide com o LSB calculado. Aumente gradualmente a resolução do ADC até que o MMQ não seja mais igual ao calculado, em torno de 10bits. Aumente a Resolução para 16bits e meça qual a razão entre LSB medido e LSB calculado.

– Aumente agora o valor da frequência de amostragem do ADC para 44100 Hz e veja o que ocorre com a razão calculada.

Feitas as considerações acima, terminamos todos os tópicos do tutorial!

Análise de Ondas com Conversor AD

4.1 Exercício

Abra o arquivo [**ex2.vi**] e execute-o, siga as instruções da tela e complete o exercício.

4.2 Exercícios Complementares