

Relatório Final

Etapa 2

Laboratório de Instrumentação Elétrica
PSI 3214

Prof. Marcelo N.P. Carreño e Profa. Inés Pereyra
Turma 6



21/11/22

Antonio Oliveira Pedroso da Cruz
Daniel Martins Cabanas
João Pedro Lima Affonso de Carvalho

NºUSP: 11798787
NºUSP: 12486742
NºUSP: 11260846

INTRODUÇÃO

Processamento de sinais elétricos é uma atividade essencial para a vida humana. Desde de eletrocardiogramas, mapeamento da temperatura em nossa crosta terrestre até a afinação de instrumentos musicais se fundamentam na análise dos sinais em particular na frequência desses sinais e seu respectivo espectro de frequências. Além da capacidade computacional requerida para essa tarefa, existem vários outros outros empecilhos que dificultam essa análise

Com isso em mente, o algoritmo capaz de identificar as frequências que compõem um sinal é a *Transformada de Fourier*. Desenvolvido por *Jean-Baptiste Joseph Fourier* e anunciado pela primeira vez em 1807 (HIGHLIGHTS, 2016) [1], o método é capaz de representar um sinal no domínio das frequências como representado na *Figura 1*.

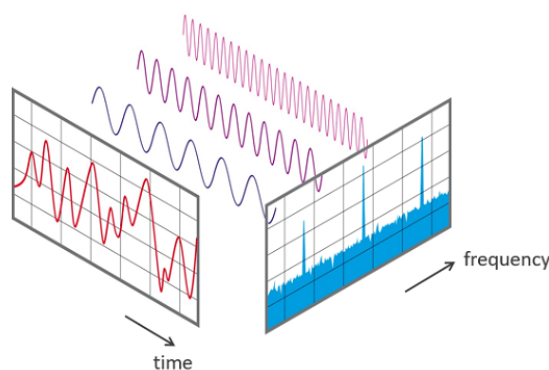


Figura 1 - Transformada de Fourier
Fonte: FAST

Ou seja, trata-se de uma aplicação que transforma um sinal no domínio do tempo para um sinal no domínio das frequências.

A *Transformada de Fourier* é obtida através das séries de Fourier quando fazemos o limite do período tender ao infinito, o que só funciona para sinais contínuos. Como na maioria dos casos em sistemas digitais, os sinais fornecidos, após a conversão AD são de natureza discreta. Para esse caso deve-se utilizar uma versão adaptada do modelo de Fourier: a *Transformada Rápida de Fourier* [2], que se baseia na forma discreta da *Transformada de Fourier*. Com isso, é possível obter as frequências que compõem o sinal com o menor atraso possível, sendo aplicada na maior parte dos problemas que precisam identificar frequências no tempo mais rápido possível.

O projeto que será apresentado ao longo desse relatório se baseia principalmente nesse conceito para processar os sinais adquiridos e cumprir os objetivos determinados.

OBJETIVOS

Para essa etapa do projeto, foram determinados os seguintes objetivos:

- Implementar uma VI usando o LabView para ler o sinal do arduino e representá-lo em função do tempo, bem como seu espectro de frequências, como foi feito na etapa anterior
- Implementar uma lógica no VI capaz de identificar uma sequência de notas musicais, modificando o circuito caso necessário

PROCEDIMENTOS

1) Circuito Utilizado

Assim como na primeira etapa, o circuito conta com um microfone acoplado a um sistema composto por um amplificador operacional com a finalidade de amplificar o sinal por ser extremamente fraco (como discutido no relatório 1), e condicioná-lo entre 0V e 5V.

Esse sinal analógico será transmitido a um arduino com um algoritmo capaz de converter esse sinal para um sinal digital (foi utilizado o código disponibilizado sem modificações). Além disso, o arduino estará conectado ao computador por um cabo USB e os dados serão enviados em tempo real para um programa em LabView capaz de apresentar esses dados na forma de um gráfico em função do tempo, seu espectro de frequências, e retornar qual a nota musical do som detectado pelo microfone.

As figuras abaixo mostram o esquema do circuito e o circuito montado:



Figura 2 - Circuito montado

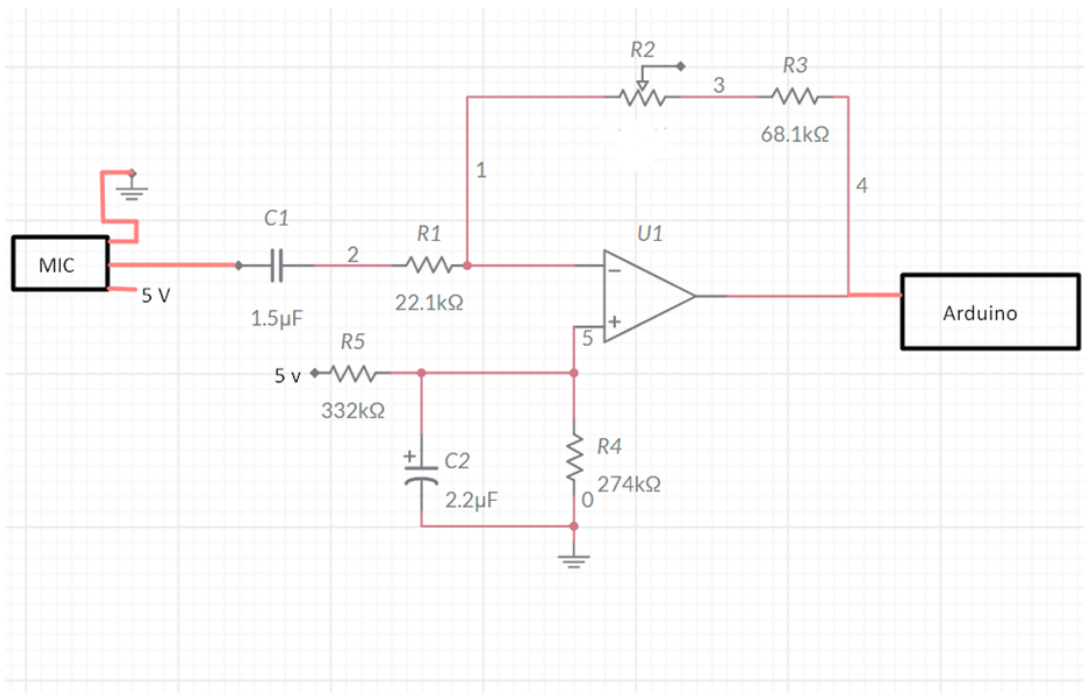


Figura 3 - Esquema do circuito

2) Virtual Instrument (VI do LabView) - Diagrama de blocos

Após o sinal ser amplificado e passar pelo arduino, ele é enviado ao VI implementado por meio de um cabo USB, onde é mostrado graficamente no domínio do tempo, é calculado o espectro de frequências e é feita a lógica de detecção das notas musicais.

Para exibir o sinal graficamente e calcular o seu espectro de frequências, o esquema utilizado foi idêntico ao da etapa anterior, com exceção de uma pequena modificação na lógica de processar o sinal vindo do arduino.

Para essa etapa, o que foi feito de diferente é que o sinal recebido pela VI é multiplicado apenas uma vez pela LSB (na etapa anterior o sinal estava, erroneamente, sendo multiplicado duas vezes) para discretizar-lo e plotar em um gráfico, utilizando o bloco *Build Waveform*.

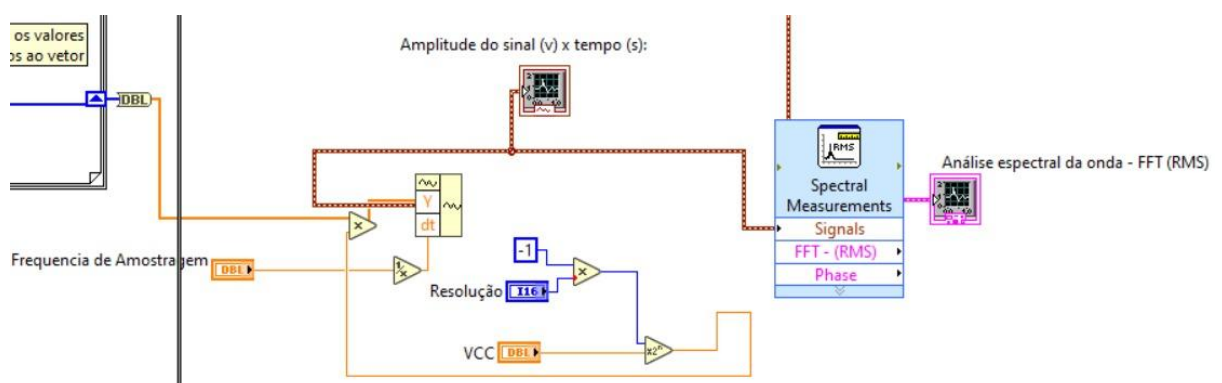


Figura 4 - Lógica da VI para processar, plotar e calcular o espectro de frequências do sinal

Para detectar as notas musicais, a implementação utilizada foi a seguinte: o sinal obtido já processado (ou seja, multiplicado pela LSB), passa por um filtro passa-faixa, o qual tem como frequência de corte inferior um valor de frequência um pouco menor que o da nota musical específica, e como corte superior um valor um pouco maior, esse sinal é enviado a um comparador (maior ou igual a), o qual verifica se a magnitude do sinal é alta o suficiente (maior que um valor numérico determinado), e, caso positivo, um indicador da respectiva nota musical acende, indicando a nota detectada. Essa lógica se repete para cada oitava de cada nota.

Para as 7 notas musicais, foram implementados detectores para 3 oitavas de cada uma, sendo elas a terceira, a quarta e a quinta oitava, e para manter apenas um indicador por nota musical, o sinal do comparador passar por um OR, o que garante que caso o output de pelo menos um dos três comparadores de cada oitava da nota musical for positivo, o indicador seja ligado. Como a banda de frequências de cada oitava é suficientemente grande, a ativação das portas OR implementadas ocorre de maneira inequívoca.

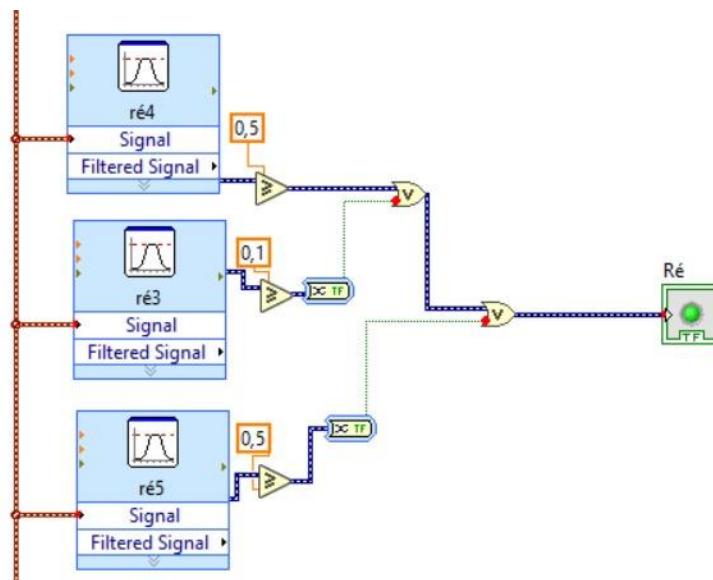


Figura 5 - Lógica de detecção da 3ª, 4ª e 5ª oitava para a nota Ré

Os valores utilizados para comparar a magnitude do sinal variam para cada oitava de cada nota, e foram determinados experimentalmente levando dois critérios em consideração: para que não sejam muito pequenos ao ponto de indicarem um falso positivo, mas também não muito altos de modo que impossibilitem a detecção.

As frequências das oitavas foram obtidas da seguinte tabela encontrada na internet:

Nota musical	-	Frequência em Hz									
		1° Oitava	2° Oitava	3° Oitava	4° Oitava	5° Oitava	6° Oitava	7° Oitava	8° Oitava	9° Oitava	-
Dó		33	66	132	264	528	1056	2112	4224	8448	16896
Dó #		34,947	69,894	139,79	279,6	559,15	1118,3	2236,6	4473,2	8946,4	17893
Ré		37,026	74,052	148,1	296,2	592,42	1184,8	2369,7	4739,3	9478,7	18957
Ré #		39,237	78,474	156,95	313,9	627,79	1255,6	2511,2	5022,3	10045	20089
Mi	20,79	41,58	83,16	166,32	332,6	665,28	1330,6	2661,1	5322,2	10644	
Fá	22,03	44,055	88,11	176,22	352,4	704,88	1409,8	2819,5	5639	11278	
Fá #	23,33	46,662	93,324	186,65	373,3	746,59	1493,2	2986,4	5972,7	11945	
Sol	24,72	49,434	98,868	197,74	395,5	790,94	1581,9	3163,8	6327,6	12655	
Sol #	26,19	52,371	104,74	209,48	419	837,94	1675,9	3351,7	6703,5	13407	
Lá	27,75	55,506	111,01	222,02	444	888,1	1776,2	3552,4	7104,8	14210	
Lá #	29,4	58,806	117,61	235,22	470,4	940,9	1881,8	3763,6	7527,2	15054	
Si	31,15	62,304	124,61	249,22	498,4	996,86	1993,7	3987,5	7974,9	15950	
Dó	33	66	132	264	528	1056	2112	4224	8448	16896	

Figura 6 - Tabela de frequências das notas musicais [3]

O valor das frequências de corte inferior e superior dos filtros passa-faixa são no máximo 10 Hz maior/menor que o valor da tabela, e no mínimo 2 Hz maior/menor, a depender de cada nota, o que também foi determinado experimentalmente, a fim de otimizar o funcionamento do programa.

Os conversores de sinal *From DD to boolean* (componente amarelo na saída de alguns comparadores) foram utilizados para que o programa permitisse a conexão com a porta lógica OR.

3) Virtual Instrument (VI do LabView) - Painel frontal

A Interface do VI implementado é extremamente similar ao do relatório anterior, consiste de um gráfico o qual exibe o sinal no domínio do tempo, outro gráfico para exibir o espectro de frequências, e 7 indicadores na parte inferior para mostrar a nota musical detectada.

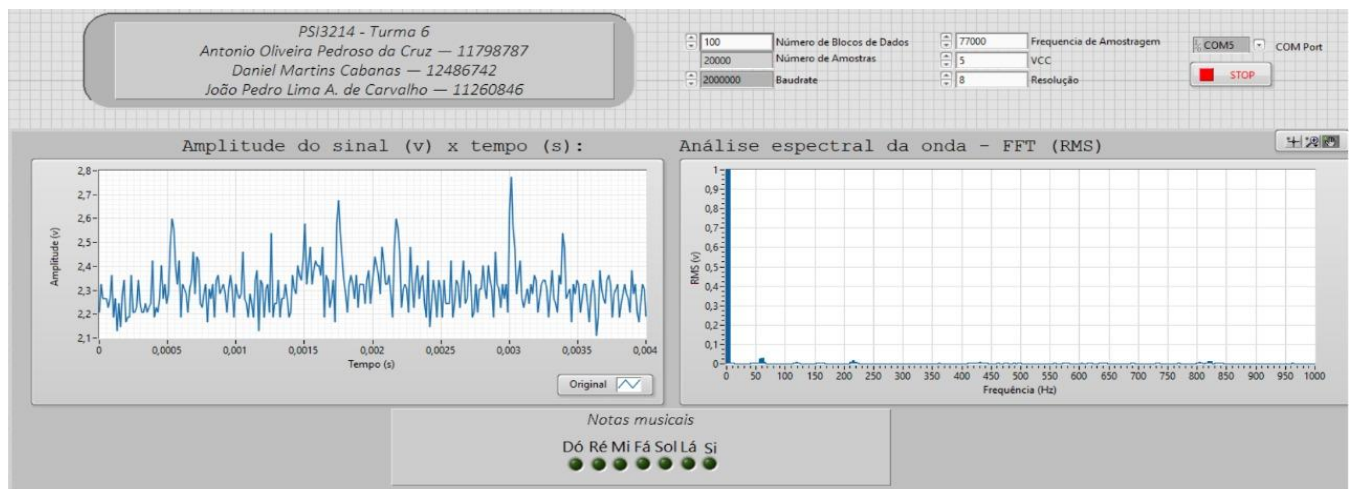


Figura 7 - Interface do VI

Além disso, há também as caixas de opções para alterar valores como resolução, VCC, e frequência de amostragem.

RESULTADOS

Na figura abaixo, o comportamento do VI para um sinal senoidal de 500 Hz:

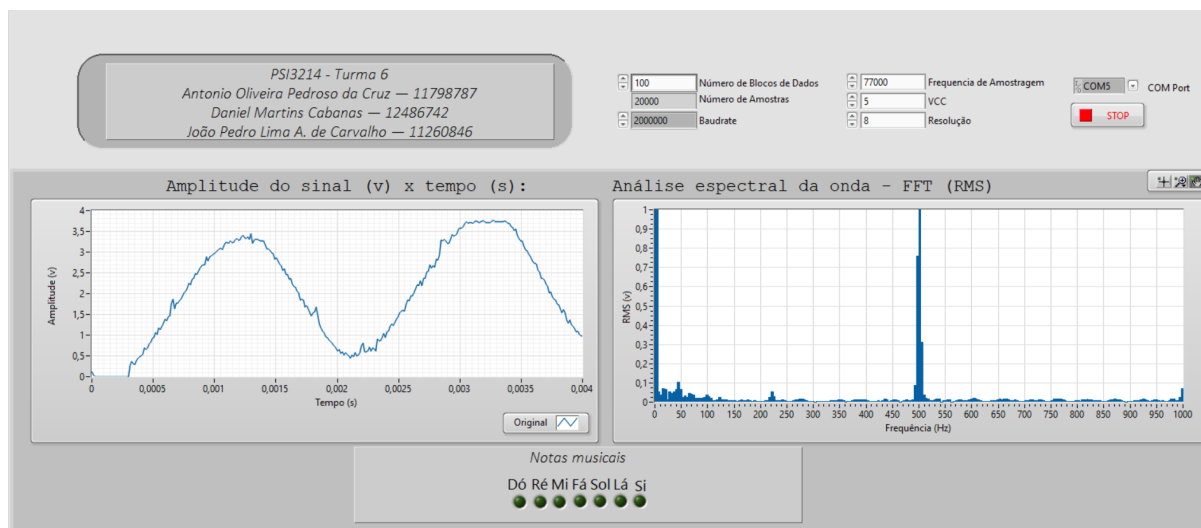


Figura 8 - Sinal senoidal de 500 Hz

Assim como na primeira etapa, essa parte do projeto funciona normalmente, entretanto, a frequência de amostragem foi ajustada para 77 kHz para aumentar a precisão do programa.

Para detecção das notas musicais, a frequência de amostragem foi alterada para 38,5 kHz, valor determinado experimentalmente, testando notas musicais e valores arbitrários para a frequência de amostragem.

Na próxima figura, os resultados para uma nota Lá na quarta oitava:

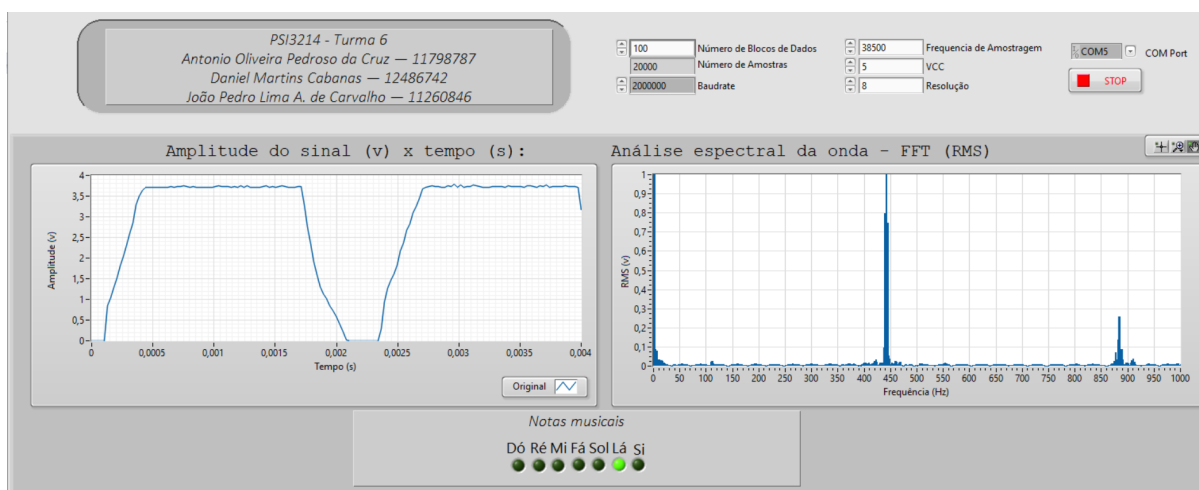


Figura 9 - Nota Lá 4ª oitava 444 Hz

Como é possível observar, o programa exibiu no espectro de frequências, predominantemente, a frequência da nota (~444 Hz), e o indicador Lá acendeu como esperado.

COMENTÁRIOS

A detecção de notas em oitavas menores (3ª ou inferior) se provou trabalhosa, dado que nessas oitavas, o espectro do sinal das notas apresentou mais de uma frequência com magnitude significativa, o que atrapalhava na lógica de detecção, sendo necessário escolher os valores de comparação com certa precisão. Isso ocorreu provavelmente devido aos efeitos do conversor digital do arduino, pois quando o sinal é convertido e discretizado, alguma informação pode se perder, modificando o sinal original. Também é plausível a suposição de que houve interferência com ruídos de baixa frequência [4], caracterizados por serem de menos do que 125 hz, e que distorceram o sinal capturado, já que da 3ª oitava para baixo as frequências características também são menores do que 125 hz.

CONCLUSÃO

De acordo com os resultados obtidos, é possível concluir que os objetivos iniciais do projeto - plotar o sinal no domínio do tempo, calcular seu espectro de frequências e detectar a nota musical - foram alcançados com êxito. Além disso, foi possível aprofundar os conhecimentos básicos de processamento de sinais, com foco em análise espectral, amplificadores operacionais, conversão analógica para digital e a montagem de circuitos integrados em geral.

Embora o sistema esteja em completo funcionamento, é possível realizar algumas melhorias, como por exemplo: diminuir a distorção do sinal original pelo conversor do arduino, aumento da granularidade de notas que ele é capaz de identificar (suporte a mais oitavas), acoplamento de filtros para suprimir a interferência de ruídos provenientes do ambiente e até a inserção de uma espuma no microfone para mitigar o efeito do vento e melhorar a qualidade de captação.

REFERÊNCIAS

HIGHLIGHTS in the History of the Fourier Transform. [S. I.]: IEEE, 25 jan. 2016. Disponível em: <https://www.embs.org/pulse/articles/highlights-in-the-history-of-the-fourier-transform/>. Acesso em: 21 nov. 2022.

FAST Fourier Transformation FFT - Basics. [S. I.], 17 fev. 2017. Disponível em: <https://www.nti-audio.com/en/support/know-how/fast-fourier-transform-fft>. Acesso em: 21 nov. 2022.

TABELA com a frequência de todas as notas musicais. [S.I.], 23 jun. 2016. Disponível em: <https://blogtudoamao.blogspot.com/2016/06/tabela-com-frequencia-de-todas-as-notas.html>. Acesso em: 21 nov. 2022.

RUÍDOS de baixa frequência - o que são infrassons?. [S.I.], [S.D.] Disponível em <https://www.ruidobaixafrequencia.pt/infrassons-e-ruido-de-baixa-frequencia/>. Acesos em: 21 nov. 2022.