

# Processamento Digital de Sinais de Voz

Turma de Computação

Rodrigo Franciozi Rodrigues da Silva 14.04014-0

Igor Amaral Correa 15.00588-7

Hugo da Silva Bernardes Gonçalves

# Objetivo

A partir de uma vogal passada por microfone por uma pessoa, definir qual vogal é pelos formantes formados no espectro de amplitude em relação a frequência da voz. Tudo isso utilizando o programa LabView e da ferramenta NI myRIO nas aulas de laboratório.

## Técnicas Utilizadas

Foram utilizadas técnicas(funções) de programação gráfica no programa em questão, como:

- For
- While
- Array
- Cluster
- Armazenamento de variável
- Case
- Threshold

Também foi utilizado cálculos matemáticos principalmente na diferenciação de vogais, por covariância.

Gráficos foram utilizados principalmente na construção do programa, na procura de erros e acertos, e na finalização do programa, para a visualização dos resultados.

## Conceitos Apreendidos

Aprendizado da linguagem do programa LabView e da ferramenta do NI myRIO, definição e utilização dos formantes de uma onda de voz, aprendizado do cálculo da matriz de covariância, entre outros.

## Dificuldades Encontradas

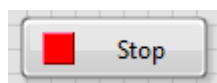
A coisa mais difícil para produzir esse projeto foi o aprendizado do LabView, em saber qual função seria a mais adequada para cada parte específica do código, pois o LabView tem uma biblioteca de função enorme, as vezes mais de 1 função que pode servir para a mesma coisa, e com isso fazendo com que seja necessário testes para ver qual a função que se encaixa melhor na aplicação desejada.

Também houve dificuldades na área de entender o que são as formantes e como “adquiri-las” através de programação.

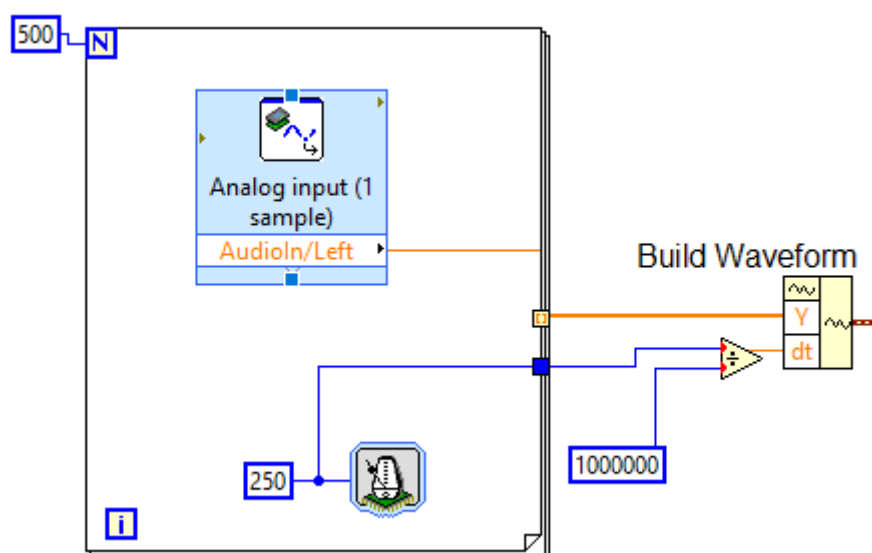
## Explicação do Código



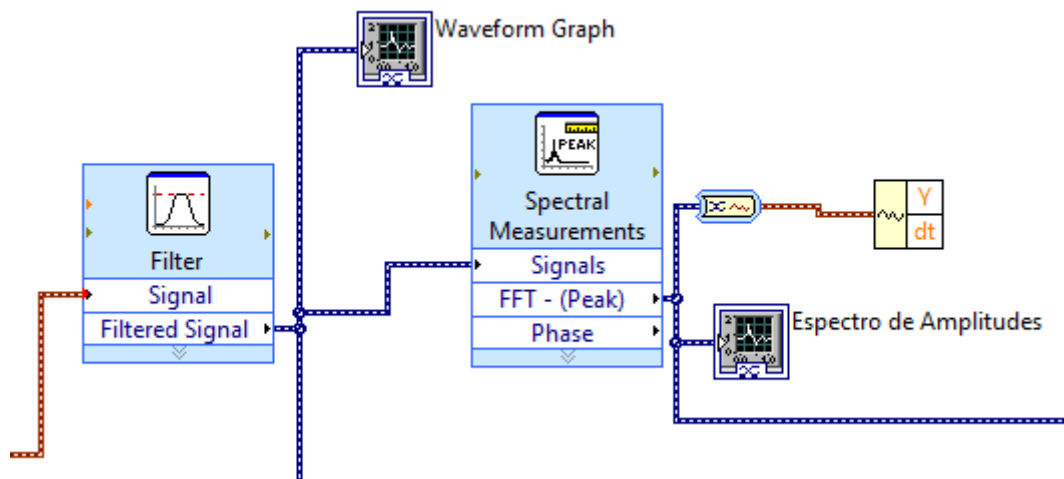
Primeiramente, o código todo foi colocado dentro de uma função While, com uma condição de parada usando um Stop Button:



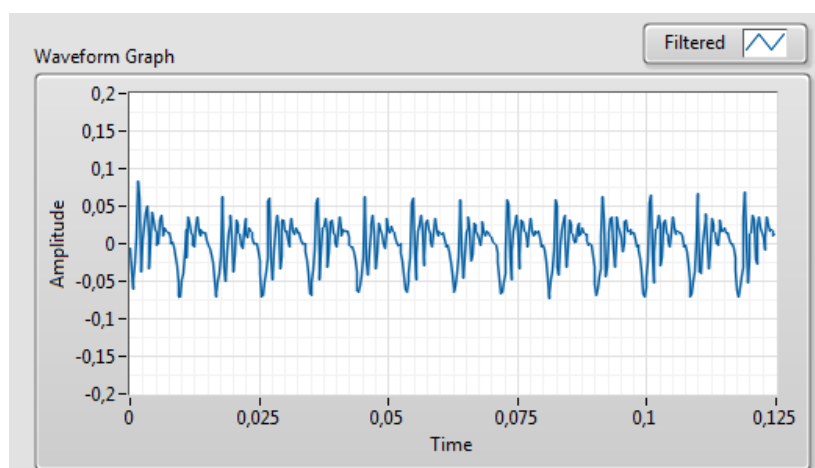
Segundamente:



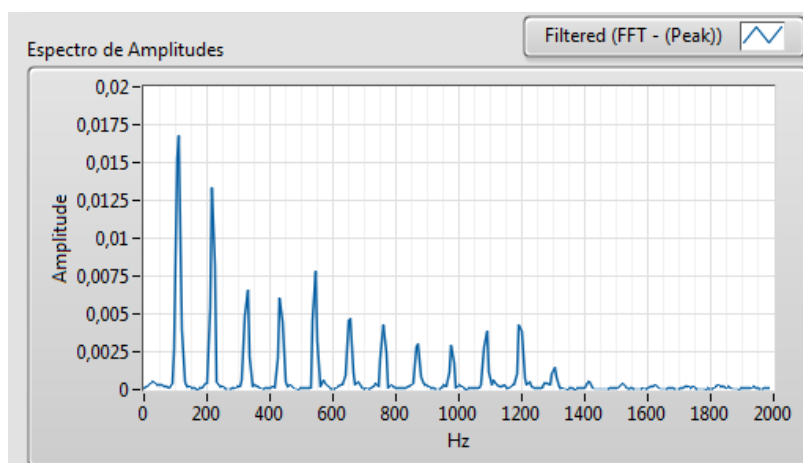
Foi colocado uma função de captação de som dentro de uma função For, onde o valor indicado no N azul significa a quantidade de vezes que o que está dentro da caixa do For será repetido. E também, dentro desse mesmo For, foi colocado uma função de espera, que faz com que apenas depois de ter sido captado a quantidade de vezes especificada por um numero, dado em micro segundos(nesse caso 250), o For passe para a proxima repetição. E por final, fora desse For, foi colocado uma função para criar uma forma de onda Y por dt.

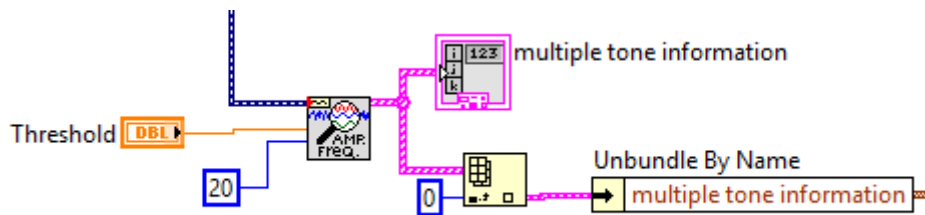


Logo depois, foi colocado uma função para se limpar o sinal adquirido pelo microfone, e colocado em um gráfico para ser analisado:



E também, depois de passar por uma função para pegar o espectro de amplitude da onda em questão, foi colocado em outro gráfico para a análise:

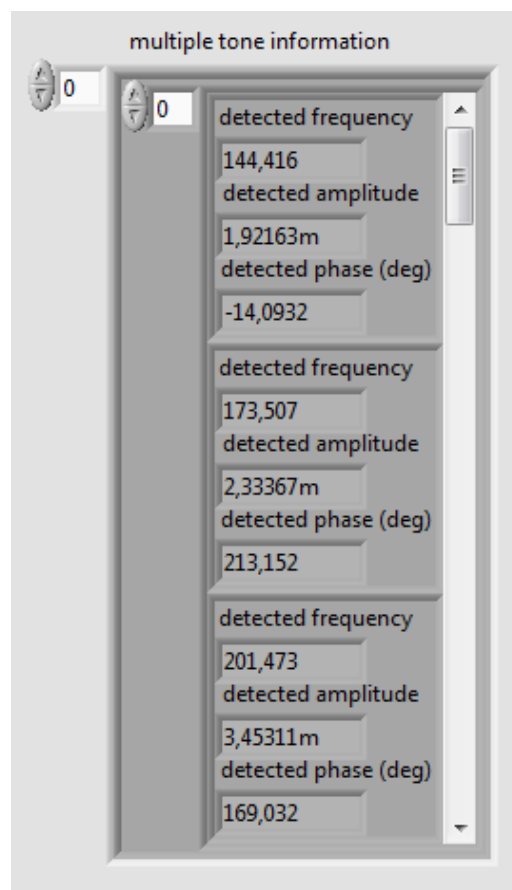




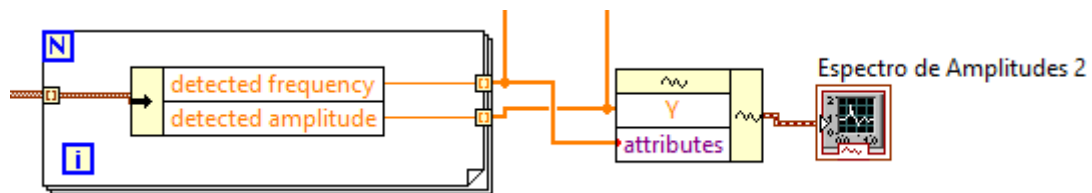
Usando a onda filtrada do sinal, foi usado um “Threshold” para pegar apenas valores acima do valor especificado:



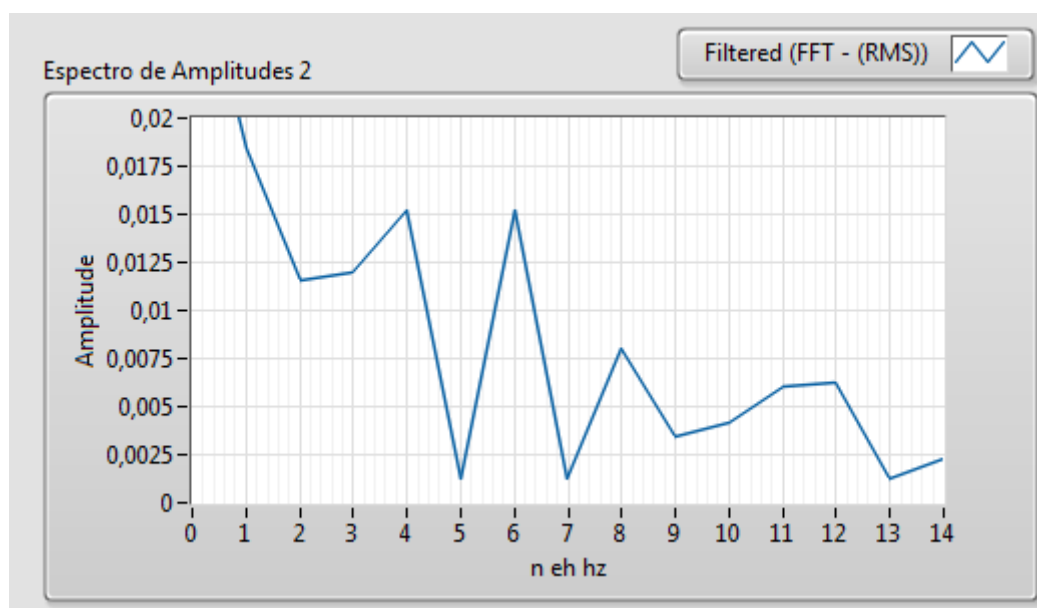
E ,como mostra na figura, 20 é o número maximo de valores a ser adquirido pela saída dessa função. Depois de pegar apenas alguns valores do sinal, foi usado uma função para mostrar esses valores adquiridos(multiple tone information):



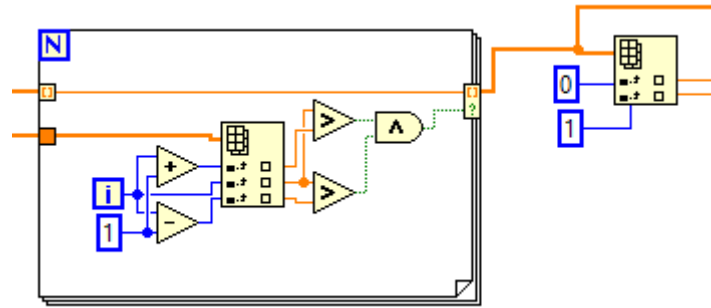
E no final, com uma função de selecionar o index de uma array, foi retirado apenas as informações necessárias(amplitude e frequência da onda).



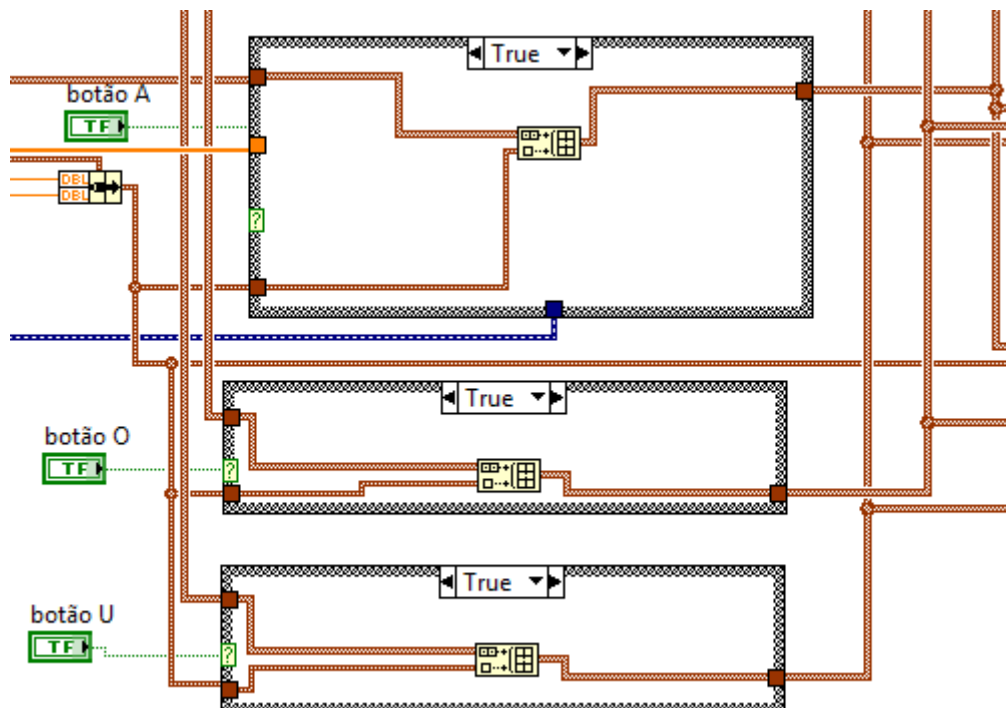
Colocando os valores de amplitude e frequência em um For, e dentro dele uma função para separar os dois valores, foi possível criar mais um gráfico de amplitude por frequência, mas dessa vez apenas com os valores anteriormente passados pelo Threshold:



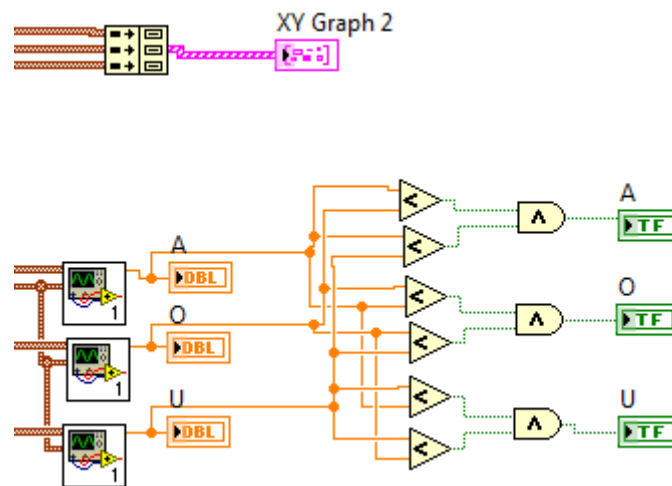




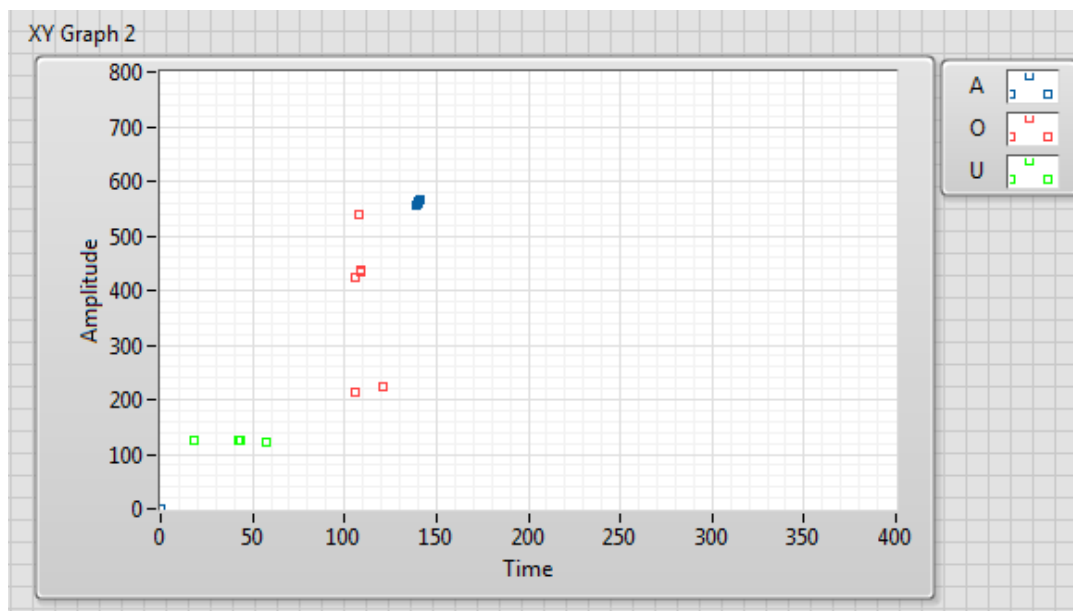
Usando novamente a função For, e colocando o número de repetições igual ao quantidade de valores que foram passados no Threshold, foi comparado a amplitude de cada casa do vetor com sua casa anterior e sua casa posterior, e se esse valor fosse o maior dentre esses 3, armazenou-se em um outro vetor de 2 casas apenas, ou digamos, está sendo pego os dois primeiros formantes da onda de voz passada pelo microfone e armazenando em um vetor para futuras análises.



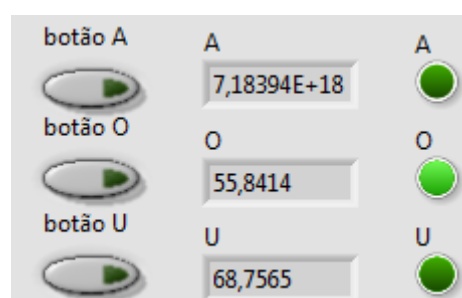
Pegando os resultados dos primeiros dois formantes, dependendo do botão clicado, armazenando-se em uma das arrays de valores de A, O ou U.



Depois de separar em diferentes arrays os valores de A, O e U, plotou-se em um gráfico de pontos XY:



E depois passando as arrays por uma função de covariância, analisando o resultado com os demais arrays, fazendo piscar uma luz da vogal que está mais próxima da voz passada pelo microfone:



## Objetivos Atingidos

Dos objetivos propostos pela atividade, mesmo com programa não conseguindo diferenciar perfeitamente as vogais expressadas pelo microfone, pode-se dizer que todos os objetivos foram atingidos, pois a voz humana é muito imprecisa, logo, fazendo um programa simples como foi proposto, não se pode esperar uma diferenciação de vogais perfeita.

## Conclusões

A voz humana é muito complexa para se definir com tanta simplicidade, mas com muitos estudos e diferentes aplicações, é possível diferenciar a pessoa que está gesticulando e até sobre o assunto que ela está falando. Porém, com pouco tempo, já foi possível diferenciar vogais, e aprender muitas coisas sobre frequência e amplitude de ondas, e outros conceitos da fala humana em específico.