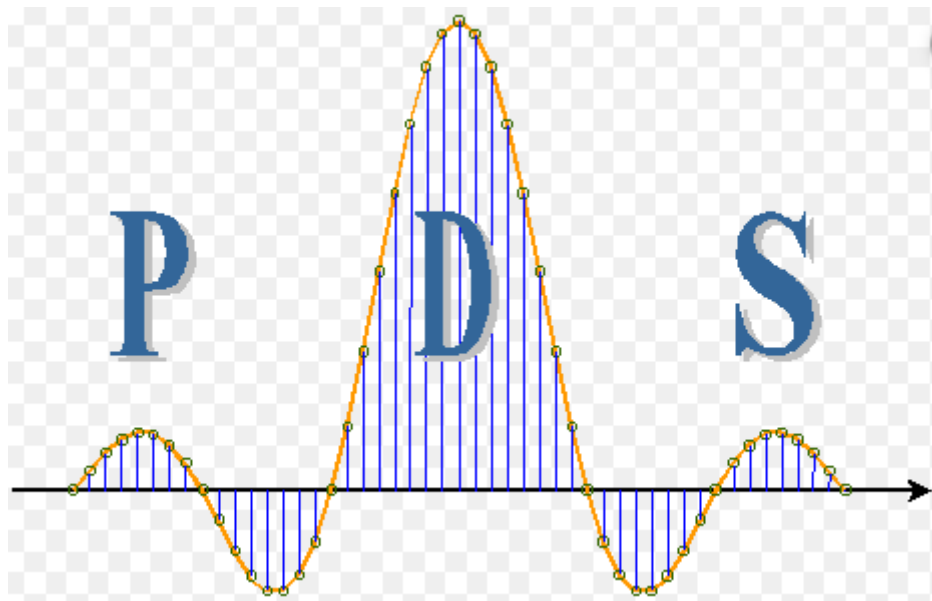


# Processamento Digital de Sinal

## Trabalho Prático 2



### Grupo 4:

43861 – Francisco Chicharro

43864 – Joao Monteiro

43874 – Joao Florentino

Docente: Paulo Marques

22/12/2019

## Índice

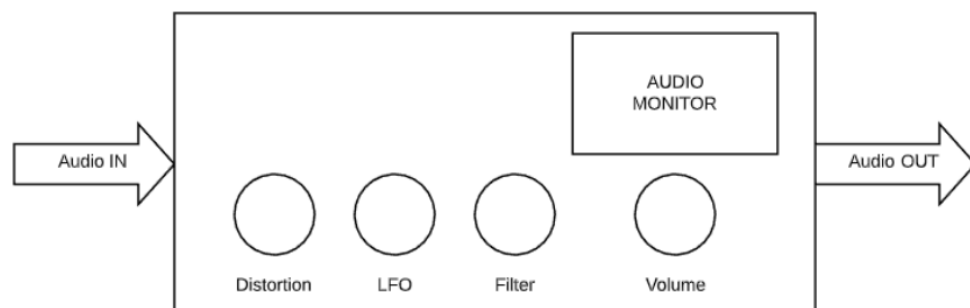
Índice Figuras.....	2
Objetivos .....	3
Introdução .....	4
Exercício 1 .....	6
Exercício 2 .....	7
Exercício 3 .....	9
Exercício 4 .....	12
Conclusões.....	13
Referências bibliográficas .....	14

## Índice Figuras

Figura 1:Disfarce de Voz.....	6
Figura 2:Expressão geral para Transformada Z .....	7
Figura 3:Diagrama de blocos do Oscilador .....	8
Figura 4: Expressão geral do filtro parte 1 .....	9
Figura 5:Expressão geral do filtro parte 2 .....	10
Figura 6:Filtro para $R_{\text{polo}} > R_{\text{zero}}$ .....	10
Figura 7:Filtro para $R_{\text{zero}} > R_{\text{polo}}$ .....	10
Figura 8:Modelo geral do processador efeito de áudio .....	12

## Objetivos

- Implementar um algoritmo de distorção de fala de acordo com o exemplo dado na aula teórica e posteriormente testar o sistema com sinais sintéticos e de voz humana.
- Recorrendo à Transformada Z implemente um gerador de sinal sinusoidal com amplitude e frequência programáveis. Fazendo uso do DAC do Arduino DUE, determinar a gama de amplitude e de frequência suportadas.
- Por mapeamento de polos e zeros no plano complexo Z projetar e implementar um filtro que permita selecionar uma banda estreita do espectro de frequência. Verifique também a influência da localização relativa dos polos e zeros na seletividade do filtro.
- Projetar e implementar um processador de efeitos de áudio com as funcionalidades mínimas propostas: Distorção, modulação de amplitude com LFO, filtragem. Todos os efeitos deverão ser programáveis em tempo real através de push-buttons e/ou potenciômetros.



## Introdução

Por definição, a transformada Z de um sinal discreto é:  $\sum_{n=-\infty}^{\infty} x[n] * z^{-n}$ , onde  $z$  é um número complexo.

Estas transformadas são baseadas em séries de potências e têm grande importância nos métodos atuais de análise de sistemas de controlo discreto, em processos de amostragem, no processamento de sinais, etc. Possui a capacidade de se realizar convulsão de maneira facilitada, apenas fazendo uma multiplicação simples dos sinais.

A transformada de Z não converge para todos os valores de Z. A região de convergência (ROC) de  $X(z)$  é o conjunto de todos os valores de  $z$  para qual  $X(z)$  alcança valores finitos.

Os polos e zeros analisam o desempenho de um de um sistema e ajudam a verificar a sua estabilidade. São os Polos e os Zeros que controlam o funcionamento de um sistema e ajudam a desenhar filtros para determinadas frequências. As frequências para as quais os valores do dominador e nominado se tornam zero em uma função de transferência são chamados de polos e zeros. Uma função de transferência é uma função com variáveis complexas. Pode ser obtida através de equações algébricas simples. Os polos são as raízes do denominador de uma função de transferência enquanto que os zeros são os numeradores:

$$H(s) = \frac{N(s)}{D(s)}$$

A transformada Z é uma ferramenta bastante utilizada para passar um sinal do domínio do tempo para o domínio da frequência, neste caso complexa, em relação aos sinais discretos. Oferece uma maneira eficiente de tratar filtros.

Um filtro tem como função selecionar, rejeitar ou igualizar uma ou várias gamas de frequência de um sinal elétrico. Os filtros constituem uma das aplicações mais comuns da eletrónica, sendo amplamente utilizados na aquisição e processamento de sinais áudio, vídeo e de dados, em sistemas de telecomunicações, etc. Os filtros elétricos podem ser de cinco tipos básicos: Passa-baixo, passa-alto, passa-banda, rejeita-banda e passa tudo.

Focou-se em particular neste projeto no filtro passa-banda. É um dispositivo que permite a passagem das frequências de uma certa faixa e rejeita (atenua) as frequências fora dessa faixa. Estes filtros podem ser também obtidos através da combinação entre um filtro passa-baixo e um filtro passa-alto.

Um filtro ideal possuiria uma banda passante totalmente plana (sem atenuação), e iria atenuar completamente todas as outras frequências fora desta banda. O que acontece na prática é que nenhum filtro é ideal. O filtro não atenua todas as frequências fora da banda desejada e existe até uma região em particular em que todas as frequências são atenuadas mas não rejeitadas. Esta região é conhecida como *roll-off* do filtro e é geralmente expresso em dB de atenuação por oitava de frequência.

Distorção nas telecomunicações e no processamento de sinais é a transmissão infiel de sinais, decorrente da atenuação e do deslocamento de fase de diferentes componentes de frequência. consiste numa alteração da forma do sinal durante a sua propagação desde o emissor até ao recetor. A distorção pode resultar do comportamento não-linear de alguns dos componentes que compõem o percurso do sinal ou pela simples resposta em frequência do meio de transmissão.

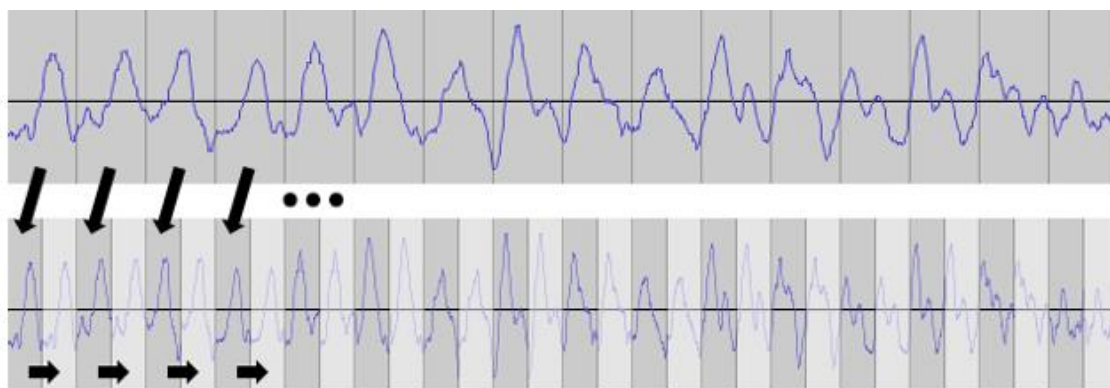
Um LFO (Low Frequency Oscillator) permite gerar uma forma de onda com frequência abaixo da capacidade de audição e possibilita a seleção de frequência e forma de onda que serão utilizadas para modificar o sinal audível. Pode modificar também o sinal provindo do gerador de ruído. Os LFOs possuem também a possibilidade de selecionar o tempo de atraso e a quantidade de fade in (delay time)

## Exercício 1

**P: Implemente um algoritmo de distorção de fala de acordo com o exemplo dado na aula teórica. Teste o sistema com sinais sintéticos e de voz humana.**

Neste exercício, de acordo com o exemplo fornecido na aula teórica, implementámos um algoritmo para distorção de voz através de um buffer onde colocámos amostras nos vários índices do mesmo. Fazíamos avançar as amostras dependendo da nossa preferência: torná-la mais grave ou mais aguda através da alteração do período.

Um período menor dará uma frequência maior, e quanto maior for frequência mais comprimidas irão estar as amostras, o que irá dar origem a um som mais agudo. Pelo contrário, aumentando o período a frequência será menor, o que tornará o som mais grave pois as amostras vão estar mais espaçadas:



*Figura 1:Disfarce de Voz*

## Exercício 2

**P: Recorrendo à Transformada Z implemente um gerador de sinal sinusoidal com amplitude e frequência programáveis. Fazendo uso do DAC do Arduino DUE, determine a gama de amplitude e de frequência suportadas**

O sistema terá como entrada  $X[n] = \delta[n]$  e como saída terá uma senoide. Assim, a função de transferência irá ser  $h[n] = y[n] = \cos(\omega_0 n)$ .

A frequência digital ( $\omega_0$ ) irá ser  $\omega_0 = \frac{2\pi f}{f_s}$  onde  $f$  será a frequência programável e  $f_s$  a frequência de amostragem.

Teremos como expressão geral:

The image shows two handwritten equations on lined paper. The top equation is the Z-transform:  $X(z) = \sum_{n=-\infty}^{+\infty} x[n] z^{-n}$ , with  $z = re^{j\theta}$  written to the right. The bottom equation is the Discrete-Time Fourier Transform (DTFT):  $X(\omega) = \sum_{n=-\infty}^{+\infty} x[n] e^{-j\omega n}$ . This second equation is enclosed in a hand-drawn rectangular box, and an arrow points from the text 'DTFT' to the box.

Figura 2: Expressão geral para Transformada Z

Desenvolvendo a transformada da função transferência:

$$H(z) = TZ(h[n]) = \sum_{n=-\infty}^{\infty} \cos(\omega_0 n) z^{-n} = \frac{1}{2} \left( \sum_{n=0}^{\infty} e^{j\omega_0 n} z^{-n} + \sum_{n=0}^{\infty} e^{-j\omega_0 n} z^{-n} \right) =$$
$$= \frac{z^2 - \cos(\omega_0)z}{z^2 - \cos(\omega_0)z + 1}$$

Uma vez que o sistema é não causal, para o tornar realizou-se o avanço do sinal de entrada multiplicando a expressão por  $\frac{1}{z^2}$ :

$$H(z) \times \frac{1}{z^2} = \frac{z^2 - \cos(\omega_0)z^{-1}}{1 - \cos(\omega_0)z^{-1} + z^{-2}} \quad (=)$$



$Y(z)(1 - \cos(\omega_0)z^{-1} + z^{-2}) = X(z)[1 - \cos(\omega_0)z^{-1}]$ , realizando a transformada inversa, chegamos a expressão geral para implementação:

$$Y[n] = Y[n-1] \times 2\cos(\omega_0) - Y[n-2] + X[n] - \cos(\omega_0)X[n-1]$$

Para melhor compreensão, através do exemplo teórico do diagrama de blocos do oscilador da aula teórica, adaptou-se o mesmo ao nosso exercício:

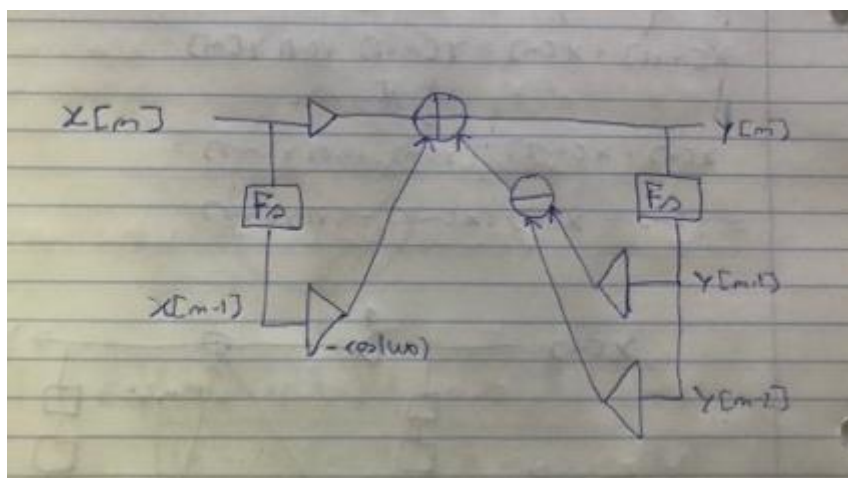


Figura 3: Diagrama de blocos do Oscilador

Através da expressão obtida, desenvolveu-se o código necessário para gerar o oscilador com a frequência e amplitude programável.

### Exercício 3

**P:** Por mapeamento de polos e zeros no plano complexo Z projete e implemente um filtro que permita selecionar uma banda estreita do espectro de frequência. Considere  $F_s=10\text{kHz}$ . Verifique a influência da localização relativa dos polos e zeros na seletividade do filtro.

Neste exercício foi solicitado que implementássemos um filtro que numa certa frequência deixasse passar o sinal (banda estreita) e fora dessa frequência o sinal era “cortado”. Para tal, recorreu-se novamente ao mapeamento de polos e zeros para chegar á expressão geral que devia ser implementada para o feito:

$$\begin{aligned}
 H(z) &= \frac{(z - \rho z e^{j\omega_0})(z - \rho z e^{-j\omega_0})}{(z - \rho r e^{j\omega_0})(z - \rho r e^{-j\omega_0})} = \\
 &= \frac{z^2 - z \rho z e^{j\omega_0} - z \rho z e^{-j\omega_0} + (\rho z^2 e^{j\omega_0 - j\omega_0})}{z^2 - z \rho r e^{j\omega_0} - z \rho r e^{-j\omega_0} + (\rho r^2 e^{j\omega_0 - j\omega_0})} = \\
 &= \frac{z^2 - 2 \rho z e^{j\omega_0} - 2 \rho z e^{-j\omega_0} + \rho z^2}{z^2 - 2 \rho r e^{j\omega_0} - 2 \rho r e^{-j\omega_0} + \rho r^2} = \\
 &= \frac{z^2 - 2 \rho z (e^{j\omega_0} + e^{-j\omega_0}) + \rho z^2}{z^2 - 2 \rho r (e^{j\omega_0} + e^{-j\omega_0}) + \rho r^2} = \\
 &= \frac{z^2 - 2 \rho z \cos(\omega_0) + \rho z^2}{z^2 - 2 \rho r \cos(\omega_0) + \rho r^2} = \\
 &= \frac{1 - 2 \rho z \cos(\omega_0) z^{-1} + \rho z^2 z^{-2}}{1 - 2 \rho r \cos(\omega_0) z^{-1} + \rho r^2 z^{-2}} = \frac{Y(z)}{X(z)}
 \end{aligned}$$

Figura 4: Expressão geral do filtro parte 1

$$Y(z) (1 - 2rp \cos(\omega_0) z^{-1} + rp^2 z^{-2}) = X(z) (1 - 2rp \cos(\omega_0) z^{-1} + rp^2 z^{-2})$$

$$\downarrow Tz^{-1}$$

$$Y(m) = X(m) - 2 \cos(\omega_0) rp X(m-1) + rp^2 X(m-2) +$$

$$2 rp \cos(\omega_0) X(m-1) - rp^2 X(m-2)$$

Figura 5: Expressão geral do filtro parte 2

Para perceber melhor o comportamento da localização dos polos e zeros assim como a sua influência quanto á frequência de corte, realizou-se em código MATLAB um programa que consegue representar este comportamento:

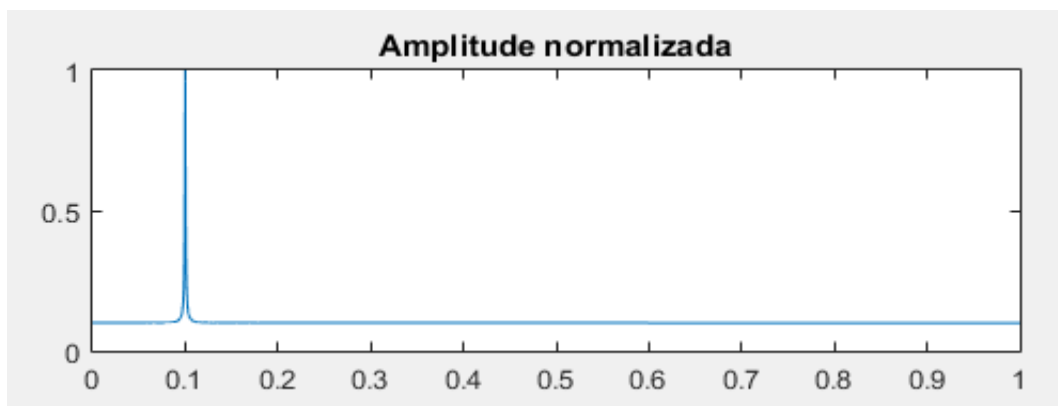


Figura 6: Filtro para  $R_{polo} > R_{zero}$

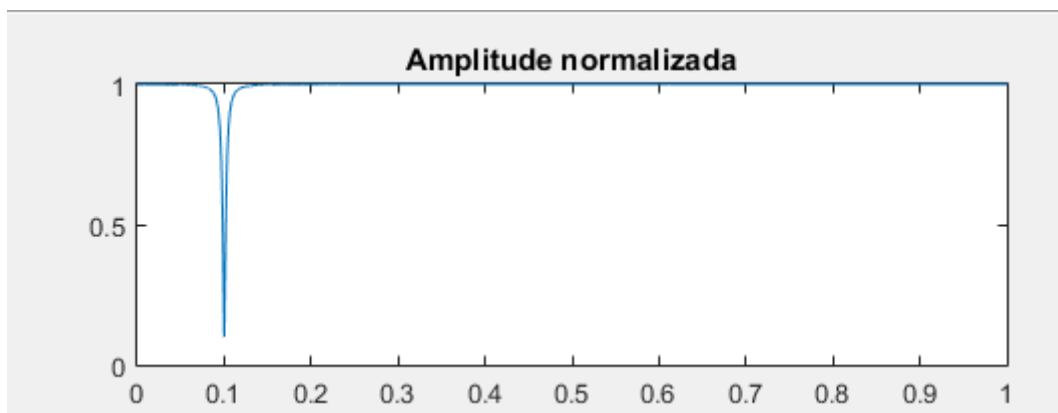


Figura 7: Filtro para  $R_{zero} > R_{polo}$

Nestes gráficos podemos visualizar que, ao contrário do normal que seria a apresentação de todas as frequências, este vai apenas mostrar o comportamento da frequência de corte escolhida.

Sempre que a amplitude dos zeros for maior que a amplitude dos pólos, a frequência de corte escolhida é excluída, mas se a amplitude dos polos for maior que a amplitude dos zeros, apenas vai passar a frequência pretendida.

## Exercício 4

**P :**Projete e implemente um processador de efeitos de áudio com as seguintes funcionalidades mínimas: **Distorção**, **modulação de amplitude com LFO**, **filtragem**. Todos os efeitos deverão ser programáveis em tempo real através de push-buttons e/ou potenciômetros.

Neste último exercício, é solicitado a realização de um processador de efeitos de áudio onde várias funcionalidades vão estar inseridas. A um sinal de entrada “Audio IN” seriam aplicadas algumas funções de forma a alterar o mesmo :

**Distorção:** Através desta função o sinal de entrada vai ser distorcido, sendo que através do exercício de “disfarce” da voz humana será possível fazer a distorção

**LFO:** Através de uma modelação LFO vai ser possível adaptar o sinal em amplitude em algumas partes do mesmo.

**Filter:** A implementação de um filtro passa banda sobre uma banda estreita vai permitir filtrar o sinal numa frequência desejada.

**Volume:** Através do volume, será possível controlar o ganho do sinal

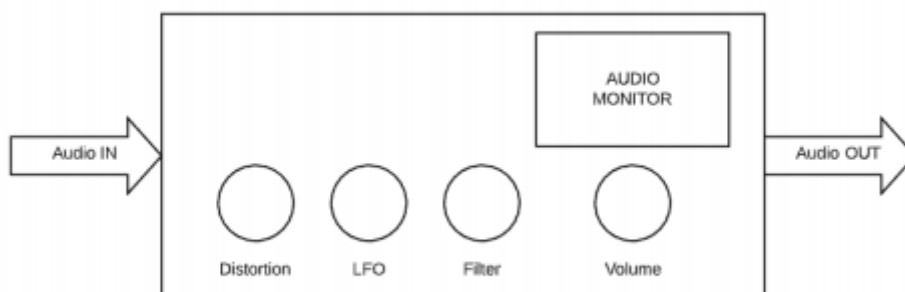


Figura 8:Modelo geral do processador efeito de áudio

## Conclusões

Com este trabalho adquirimos conhecimentos e uma maior destreza a trabalhar com o Arduino e perceber as suas limitações. Aprendemos também como utilizar a Transformada Z para implementar osciladores sinusoidais de frequência programável e fazer filtros passa banda para uma frequência de corte específica. O facto de nos termos deparado com alguns problemas ao longo do desenvolvimento deste projeto fez com que aprofundássemos o nosso estudo para que conseguíssemos resolver o que estava de errado, adquirindo uma maior experiência em processamento de sinal.

## Referências bibliográficas

[http://webx.ubi.pt/~felippe/texts2/an\\_sinais\\_cap6.pdf](http://webx.ubi.pt/~felippe/texts2/an_sinais_cap6.pdf) (acedido a 9 de dezembro de 2019)

[https://pt.wikipedia.org/wiki/Filtro\\_passa-faixa](https://pt.wikipedia.org/wiki/Filtro_passa-faixa) (acedido a 20 de dezembro de 2019)

<http://www3.dsi.uminho.pt/adriano/Teaching/Comum/FactDegrad.html> (acedido a 20 de dezembro de 2019)

[https://pt.wikipedia.org/wiki/Distor%C3%A7%C3%A3o\\_\(telecomunica%C3%A7%C3%B5es\)](https://pt.wikipedia.org/wiki/Distor%C3%A7%C3%A3o_(telecomunica%C3%A7%C3%B5es)) (acedido a 20 de dezembro de 2019)

<https://web.fe.up.pt/~ajf/pds/> imagem (acedido a 22 dezembro 2019)