



TRABALHO DE CONCLUSÃO DE CURSO

ESTIMADOR DE DIREÇÃO DE CHEGADA

Autor: Gabriel da Silva Lima

**UNIVERSIDADE DE BRASÍLIA**

**FACULDADE DE TECNOLOGIA**

Orientador: Prof. Ricardo  
Zelenovsky

Brasília, maio de 2021.

**UNIVERSIDADE DE BRASÍLIA**  
**FACULDADE DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**ESTIMADOR DE DIREÇÃO DE CHEGADA**

**GABRIEL DA SILVA LIMA**

TRABALHO DE CONCLUSÃO DE CURSO SUBMETIDO AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO GRAU DE ENGENHEIRO ELETRICISTA.

## **ORIENTADOR: RICARDO ZELENOVSKY**

### **FICHA CATALOGRÁFICA**

LIMA , GABRIEL DA SILVA

Estimador de direção de chegada [Distrito Federal], 2021.

Trabalho de Conclusão de Curso – Universidade de Brasília, Faculdade de Tecnologia

Departamento de Engenharia Elétrica. Orientação: Prof. Dr. Ricardo Zelenovsky.

1. Estimador de fonte sonora

2. Orientação da plataforma

3. Estimção DOA

I. ENE/FT/UnB II.

### **REFERÊNCIA BIBLIOGRÁFICA**

LIMA, GABRIEL DA SILVA (2021). Plataforma móvel guiada pelo som. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2021, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 54p.

## AGRADECIMENTOS

*Agradeço primeiramente a Deus pelo dom da vida e pela oportunidade de chegar até aqui.*

*Agradeço aos meus familiares pelo incondicional apoio em todos os momentos da minha vida. Em especial à minha mãe que nunca mediu esforços em me incentivar a seguir em frente, mesmo nos momentos de dúvida e dificuldade. Ao meu pai, que mesmo estando geograficamente distante sempre garantiu minha provisão, sem a qual eu não teria como chegar até esse momento. À minha irmã Camila, que me auxiliou nos demais afazeres enquanto eu estava me dedicando com o projeto. Aos meus avós que sempre me apoiaram e acreditaram em mim, mesmo quando eu mesmo não acreditava.*

*Agradeço a minha igreja local, por todo o apoio fornecido durante o projeto. Sem eles, dificilmente teria condições de realizar os testes em estúdio apresentados nesse projeto.*

*Ao meu orientador, Prof. Ricardo Zelenovsky, por toda a ajuda e disposição, mesmo fora do horário convencional das reuniões. Por toda a paciência que teve comigo ao me explicar o funcionamento do projeto e pelos momentos em aula, que me ensinou e me fez apreciar a disciplina.*

*Aos professores José Edil e Daniel Café por aceitarem o convite para participar da banca de avaliação do projeto.*

## RESUMO

Este trabalho aborda a melhoria e avaliação de um sistema estimador de direção de chegada (DOA) de um sinal sonoro em um ambiente fechado, previamente desenvolvido por alunos da Universidade de Brasília.

Para a captação dos sinais são utilizados dois microfones construídos por alunos e professores. Para se identificar a posição da fonte sonora é utilizado um cálculo de correlação temporal entre os sinais captados, medindo o erro quadrático médio entre as amostras e determinar o atraso em tempo discreto.

O projeto todo foi feito no microcontrolador MSP430-F5529, em seu próprio ambiente de desenvolvimento *Code Composer Studio*, em linguagem C.

## ABSTRACT

*This work deals with the improvement and evaluation of a direction of arrival (DOA) estimator system of an audible signal in a closed environment, previously developed by students of the University of Brasília.*

*To capture the signals, two microphones built by students and teachers are used. To identify the position of the sound source, a temporal correlation calculation between the captured signals is used, measuring the mean quadratic error between the samples and determining the delay in discrete time.*

*The whole project was done in the MSP430-F5529 microcontroller, in its own Code Composer Studio development environment, in C language.*

## SUMÁRIO

1	AMBIENTAÇÃO.....	12
1.1	Motivação: .....	12
1.2	Objetivo: .....	12
1.3	Metodologia: .....	13
1.4	Organização: .....	13
2	EMBASAMENTO TEÓRICO .....	14
2.1	Arranjo de sensores.....	14
2.2	Bandas de Frequência .....	16
2.3	Ambiguidade e distância entre sensores .....	18
2.4	Filtros digitais: .....	23
3	HARDWARE .....	24
3.1	MSP430 .....	24
3.2	Memória externa .....	24
3.3	Microfone e circuito pré-amplificador:.....	25
4	SOFTWARE PARA ESTIMATIVA DOA.....	28
4.1	<i>Firmware</i> .....	28
4.2	Método de estimação de DOA.....	28
4.3	Etapas da estimativa.....	31
4.3.1	Primeira etapa do tratamento: aplicação de filtro digital .....	32
4.3.2	Segunda etapa do tratamento: identificação da faixa de maior energia .....	32
4.3.3	Terceira etapa do tratamento: estimativa realizada com sinais filtrados.....	33
4.4	Resolução do estimador e limitações.....	35

5	RESULTADOS E ANÁLISE.....	36
5.1	Análise do método no MATLAB .....	37
5.2	Resultados Iniciais .....	41
5.3	Avaliação de resultados .....	49
5.4	Limitações do projeto .....	51
6	CONCLUSÃO.....	52
	REFERÊNCIAS BIBLIOGRÁFICAS.....	53

## LISTA DE FIGURAS

Figura 2-1: Arranjo linear uniforme de microfones para estimativa [10].	14
Figura 2-2: Frente de onda bidimensional [10]	15
Figura 2-3: Perfil espectral de uma voz masculina [1]	17
Figura 2-4: Perfil espectral de um assovio [1].	17
Figura 2-5: Situação normal em que o atraso é detectado [10].	18
Figura 2-6: Situação em que o atraso é igual a $\lambda$ [10].	19
Figura 2-7: Dois atrasos podem ser detectados caso ocorra Aliasing[10].	19
Figura 2-8: Limite da zona sem Aliasing é $\pm 34,87^\circ$ , valores além do máximo terão Aliasing [10].	21
Figura 2-9: Sistema de estimativa DOA montado.	22
Figura 2-10: Diagrama de blocos do circuito.	22
Figura 2-11: Design de filtro IIR em ambiente MATLAB [10].	23
Figura 3-1: Microfone acoplado à placa do circuito utilizado [9].	25
Figura 3-2: Resposta de magnitude versus frequência do microfone de eletreto usado no presente trabalho [9].	26
Figura 3-3: Circuito do estágio pré-amplificado [9]	26
Figura 3-4: Alimentação do circuito [9]	27
Figura 3-5: Ligação entre microfones, placa de circuito e MSP430 [9]	27
Figura 4-1: Ângulos possíveis para a estimativa [10]	29
Figura 4-2: Montagem proposta de arranjo com dois microfones [10].	29
Figura 4-3: Senoides defasadas.	33
Figura 4-4: Gráfico da diferença das senoides.	34
Figura 4-5: Representação da operação de correlação efetuada [10].	35
Figura 4-6: Diagrama de blocos para o sistema de processamento.	36
Figura 5-1: Sinal dos microfones de um ensaio sem filtro.	37
Figura 5-2: Sinais sonoros filtrados.	38
Figura 5-3: Canais filtrados escalonados.	39
Figura 5-4: Canais filtrados superpostos.	39
Figura 5-5: Defasagem entre os sinais dos dois microfones em uma aquisição.	40
Figura 5-6: Juiz do DOA.	41
Figura 5-7: Gráfico das estimativas em $30^\circ$ com 15 janelas.	42

Figura 5-8: Gráfico das estimativas em 30° com 30 janelas. ....	42
Figura 5-9: Gráfico das estimativas em 30° com 15 janelas, sem Arduino. ....	43
Figura 5-10: Gráfico das estimativas em -30° com 15 janelas, sem Arduino. ....	43
Figura 5-11: Gráfico das estimativas em 30° com 30 janelas, sem Arduino. ....	43
Figura 5-12: Gráfico das estimativas em -30° com 30 janelas, sem Arduino. ....	44
Figura 5-13: Montagem do sistema em um estúdio musical. ....	44
Figura 5-14: Gráfico das estimativas em 30° no estúdio.....	45
Figura 5-15 Gráfico das estimativas em 45° no estúdio. ....	45
Figura 5-16: Gráfico das estimativas em para azimutes variado em 2 kHz no estúdio.....	46
Figura 5-17: Gráfico das estimativas em 30°, 2 kHz, sem Arduino no estúdio. ....	46
Figura 5-18: Gráfico das estimativas em 30°, 1 kHz, sem Arduino no estúdio. ....	47
Figura 5-19: Gráfico das estimativas em 30° do Assovio no estúdio. ....	47
Figura 5-20: Gráfico das estimativas em 45° e 60° do Assovio no estúdio. ....	48
Figura 5-21: Gráfico das estimativas em 0° do Assovio no estúdio. ....	48
Figura 5-22: Uma estimativa ruim do exemplo mostrado na seção 5.1. ....	49
Figura 5-23: Demonstração de como o mesmo ângulo pode valer para duas fontes diferentes [10] .....	51

## LISTA DE SÍMBOLOS

### Símbolos Gregos

$\theta$	Azimute de incidência	[rad]
$\varphi$	Defasagem angular	[rad]
$\lambda$	Comprimento de onda	[m]

### Outros Símbolos

$d$	Distância entre elementos em um arranjo de sensores	[m]
$v$	Velocidade da onda	[m/s]
$f$	Frequência da onda	[Hz]
$t$	Tempo	[s]
$f_{ams}$	Frequência de amostragem do sinal	[Hz]
$\tau$	Atraso entre sinais de tempo contínuo	[s]
$K$	Atraso em tempo discreto	

### Siglas

ADC	<i>Analogic to digital converter</i>
AWGN	<i>Additive White Gaussian Noise</i>
A/D	Analógico para digital
DOA	Direction of arrival
FFT	<i>Fast Fourier Transform</i>
IIR	<i>Infinite Impulse Response</i>

FIR *Finite Impulse Response*

MATLAB Marca registrada da MathWorks, Inc.

SNR *Signal - Noise Ratio*

SPI *Serial Peripheral Interface*

# 1 AMBIENTAÇÃO

## 1.1 Motivação:

O som é a propagação de ondas mecânicas longitudinais, que se propaga de forma circuncêntrica. Tal propagação ocorre apenas em meios materiais, como sólidos, líquidos ou gasosos. Os sons consistem também numa combinação de sinais. Esses são constituídos por alguns elementos fundamentais como velocidade de oscilação ou frequência cuja medição é feita em Hertz (Hz) e amplitude do sinal que se mede em decibéis (dB).

Com base nesses elementos, seres vivos desenvolveram sistemas de emissão e captação de sinais sonoros. Esses sistemas permitem, por exemplo, localizar possíveis presas ou predadores com base nos sinais sonoros por eles emitidos.

Tendo em vista tais sistemas naturais, as aplicações em emissão e captação de sinais sonoros por robôs se mostram promissoras. Por exemplo, seria possível enviar pequenos robôs para procurar vítimas em meio a escombros de um desmoronamento; outra aplicação possível é para robôs que forem empregados no auxílio de pessoas idosas, sejam capazes de solicitar socorro caso escutem algum acidente doméstico. Além desses exemplos, um bom sistema de localização com base em sinais sonoros seria de grande auxílio para sistemas já existentes.

O problema da estimação da direção de chegada de um sinal sonoro é denotada pela sigla DOA (Direction of Arrival). O mecanismo tradicionalmente estudado pela comunidade acadêmica nessa área faz uso de sinais coletados por arranjo de sensores para calcular a direção da fonte sonora.

## 1.2 Objetivo:

O sistema DOA, já construído e desenvolvido por trabalhos anteriores, é composto por um circuito com dois microfones para a captação dos sinais sonoros e um microprocessador MSP430 da Texas Instruments para o processamento dos sinais recebidos e a estimação da direção da fonte sonora. Contudo, o estimador atual apresenta diversas inconsistências nos resultados. Sendo assim, o principal objetivo deste trabalho é analisar as possíveis causas dessas inconsistências. Com base nos trabalhos desenvolvidos pelos alunos Yuri Souza Reis [9], André de Medeiros Araújo [1], Igor Oliveira e Renato Oliveira [8] serão apresentados os fundamentos teóricos do sistema, metodologias previamente estabelecidas e parâmetros estabelecidos para o sistema.

Nesse contexto, será feita uma investigação mais profunda das possíveis causas de tais inconsistências, submetendo os sinais captados à uma análise de qualidade através de um programa desenvolvido no MATLAB e realizando testes em ambientes diferentes, além de averiguar a influência de parâmetros do sistema na precisão, esses serão melhores descritos na seção 1.3. Por fim, espera-se ao final deste trabalho apresentar o principal fator de erro do sistema e apresentar possíveis alternativas para a solução do mesmo.

É importante frisar que o aprimoramento do sistema não faz parte dos objetivos desse trabalho. Entretanto, espera-se que com base nos resultados obtidos o sistema possa ser aprimorado em trabalhos futuros para estimar a origem de uma fonte sonora com mais precisão.

### **1.3 Metodologia:**

A pesquisa abordada neste documento é do tipo experimental, sendo assim avaliaremos os principais elementos na propagação, captação e processamento dos sinais sonoros emitidos, bem como o emprego de diferentes técnicas utilizadas para a estimação da direção de chegada fundamentados em bibliografias anteriormente desenvolvidas.

Logo em seguida, testes serão realizados alterando as condições do estímulo sonoro, tais como: posição da fonte sonora, frequência emitida pela fonte, ou mesmo parâmetros do código de processamento do sinal. Testes serão realizados em um estúdio musical que possui bom isolamento acústico, para averiguar os efeitos como: eco, ruído externo e outros. Os resultados desses testes serão avaliados pela ferramenta desenvolvida nesse trabalho e serão apresentados em forma de gráficos. Após a análise, será possível determinar qual é o motivo do sistema apresentar inconsistências nos resultados.

### **1.4 Organização:**

O Capítulo 2 apresenta sobre o embasamento teórico deste trabalho, abordando o arranjo linear de microfones e os conceitos-chave que serão utilizados em capítulos posteriores. Como dito anteriormente, esse trabalho é uma continuação de um projeto previamente desenvolvido, citado nas referências bibliográficas. Sendo assim, grande parte do embasamento teórico aqui apresentado é comum com os trabalhos anteriores.

O Capítulo 3 aborda o hardware utilizado no projeto, apresentando uma explicação sobre o que foi empregado no projeto para alcançar o objetivo final, como sensores, processador, memória externa e módulos eletrônicos.

O Capítulo 4 aborda o algoritmo utilizado no projeto, explicitando o método empregado na estimação de direção de chegada. Este capítulo foi dividido em seções, abordando as etapas do processo. Discute-se também o efeito de limitações de hardware no processo de estimação.

O Capítulo 5 apresenta os resultados de ensaios realizados e as simulações feitas com a ajuda do MATLAB para verificar a qualidade do método proposto. Este capítulo também apresenta algumas limitações encontradas, bem como hipóteses testadas durante o projeto.

Por fim, o capítulo 6 apresenta as conclusões obtidas com o projeto proposto. Discutem-se também a possibilidade de trabalhos futuros e o que se pode melhorar para tornar o sistema mais eficiente para ser implementado.

## 2 EMBASAMENTO TEÓRICO

### 2.1 Arranjo de sensores

Para estimar a direção de chegada de um sinal sonoro é necessário que haja dois ou mais microfones dispostos de maneira adequada, uma vez que a estimativa se baseia no atraso do som captado por cada um dos microfones.

Para esse projeto os microfones foram dispostos dois microfones espaçados de distância de 15 *cm* entre o centro de cada um [9]. O ângulo de azimuth  $\theta$  corresponde à direção de chegada do sinal, e é formado pela interseção da direção da frente de onda do sinal sonoro  $s(t)$  com a reta perpendicular ao eixo dos microfones. A figura 2.1 apresenta de forma clara as convenções adotadas.

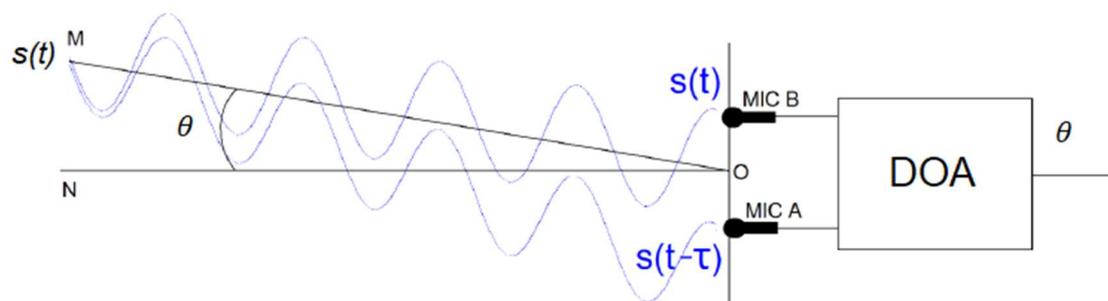
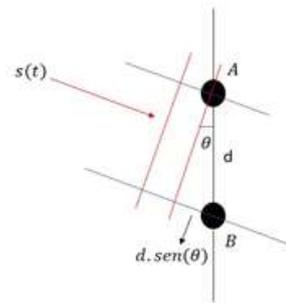


Figura 2-1: Arranjo linear uniforme dos microfones para estimativa [10].

Analisando a onda de forma bidimensional, ao incidir sobre o arranjo de microfones, um dos microfones recebe o sinal de forma defasada em relação ao outro, dependendo do ângulo da fonte sonora. Sendo assim, a diferença entre os caminhos que levam o sinal  $s(t)$  aos dois microfones distância da defasagem é dada por:  $d * \text{sen}(\theta)$ , como pode ser visto na figura 2-2.



**Figura 2-2: Frente de onda bidimensional e a indicação da diferença de percurso [10].**

O atraso temporal que cada microfone sofre, depende da distância de defasagem entre os microfones, e também depende da velocidade da onda. Na equação (2.1) a seguir é demonstrada essa formulação onde  $v$  é a velocidade do som,  $d$  é a distância entre os microfones,  $\theta$  é o ângulo de azimute e  $\tau$  é o atraso temporal.

$$\tau = \frac{d * \text{sen}(\theta)}{v}$$

**Equação (2.1)**

Com o atraso temporal também pode ser obtida a defasagem angular, que consiste em multiplicar o atraso temporal pela frequência angular, como mostra a equação (2.2).

$$\varphi = \omega * \tau = 2\pi f * \frac{d * \text{sen}(\theta)}{v}$$

**Equação (2.2)**

Como o som é uma onda, o atraso pode ser demonstrado também de acordo com o comprimento de onda  $\lambda$ , mostrando que, para um mesmo sinal, as componentes de frequências diferentes vão experimentar atrasos distintos. Decompondo a velocidade do som  $v$ , é obtida a equação (2.3).

$$\varphi = 2\pi \frac{d * \text{sen}(\theta)}{\lambda}$$

**Equação (2.3)**

Tomando o primeiro microfone como referência, a equação (2.3) é equivalente ao segundo microfone que está a apenas  $d$  de distância do primeiro sensor. Para fazer a análise, no caso de mais microfones, a referência continua a mesma, então a defasagem experimentada pelo terceiro microfone será o dobro do segundo, e a defasagem do quarto microfone será o triplo do segundo, e assim por diante [1]. Esse padrão se deve à distância que o microfone está com relação ao microfone de referência. Dessa forma, esse arranjo pode ser tratado como uma matriz, onde  $x_i(t)$  é a saída do  $i$ -ésimo microfone, em um total de  $M$  sensores, tomando a saída  $x_0(t)$  do microfone 0 como referência, dando a equação matricial (2.4).

$$\begin{pmatrix} x_0(t) \\ x_1(t) \\ x_2(t) \\ \vdots \\ \vdots \\ \vdots \\ x_{M-1}(t) \end{pmatrix} = \begin{pmatrix} 1 \\ e^{-j\varphi} \\ e^{-2j\varphi} \\ \vdots \\ \vdots \\ \vdots \\ e^{-(M-1)j\varphi} \end{pmatrix} \cdot s(t) + \begin{pmatrix} n_0(t) \\ n_1(t) \\ n_2(t) \\ \vdots \\ \vdots \\ \vdots \\ n_{M-1}(t) \end{pmatrix}$$

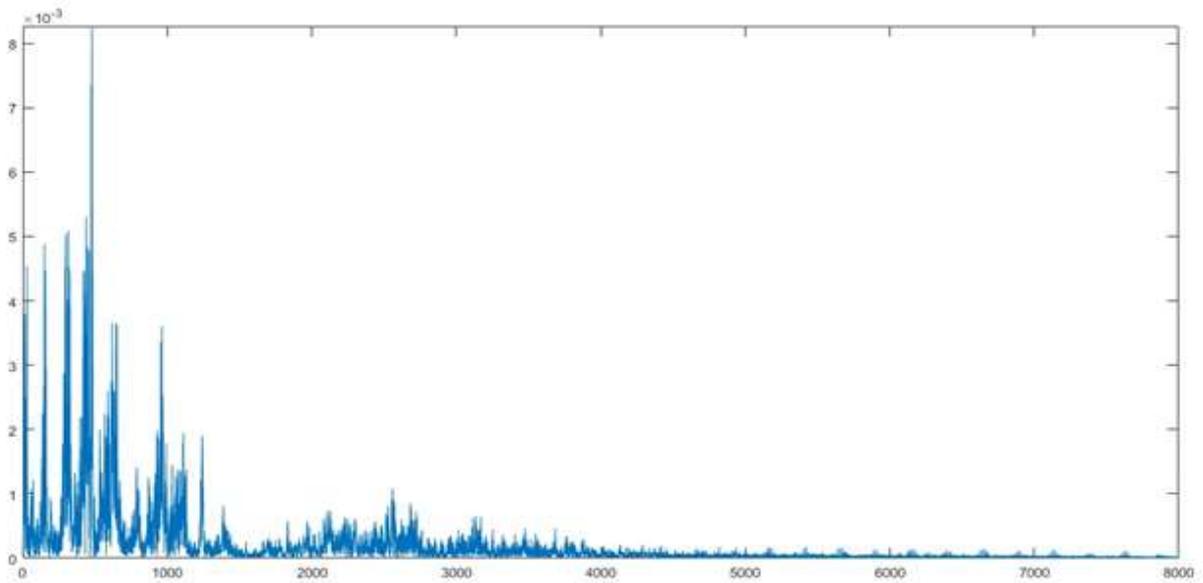
**Equação (2.4)**

Em cada microfone, juntamente com o sinal de entrada, há o ruído  $n(t)$  que permeia o próprio sistema, e foi adicionado à equação matricial. Como é fácil de concluir a equação acima só é válida se  $s(t)$  for um sinal em banda estreita. Com o modelo  $s(t) = b(t)e^{j\omega_0 t}$ , é preciso que  $b(t)$  tenha banda limitada [1].

## 2.2 Bandas de Frequência

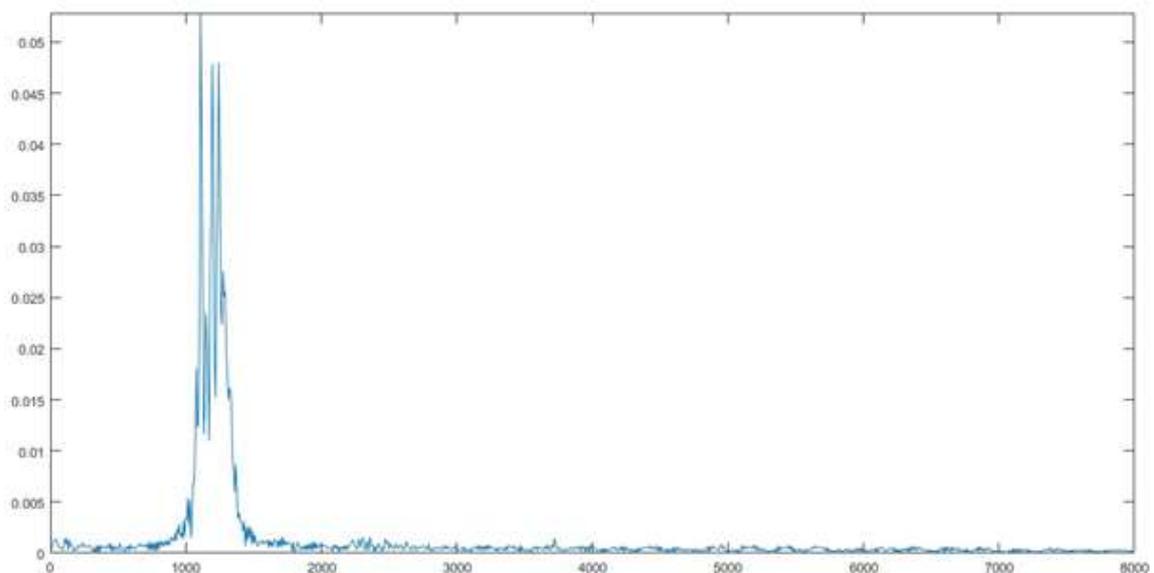
A frequência da voz humana é muito variada, ainda mais quando comparamos sexos distintos. A figura 2-3 mostra o perfil espectral da voz masculina, essa foi amostrada em 16 kHz e filtrada em 8 kHz, e possui muita energia na faixa que vai desde 100 Hz até 4 kHz, o que representaria, de certa forma, a banda larga deste sinal. Caso o sinal tenha seja de banda larga, o sistema precisará de um maior poder de processamento para fazer a estimativa de direção adequada, pois seria necessário fatiar o sinal em faixas de frequência.

A larga distribuição da energia sonora (banda larga) faz com que os microfones experimentem atrasos distintos resultando em uma estimativa prejudicada [1].



**Figura 2-3: Perfil espectral de uma voz masculina [1].**

Tendo isso em vista, o sinal incidente será modelado como banda estreita. Os testes iniciais foram feitos com frequência da fonte em  $f = 1 \text{ kHz}$ , fazendo uso de um gerador de frequências. Em seguida, as estimativas foram feitas com assovio, uma vez que esse varia pouco entre os seres humanos, mesmo entre homens e mulheres, e ainda se encontra em uma faixa de frequência relativamente pequena, como pode ser visto na figura 2-4.



**Figura 2-4: Perfil espectral de um assovio [1].**

## 2.3 Ambiguidade e distância entre sensores

Como visto na seção 2.1, a distância  $d$  entre os microfones faz parte do cálculo da estimativa (Equação 2.2) da DOA. Dessa forma, para garantir que tal distância seja corretamente escolhida faz-se necessário considerar um fator importante: o Aliasing Espacial.

### 2.3.1 Aliasing Espacial

O aliasing espacial ocorre quando o sistema detecta um atraso falso, ou quando não detecta atraso algum. Em tais casos, como o sistema se baseia no atraso do som captado pelos microfones, tem-se que a estimativa é calculada de forma equivocada. Para demonstrar esse efeito, observe a imagem a seguir:

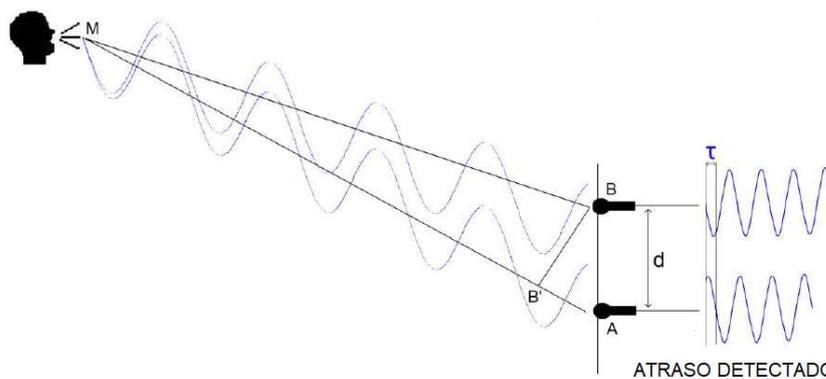


Figura 2-5: Situação normal em que o atraso é detectado [10].

Nessa situação o som chega simultaneamente em B e em B'. Como foi explicado na seção anterior o tempo extra necessário para o som chegar em A é denominado  $\tau$ . Esse parâmetro é medido em tempo, ou em distância se compararmos com o comprimento de onda  $\lambda$ . Na figura 2-5, o atraso é menor do que  $\lambda/2$ .

A distância B'A depende do espaçamento entre os microfones,  $d$ . Se  $d$  aumenta, B'A também aumentará, uma vez que  $B'A = d \cdot \text{sen}(\theta)$ . O problema do aliasing espacial aparece quando B'A aumenta até chegar ao ponto em que o atraso é maior do que  $\lambda/2$ . A figura 2-6 seguir exemplifica essa situação:

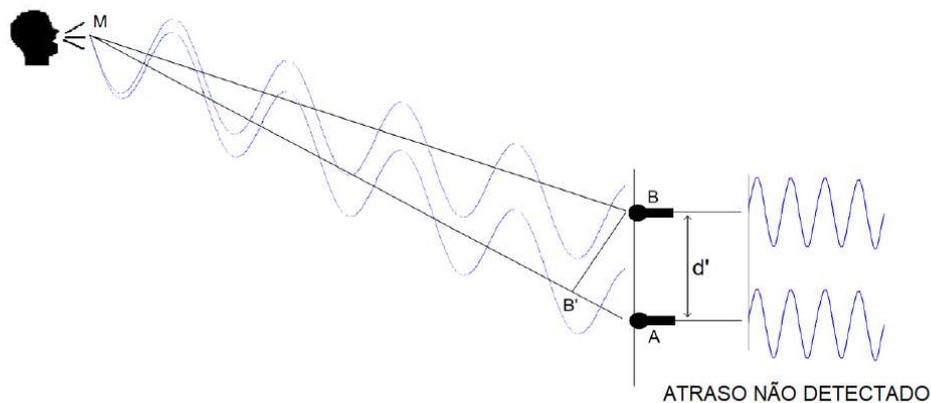


Figura 2-6: Situação em que o atraso é igual a  $\lambda$  [10].

Ver duas senoides em azul na porção direita da figura. Notar que a distância  $BA'$  é igual a um comprimento de onda, para o sistema os sinais parecem estar em fase, ou seja, ângulo de incidência é igual a zero. O atraso é o tempo necessário para o sinal percorrer a distância  $B'A$ . Se a distância  $B'A$  for igual ao comprimento de onda, tem-se que os sinais capturados pelos microfones será o mesmo. Portanto, mesmo havendo atraso entre os microfones, o sistema não será capaz de detectá-lo. Com base nos sinais adquiridos pelos microfones, é possível provar que o atraso detectado não será correto.

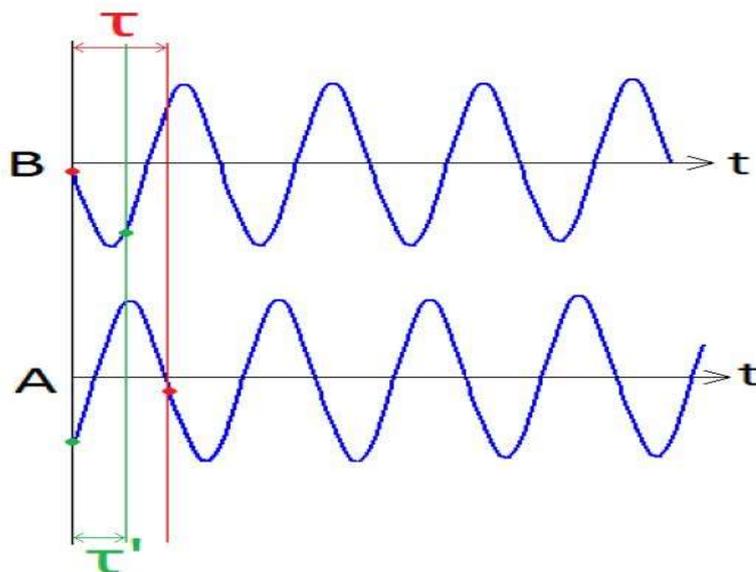


Figura 2-7: Dois atrasos que podem ser detectados caso ocorra Aliasing [10].

De acordo com a figura 2-7, o atraso verdadeiro é o indicado como vermelho. A fonte sonora está mais próxima do microfone B, logo o sinal chega em B antes de chegar em A.

Contudo, se esse sinal passa por um sistema cujo objetivo é determinar o atraso, o sistema irá retornar o atraso  $\tau'$ . Onde  $\tau'$  é menor do que  $\lambda/2$ , sendo assim o sistema irá errar a identificação do atraso real [10]. De modo que o estimador irá concluir que a fonte sonora está mais próxima do microfone A do que do microfone B. Portanto, a condição que deve ser obedecida para evitar o aliasing espacial é:

$$B'A \leq \frac{\lambda}{2}$$

**Equação (2.5)**

No ponto crítico de  $B'A = \lambda/2$ , é possível determinar o ângulo máximo que o estimador pode alcançar sem espacial aliasing. Para este caso em particular, supondo que a frequência da fonte sonora seja de 2 kHz, tem-se:

$$B'A = d \cdot \text{sen}(\theta)$$

**Equação (2.6)**

$$\frac{v}{f \cdot 2} = d \cdot \text{sen}(\theta)$$

**Equação (2.7)**

$$\theta = \text{sen}^{-1}\left(\frac{v}{f \cdot d \cdot 2}\right) = \text{sen}^{-1}\left(\frac{343}{2000 * 0,15 * 2}\right) = 34,87^\circ$$

**Equação (2.8)**

Logo, não será possível fazer qualquer estimativa para ângulos maiores do que  $34,17^\circ$  usando uma frequência de 2 kHz, como está ilustrado na figura a seguir.

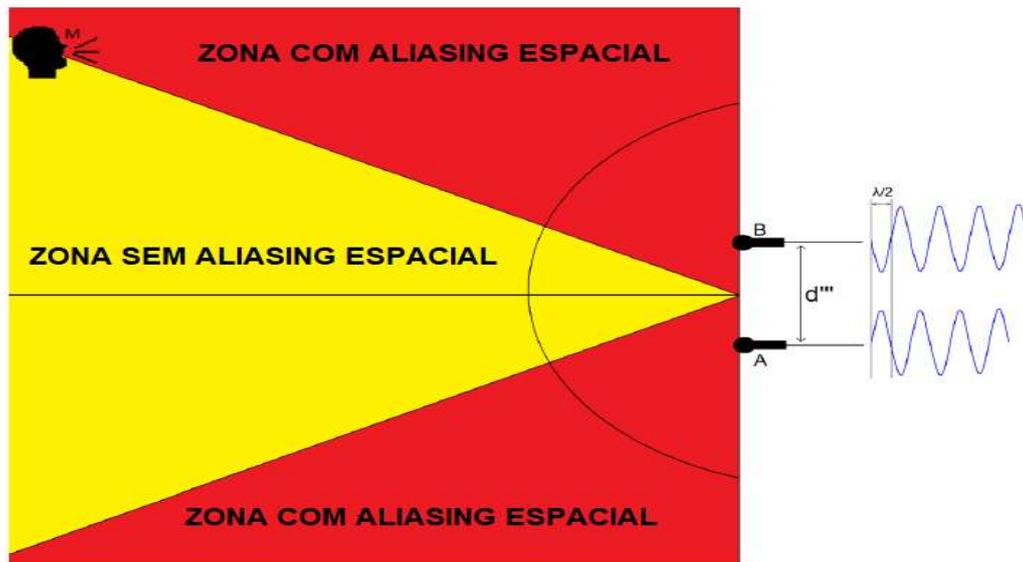


Figura 2-8: Limite da zona sem Aliasing é  $\pm 34,87^\circ$ , valores além do máximo terão Aliasing [10].

Contudo, uma vez que o objetivo desse projeto é ter uma estimativa que vá de  $-90^\circ \leq \theta \leq 90^\circ$ . Logo, se faz necessário encontrar um meio de evitar o aliasing espacial de modo que a distância entre os microfones deve cumprir o seguinte critério para que não ocorra aliasing espacial:

$$d \leq \frac{\lambda}{2}$$

Equação (2.9)

Para se obter a distância máxima entre os microfones é necessário encontrar o valor mínimo de  $\lambda$ . Por estarmos considerando a faixa espectral do assovio, consideramos valores entre  $1 \text{ kHz}$  e  $2 \text{ kHz}$ , como demonstrado na seção 2.3 [10].

$$d \leq \frac{\lambda_{min}}{2} ; d \leq \frac{343}{2 * 2000} = 8,57 \text{ cm}$$

Equação (2.10)

Assim sendo, temos que para uma frequência de  $2 \text{ kHz}$  é necessário que a distância máxima entre os microfones seja de  $8,57 \text{ cm}$ . Porém, quanto maior for a distância  $d$  entre os microfones, melhor será a precisão da estimativa, isso será melhor abordado na seção 4.1.

Uma outra solução possível é reduzir a frequência da fonte para  $1 \text{ kHz}$ , isso permitiria aumentar a distância para  $15 \text{ cm}$ . O máximo seria  $17 \text{ cm}$  [1].

Vale ressaltar que uma pequena variação na distância entre os microfones pode gerar um erro grosseiro na estimativa. Logo, para mitigar esse erro, os microfones foram fixados com parafusos em uma placa de madeira separados por uma distância de 15 cm. As figuras 2-9 e 2-10 mostram, respectivamente, o sistema montado e o diagrama de blocos do circuito.

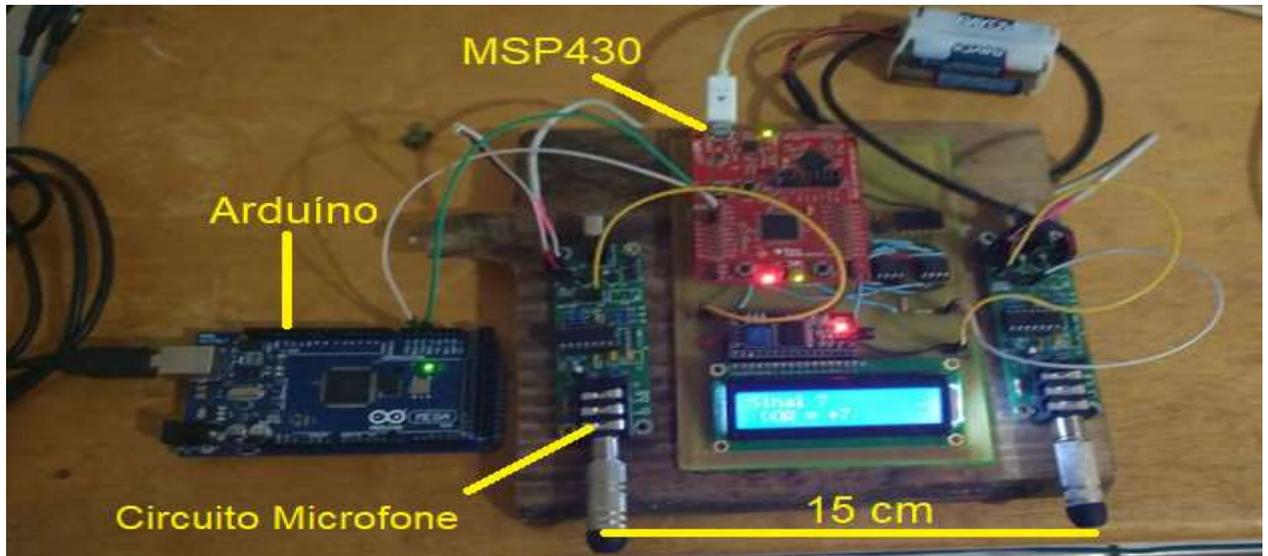


Figura 2-9: Sistema de estimativa DOA montado.

Para facilitar seu emprego e beneficiar a mobilidade, o sistema prevê interface para um módulo *Bluetooth*. Assim, seria possível adquirir sinais, fazer estimativas e depois enviar os dados e resultados para o PC, com finalidade de conferência e análise dos resultados. O módulo *Bluetooth* HC-05 usado com esta finalidade se revelou um certo entrave, por diversas vezes demandando por um pareamento. Para simplificar os ensaios, foi usado um Arduino para simular esse canal *Bluetooth* e enviar serialmente os dados para o PC. Na verdade, poder-se-ia usar diretamente a comunicação serial da *Launchpad* com o PC, porém o programa já estava todo preparado para o HC-05 e a melhora desta solução foi preterida por outros problemas.

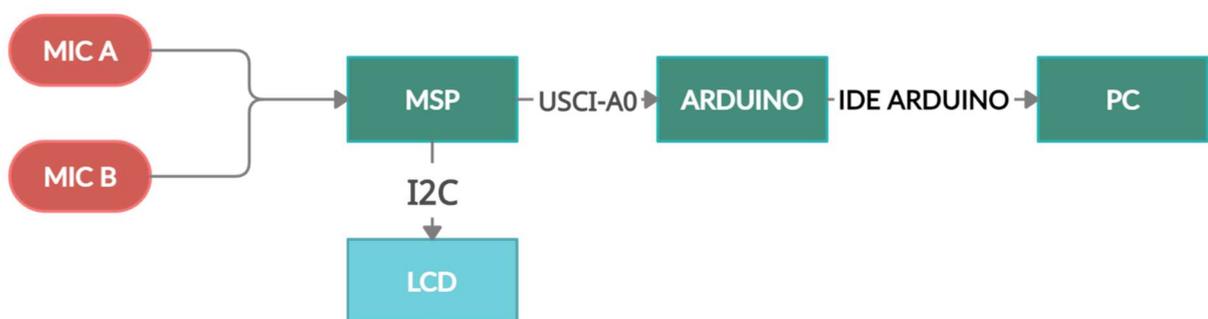


Figura 2-10: Diagrama de blocos do circuito.

O programa possui dois modos de operação principais. No modo 0, não há transmissão dos dados armazenados na memória para o Arduino, há apenas a apresentação do DOA final na tela do LCD. No modo de operação 1, todos os dados que forem armazenados na memória são transmitidos ao Arduino que os envia para o PC. Usa-se o monitor serial da IDE do Arduino para receber e visualizados tais dados. Daí, eles podem ser salvos num arquivo texto para serem analisados pelo MATLAB.

## 2.4 Filtros digitais:

A aquisição do sinal sonoro pelos microfones e o processo de conversão analógico/digital insere ruído. Neste trabalho modelamento tal ruído como sendo Aditivo Gaussiano-branco (AWGN). Sua presença no sinal original, pode causar erros consideráveis na estimativa da direção de chegada do sinal sonoro. Dessa forma, o uso de filtros digitais faz-se necessário para que o ruído seja atenuado e a razão sinal-ruído seja elevada. Logo, é fundamental que seja feita uma análise do perfil espectral do típico sinal sonoro de um assovio [9].

Por fim, definiu-se um filtro adequado para o tratamento do sinal de assovio. Com base nos parâmetros de interesse, o filtro escolhido possui as seguintes características: filtro passa-baixa, com frequência de corte de  $3\text{ kHz}$ , projetado segundo a classe IIR (resposta infinita ao impulso, do inglês Infinite Impulse Response) Butterworth de 4ª ordem. A ordem é intencionalmente baixa para não onerar o processamento. Esse filtro foi projetado usando ferramentas do MATLAB.

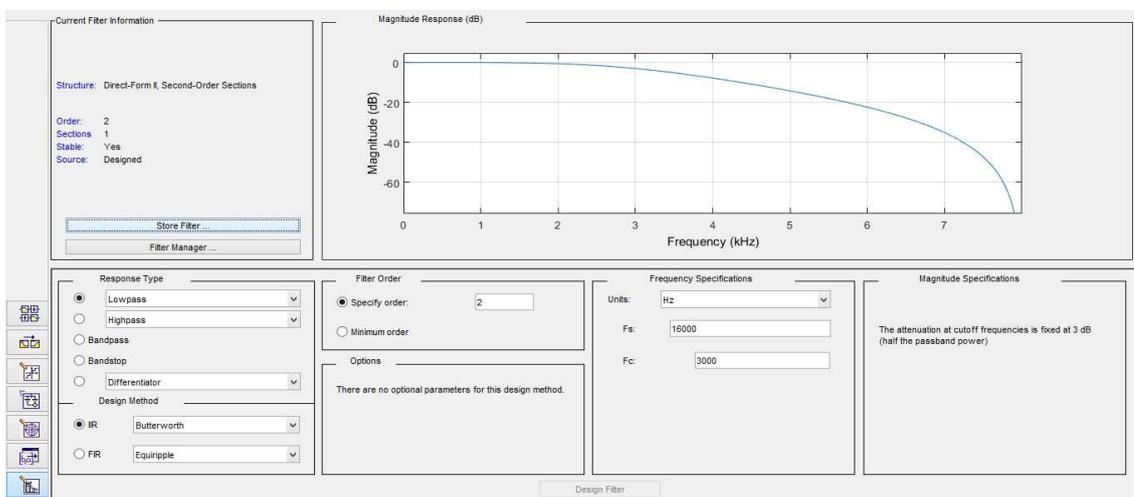


Figura 2-11: Imagem principal do projeto do filtro IIR em ambiente MATLAB [9].

### 3 HARDWARE

Neste capítulo é apresentada toda a parte de projeto de hardware do sistema, englobando as etapas de orientação e movimentação da plataforma, e também a aquisição, amplificação, digitalização e processamento do sinal sonoro. Dentre os eletrônicos, serão abordados o MSP 430 e seus protocolos, a memória externa, o microfone e o circuito pré-amplificador.

#### 3.1 MSP430

A família TI MSP430™ de microcontroladores de potência ultrabaixa consiste em vários dispositivos com conjuntos de periféricos direcionados para uma variedade de aplicações. A arquitetura, combinada com extensos modos de baixo consumo de energia, é otimizada para alcançar maior vida útil da bateria em aplicações de medição portáteis, além de possibilitar ao usuário elaborar diversos algoritmos bem como implementar diversas funções. Esses são programados em linguagem C, através do software Code Composer Studio, desenvolvido e disponibilizado pela Texas Instruments. O microcontrolador possui uma poderosa CPU RISC de 16 bits, registradores de 16 bits e geradores constantes que contribuem para uma excelente eficiência do código. O microcontrolador possui quatro temporizadores de 16 bits, um conversor de 12 bits de alto desempenho (ADC), quatro unidades para comunicação serial, um multiplicador de hardware, DMA, um módulo RTC com recursos de alarme e 63 pinos [4].

#### 3.2 Memória externa

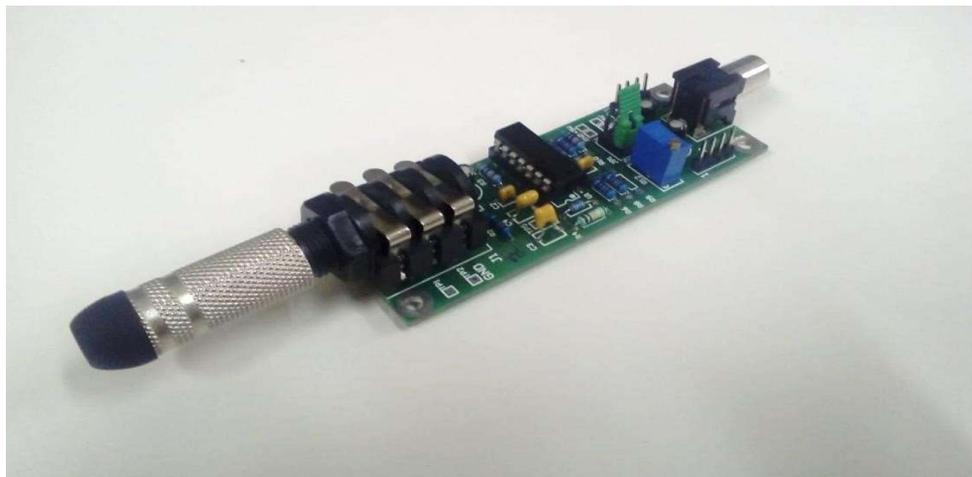
Para auxiliar a memória interna do MSP430, foi utilizada a comunicação serial periférica (SPI) da plataforma para enviar e salvar os dados recém convertidos pelo conversor A/D em uma memória SRAM. A memória utilizada foi a 23LC1024, demonstrada na figura 3.4, fabricada pela Microchip. Esse periférico possui 128 KB de memória, permitindo assim armazenar os dados convertidos dos dois microfones até que sejam processados [16]. O projeto faz uso de duas dessas memórias, assim sua capacidade é de armazenar até 256 K amostras de 8 bits. O barramento SPI se baseia no processo Mestre e Escravo, onde o mestre é o que gerencia o barramento, responsável por gerar a linha de sincronismo (SCK) além de inicializar e finalizar as transações, enquanto o escravo só se atua quando acionado pela linha de seleção. A comunicação SPI é bidirecional, enquanto o mestre envia os dados pela linha MOSI (*Master Output Slave Input*), o escravo envia dados para o mestre pela linha MISO (*Master Input Slave Output*) [6]. No projeto foram feitas

algumas funções de escrita e leitura na memória, além de configuração da SPI para facilitar o manuseio do programa.

A conexão do MSP430 com a memória será feita por serial SPI, de acordo com os *datasheets* tanto do MSP430 e da memória utilizada. Assim que o dado é recebido pelo microfone ele é digitalizado e então escrito na memória na forma de vetor. Ao fim da etapa de digitalização, ele está pronto para ser lido e processado diretamente da memória. Inicia-se, então, seu tratamento e a posterior estimação da direção de chegada.

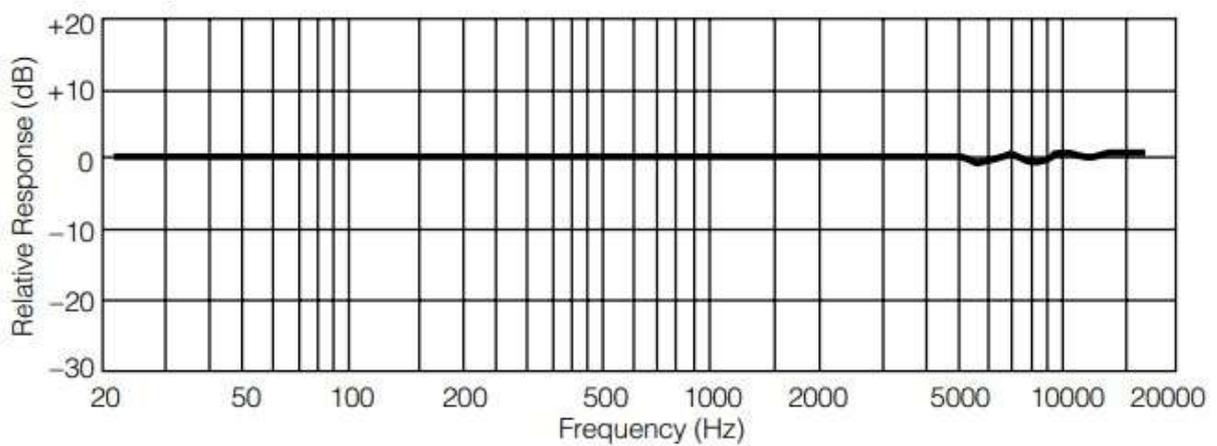
### 3.3 Microfone e circuito pré-amplificador:

Neste trabalho, foram utilizados microfones previamente para um projeto de detecção da direção de chegada (DOA) e aproveitados para o presente trabalho [9].



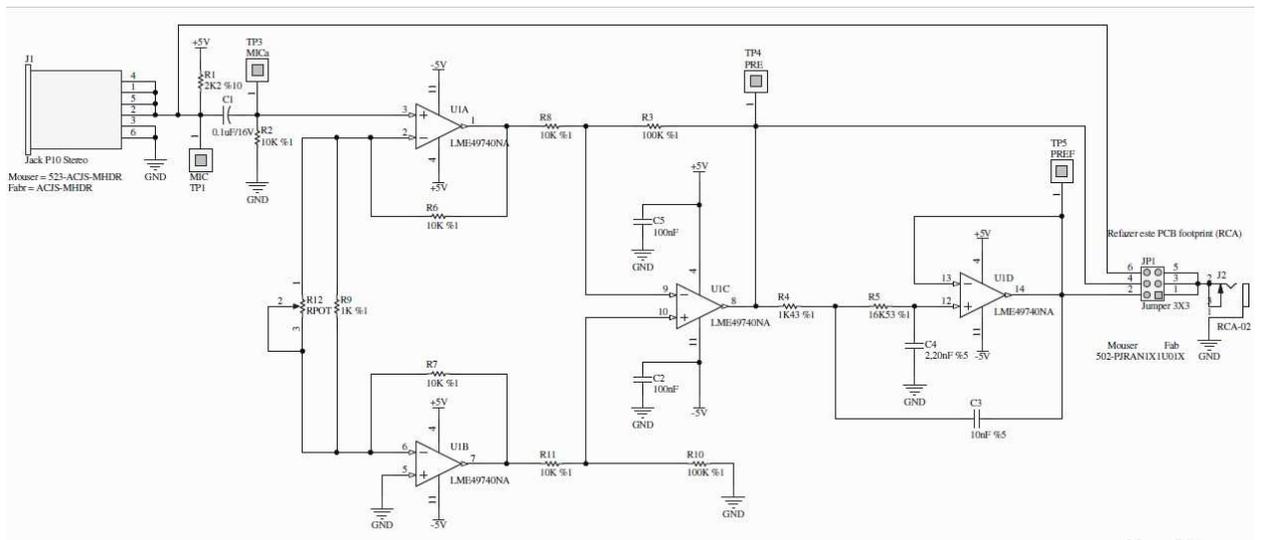
**Figura 3-1: Microfone acoplado à placa do circuito utilizado [9].**

Esses microfones são do tipo omnidirecionais, ou seja, eles são capazes de captar sinais oriundos de todas as direções de maneira igual. Além disso, apresentam SNR de 62 dB, a curva de resposta em frequência do microfone é linear para a banda de 20 Hz até 20 kHz, o que os tornam adequados para a captação da voz humana. Na figura a seguir fica evidenciado que a resposta em frequência é constante na banda estabelecida.



**Figura 3-2: Resposta de magnitude versus frequência para o microfone de eletreto usado no presente trabalho [9].**

O circuito de pré-amplificação do sinal do microfone foi montado em uma placa de circuito impresso utilizando amplificadores operacionais contidos no chip LME49740, que possui quatro amplificadores embutidos [9]. O sinal recebido passa para o amplificador de instrumentação construído com 3 amplificadores. Em seguida, é submetido a um filtro passa-baixas de frequência de corte de  $4,378\text{ kHz}$  que também realiza o papel de buffer de saída. Esta saída, após um divisor resistivo, é conectada à entrada do conversor analógico/digital do Arduino para digitalização do sinal. O circuito descrito pode ser visto na figura 3-3.



**Figura 3-3: Circuito do estágio de pré-amplificação [9].**

É importante comentar que esse pré-amplificador foi desenvolvido, originalmente, para operar com fontes complementares. No presente uso, emprega-se 4 baterias de 1,5 V para alimentar os operacionais com  $\pm 3$  V. Porém, o conversor ADC do MSP não trabalha com tensão negativa, assim, o -3V do pré-amplificador é conectado à terra da *Launchpad*. Com isso o sinal tem valor médio de 3 V, excursionando de 0 até 6V. Usa-se então um divisor resistivo para reduzir esta excursão à metade [9]. O Circuito de alimentação dos microfones pode ser visto na figura 3-4. Já na figura 3-5, pode-se observar como é feita a conexão dos microfones com os pinos do MSP430.

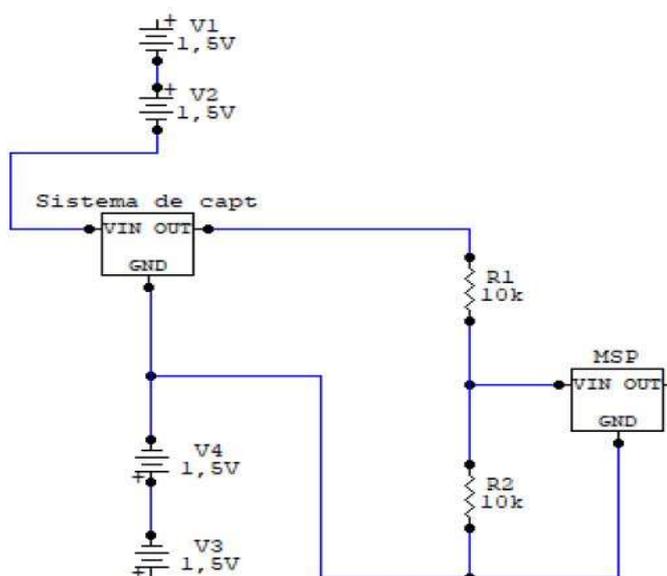


Figura 3-4: Alimentação do circuito [9].

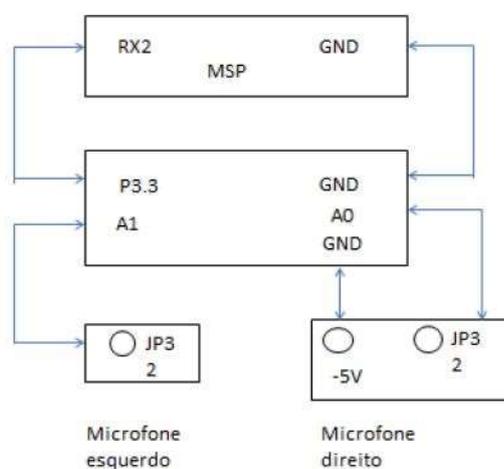


Figura 3-5: Ligação entre microfones, placa de circuito e MSP430 [9].

## 4 SOFTWARE PARA ESTIMATIVA DOA

Nesse capítulo será feita uma rápida descrição das diversas etapas usadas na estimação da direção de chegada do sinal sonoro.

### 4.1 *Firmware*

Na criação do código principal foram listadas as funções desempenhadas pelo sistema e os scripts auxiliares responsáveis por executar tais funções. A inicialização do código, então, carrega o valor das constantes utilizadas ao longo da execução e configura elementos de controle tais como pinos de entrada e saída, botões, timers, interrupções, conversor analógico-digital (A/D), transmissão serial (SPI) do MSP430 com a memória externa [1].

A configuração de pinos e timers é feita através da configuração de portas programáveis (GPIO, do inglês *General Purpose Input/Output*). Dois botões foram configurados para reinicializar a execução do sistema e selecionar o modo de operação do mesmo. Leds foram utilizados para indicar etapas de processamento do sistema [1].

Para que o processo de conversão analógico-digital tenha melhor desempenho, faz-se uso do conversor A/D junto com o recurso de DMA (do inglês, *Direct Memory Access*). Dessa forma, o conversor fará a digitalização do sinal recebido, salvando-o diretamente na memória do MSP430. O controlador de DMA libera o barramento da CPU e permite armazenar diretamente o valor digitalizado na memória do MSP430 sem a necessidade de usar a CPU para essa tarefa. Além disso, configurou-se os timers TA0.1 para o disparo do ADC.

Para a apresentação de etapas de processamento, bem como o resultado final DOA, utilizou-se um display LCD.

### 4.2 Método de estimação de DOA

A fonte do assovio definida pelo azimute  $\theta$  se encontra em uma direção específica, podendo variar entre  $-90^\circ \leq \theta \leq 90^\circ$ . Em todos os ensaios a fonte sonora será posicionada a frente dos microfones. A figura 4-1 mostra os ângulos possíveis a serem estimados e a figura 4-2 mostra a disposição dos microfones adotada.

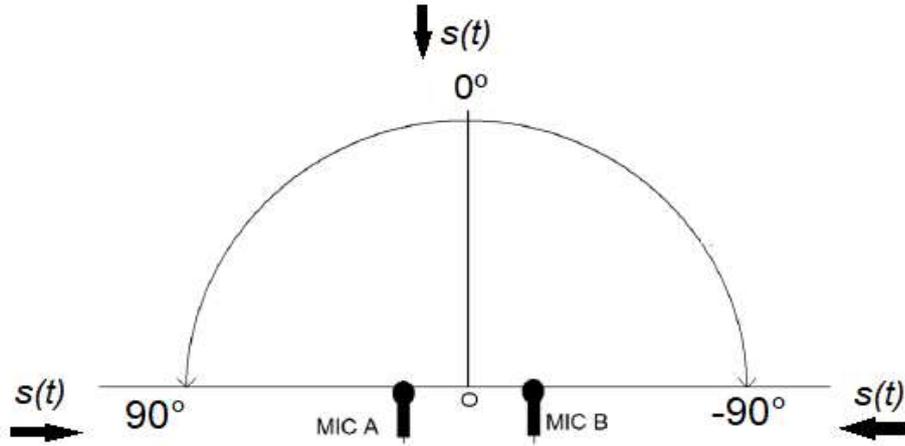


Figura 4-1: Ângulos possíveis para a estimativa [10].

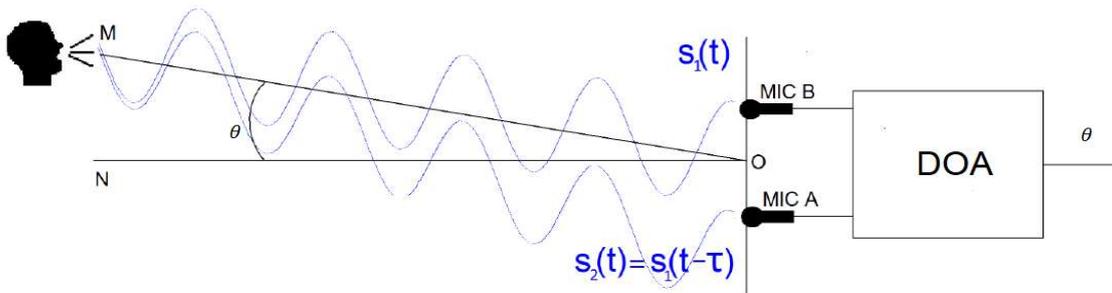


Figura 4-2: Montagem proposta de arranjo com dois microfones [10].

As equações (4.1) e (4.2) definem o sinal recebido pelos microfones como  $m_1(t)$  e  $m_2(t)$ , cada um multiplicado pelo respectivo ganho  $A_1$  e  $A_2$  [1]. Percebe-se que o sinal está defasado em  $\tau$  no sinal referente ao segundo microfone. As equações também consideram o ruído gaussiano branco, definido como  $n(t)$ , e que é intrínseco ao sistema.

$$m_1(t) = A_1 s(t) + n(t)$$

Equação (4.1)

$$m_2(t) = A_2 s(t - \tau) + n(t)$$

Equação (4.2)

Para que os sinais analógicos captados pelos microfones possam ser tratados por um microprocessador, eles precisam passar por um processo de conversão analógico-digital (A/D), tornando-se sinais digitais em tempo discretos em uma taxa de amostragem definida por:

$$f_{ams} = \frac{\text{quantidade de amostras}}{\text{tempo}}$$

**Equação (4.3)**

Para poder estimar o azimute, esse sinal deve ser concentrado em uma janela de tempo com  $n$  amostras. Assim que digitalizado, cada amostra pode ser salva em um vetor, facilitando a leitura.

Para se relacionar a  $k$ -ésima amostra em tempo discreto com o atraso temporal, uma análise dimensional é feita, mostrado na equação (4.4), onde  $k$  é o atraso relativo entre os sinais, em tempo discreto.

$$\tau_k = \frac{k}{f_{ams}} = \frac{\text{quantida de amostras}}{\frac{\text{quantida de amostras}}{\text{tempo}}} = \text{tempo}$$

**Equação (4.4)**

Desenvolvendo a equação acima com o atraso temporal, demonstrada na equação (2.1), é obtido a relação do azimute  $\theta$  com o atraso  $k$ , descrita conforme a equação (4.5):

$$\tau_k = \frac{k}{f_{ams}} = \frac{d \cdot \text{sen}(\theta)}{v}$$

**Equação (4.5)**

Com a finalidade de encontrar como resultado final o ângulo de azimute  $\theta$ , a equação (4.5) pode ser para isso da seguinte forma:

$$\theta = \text{sen}^{-1}\left(\frac{k \cdot v}{d \cdot f_{ams}}\right)$$

**Equação (4.6)**

Onde  $v$  é a velocidade do som (343 m/s),  $d$  é a distância entre os dois microfones e  $f_{ams}$  é a taxa de amostragem do sinal. Se  $k$  for corretamente estimado, a direção de chegada do sinal sonoro será dada por  $\theta$ .

A partir do Teorema de amostragem de Nyquist tem-se que um sinal de banda limitada pode ser reconstruído a partir de um determinado número de amostras igualmente separadas, caso a frequência de amostragem seja igual ou maior do que o dobro da frequência máxima do sinal.

$$f_{ams} = 2 \times f_{m\acute{a}x}$$

**Equação (4.7)**

Caso o Teorema da amostragem não seja cumprido, ocorrerá o fenômeno denominado aliasing temporal. Tal fenômeno implica em mudanças na forma do espectro do sinal e perdas de informação.

A banda de frequência de um assóvio é estreita, com seu valor máximo em torno de  $2\text{ kHz}$ , logo o valor mínimo para a frequência de amostragem é  $4\text{ kHz}$ . Entretanto, como pode ser visto na equação (4.6), o ângulo de azimute é inversamente proporcional a frequência de amostragem, isso significa que quanto maior for  $f_{ams}$ , melhor será a resolução para a obtenção desse ângulo. Logo uma maior frequência de amostragem é desejada. Por fim, optou-se por uma frequência de amostragem igual a:

$$f_{ams} = 50\text{ kHz}$$

**Equação (4.8)**

### **4.3 Etapas da estimativa**

Para a realização dessa estimação com a menor taxa de erro possível, após o sinal ser digitalizado e salvo em vetores, o MSP430 pode tratar os dados convertidos através de codificação programada em C. Todo processo de estimação feito pelo código que roda no MSP430 é transmitido para PC, permitindo assim que a ferramenta MATLAB seja usada para a verificação e validação do que foi feito no MSP430 [1]. O processo de estimação do DOA consiste em:

1. Aplicação de filtros digitais;
2. Identificação da faixa de maior energia do sinal;
3. Estimativa realizada com sinais filtrados.

### 4.3.1 Primeira etapa do tratamento: aplicação de filtro digital

Como visto na seção 2.4, o ruído e a banda do sinal devem ser atenuados para que a estimativa não seja prejudicada. Neste projeto, são utilizados dois filtros digitais com tais finalidades. O primeiro deles se trata de um processo estatístico, chamado de “Média Móvel”, que se trata de um filtro FIR – resposta finita ao impulso, do inglês *Finite Impulse Response* – que determina o valor de cada amostra de um vetor sendo igual à média entre  $n$  valores adjacentes a ele, como mostra a equação (4.9). Esses  $n$  valores são determinados pelo grau da média móvel utilizado era de tamanho 4 [9].

$$\text{Média Móvel} = \frac{1}{n} \sum_{i=1}^n p_i$$

Equação (4.9)

Esse filtro de média móvel é aplicado pelo programa nos dados assim que são convertidos pelo conversor analógico/digital, filtrando assim um pouco do ruído de alta frequência do sinal captado pelo microfone. Ele é salvo de volta na memória externa.

Depois de finalizada a conversão A/D, com os dados já filtrados pela média móvel salvos na memória externa. Então, o sinal é submetido a um outro filtro, denominado IIR, abordado na seção 2.4. O filtro foi projetado e testado inicialmente no MATLAB para a análise dos dados e então sua função de transferência foi passada diretamente para o programa do projeto [9]. A finalidade deste filtro é a de reduzir a banda do sinal. Uma vez passados pelos dois filtros, os dados são armazenados na memória e então podem avançar para a próxima etapa do tratamento.

### 4.3.2 Segunda etapa do tratamento: identificação da faixa de maior energia

A faixa de maior energia captada pelos microfones é precisamente a faixa que interessa para a estimação da DOA, pois é nessa que está o sinal sonoro a ser trabalhado. Ao se encontrar essa faixa, os dados contidos são passados para outros vetores para que sejam manipulados diretamente na memória do MSP430, não sendo mais necessário ler os dados da memória externa, melhorando assim o tempo de processamento dos dados, enquanto o resto do sinal pode ser descartado.

É importante lembrar que o pré-amplificador usado no projeto já faz uma filtragem para reduzir a energia nas bandas que não interessam ao projeto.

Para encontrar a faixa de interesse, tomou-se como referência o vetor que contém o sinal em tempo discreto do primeiro microfone. Inicialmente é calculada a média dos valores obtidos e então calcula-se o valor absoluto de cada elemento do vetor, subtraído do valor médio do sinal, em seguida é determinado um limiar para detectar o início da faixa de maior energia. A partir desse ponto é utilizada uma quantidade de dados fixa (janela de tempo) determinada a partir do início da faixa para realizar os cálculos.

### 4.3.3 Terceira etapa do tratamento: estimativa realizada com sinais filtrados

Para obter a defasagem em tempo discreto dos dois sinais obtidos pelos microfones é necessário verificar a similaridade entre esses sinais. Para isso é calculada a diferença absoluta média entre os dois sinais, enquanto um deles se desloca no tempo. Será visto adiante que usamos a diferença quadrática média [1].

Para exemplificar o método, considere duas senoides de mesma amplitude, ambas com frequência de  $1\text{ kHz}$ , com taxa de amostragem de  $50\text{ kHz}$  e 400 amostras (janela de tempo), com defasagem entre elas de 5 pontos, como mostrado na figura 4-3.

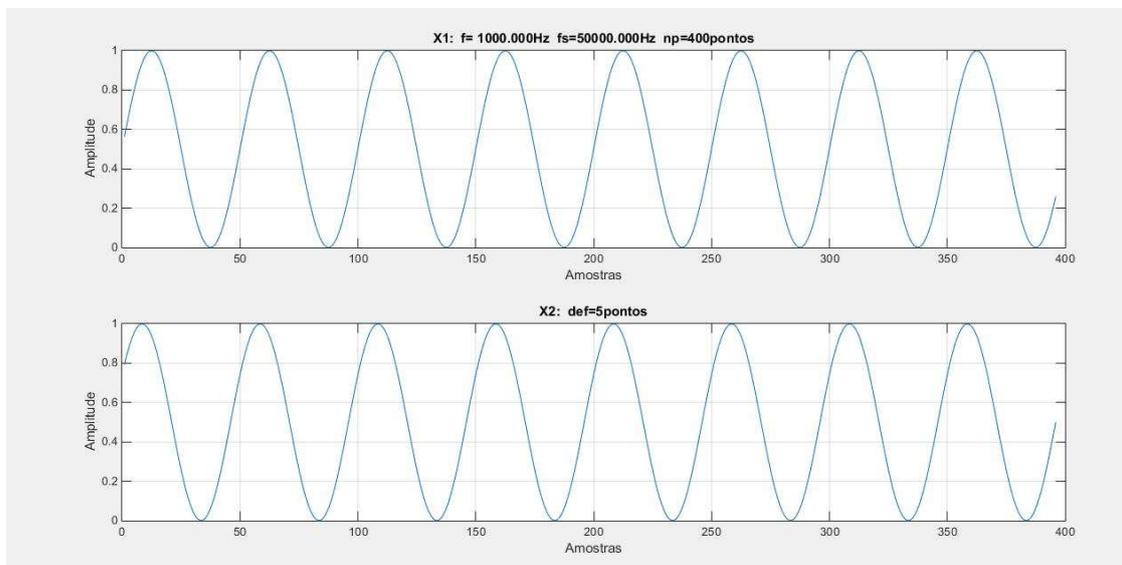
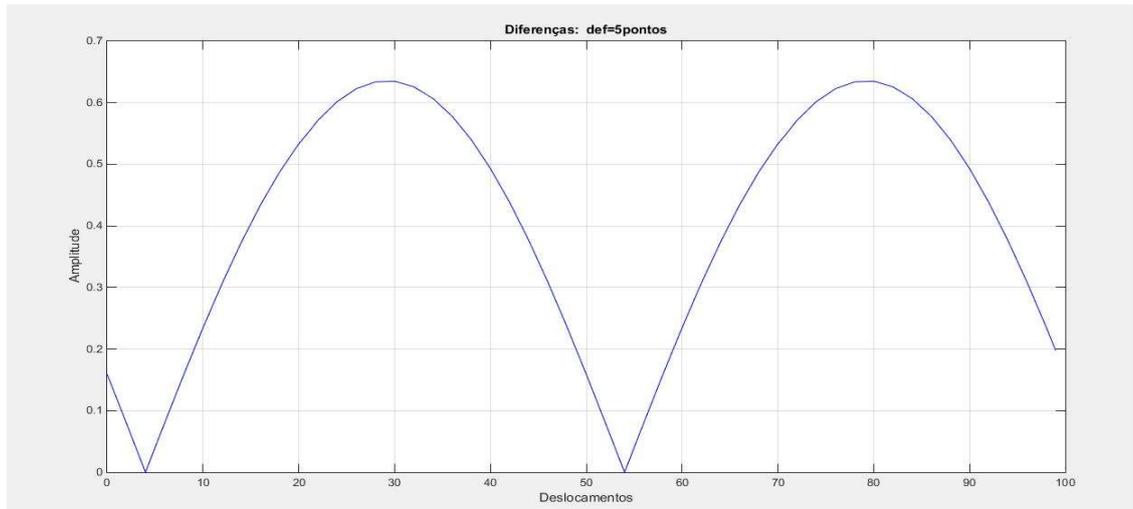


Figura 4-3: Senoides defasadas.

Ao fazermos o gráfico da diferença absoluta média entre as duas senoides ponto a ponto, tem-se:



**Figura 4-4: Gráfico da diferença das senoides.**

O tamanho do atraso entre os dois sinais é dado pelo menor valor do gráfico das diferenças, que se repete periodicamente. Esse é o valor de  $k$  necessário para determinar o azimuth da fonte sonora. Isso pode ser implementado diretamente pelo MSP430 através do cálculo da norma euclidiana. A norma euclidiana de um vetor é descrita como a raiz quadrada da soma dos valores absolutos elevados ao quadrado de cada elemento do vetor, conforme demonstra a equação (4.10):

$$|\vec{v}| = \sqrt{\sum_{i=1}^n |v_i|^2}$$

**Equação (4.10)**

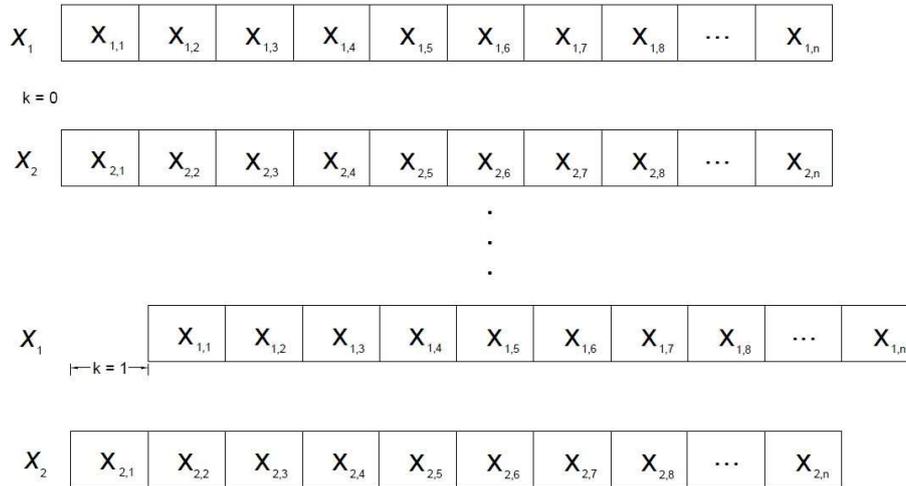
Para identificar a diferença entre os vetores que armazenam os dados digitalizados, a equação (4.10) é definida como:

$$\frac{1}{n} |\vec{x}_1 - \vec{x}_2| = \frac{1}{n} \sqrt{\sum_{i=1}^n |x_{1,i} - x_{2,i}|^2}$$

**Equação (4.11)**

onde  $x_{1,i}$  e  $x_{2,i}$  representam os  $i$ -ésimos elementos dos vetores que armazenam, respectivamente, os dados do primeiro e segundo microfone. O resultado da equação (4.11) fornece um valor de erro, também chamado de Erro Quadrático Médio, que quantifica a

semelhança entre os dois vetores. Para obter a medida de correlação entre esses dois sinais é necessário determinar o erro quadrático médio entre versões deslocadas dos vetores obtidos. Para cada deslocamento  $k$  entre os vetores, tem-se um valor calculado do erro que é salvo em outro vetor.



**Figura 4-5: Representação da operação de correlação efetuada [10].**

Dos valores armazenados nesse vetor que refletem os valores dos erros calculados, o que tiver menor resultado demonstra maior proximidade entre os dois sinais. Para o propósito deste trabalho busca-se justamente pelo elemento  $k$  que contém o menor valor em módulo dentre os valores desse vetor. Esse valor  $k$  que será utilizado na equação (4.6) para poder estimar o azimute do sinal de chegada [1].

#### 4.3.3.1 Juiz

Para inspecionar a qualidade das estimativas definiu-se um limiar denominado como ‘juiz’, suas funções são: averiguar um se há o mínimo de estimativas válidas para serem comparadas, limitar a excursão acima e abaixo da moda para calcular a DOA média. Esse processo será melhor abordado na seção 5.1.

#### 4.4 Resolução do estimador e limitações

Como foi dito anteriormente, a taxa de amostragem dos sinais é importante para a estimação de DOA por correlação. Com o conversor A/D digitalizando o sinal a uma taxa de  $f_{ams} = 25 \text{ kHz}$  para cada microfone, totalizando assim  $f_{ams} = 50 \text{ kHz}$ ; tomando  $k = 1$  e com os microfones espaçados na distância mínima de  $15 \text{ cm}$ , é possível determinar o tamanho da resolução do sistema, sendo esse igual a:  $\theta = 5,248^\circ$ , ou seja, a diferença entre

os ângulos estimados nunca será menor de  $5,248^\circ$ . A precisão do sistema é diretamente proporcional a taxa de amostragem, sendo assim, é possível aumentar a precisão do sistema ao aumentarmos a taxa de amostragem, outra técnica que possibilita o aumento da precisão é a interpolação spline cúbica, assim, de forma artificial, dobra-se a quantidade de amostras (artificialmente dobra-se a frequência de amostragem) e melhora-se a resolução [1]. Outra alternativa, como visto na seção 2.3 é aumentar o valor de  $d$  para melhorar a precisão, entretanto é necessário manter o critério de meio comprimento de onda [10].

Concluindo, a aquisição e o tratamento dos dados recebidos, segue o fluxo do diagrama de blocos apresentado na figura 4-6, que mostra todos os passos realizados pelo programa para a aquisição do azimute de *DOA*. Cada bloco foi relatado neste trabalho, desde a etapa de aquisição até a estimação do ângulo azimutal, onde o programa retorna um número inteiro para que a plataforma possa se orientar com relação à fonte do assóvio. Os próximos capítulos discutirão os resultados observados, e extrairão as conclusões finais do projeto.

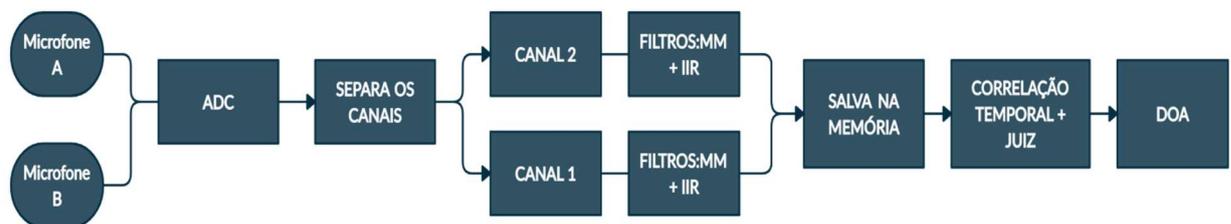


Figura 4-6 – Diagrama de blocos para o sistema de processamento

## 5 RESULTADOS E ANÁLISE

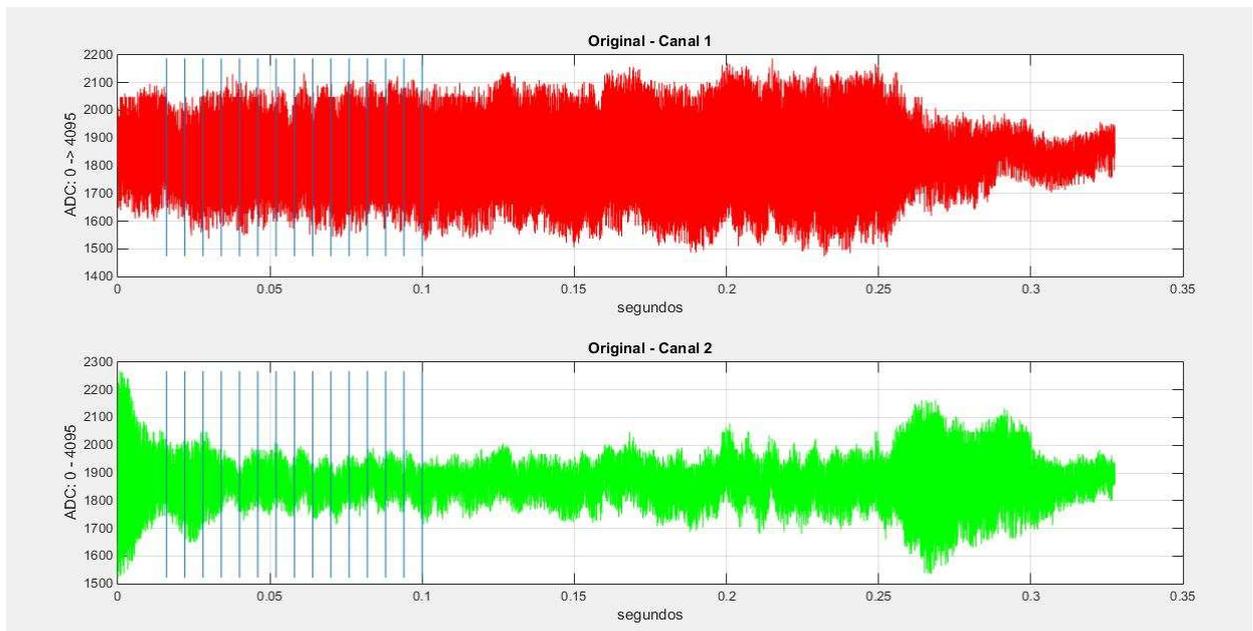
Neste Capítulo são apresentados os resultados experimentais em cada etapa do processo da estimação de direção de chegada de um sinal sonoro utilizando correlação temporal. Os dados coletados pelo MSP430 foram enviados por porta serial para o Arduino que os enviou para um monitor serial no PC. Tais dados foram gravados em um arquivo texto e disponibilizados ao MATLAB para exibição dos sinais e análise das estimativas.

Inicialmente, as estimativas foram feitas com uso de um gerador de frequências, que produzia um som normalizado na faixa de  $2\text{ kHz}$ , em seguida, para  $1\text{ kHz}$  e por fim os testes foram feitos com um assóvio não normalizado. A taxa de amostragem utilizada foi de  $f_{ams} = 50\text{ kHz}$ . Para a obtenção do sinal sonoro mais livre de ruídos externos, os ensaios

foram feitos em um ambiente fechado e por fim, mais testes foram realizados em um estúdio de som, com melhor abafamento acústico para averiguar um possível efeito de eco captado pelos microfones. Para a estimação, a fonte sonora se posicionou em um azimuth aproximado de 0°, 30°, 45° e 60°.

### 5.1 Análise do método no MATLAB

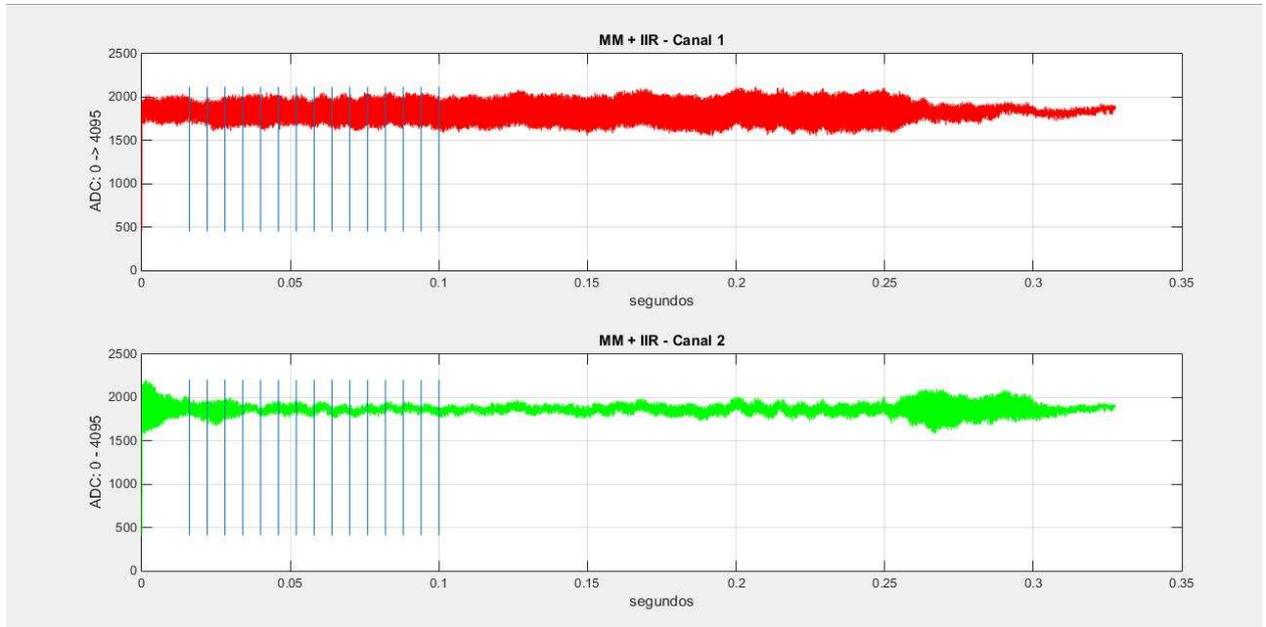
Com base nos dados transmitidos serialmente para o PC, usando o modo de operação 1, descrito na seção 2.2, é possível usar o MATLAB para analisar as operações realizadas pelo código do MSP430. Na verdade, o MATLAB apresenta de forma gráfica o resultado de todas as operações realizadas pelo código. A seguir apresenta-se com algum detalhamento todas as etapas que essa análise permite. A figura 5-1 mostra o resultado a aquisição dos sinais sonoros, ou seja, são os dados entregues pelo ADC, sem qualquer tipo de filtragem. Esses sinais possuem algum ruído, o que pode fazer com que a determinação da faixa de maior energia seja feita de forma errada. Já se nota alguma diferença entre os dois sinais.



**Figura 5-1: Sinal dos microfones de um ensaio sem qualquer filtro.**

Com a filtragem dos sinais, o procedimento para verificar a faixa do assvio torna-se mais eficaz, a figura 5-2 mostra os sinais já filtrados. Esses sinais obtidos são referentes às equações descritas em (4.2) e (4.3). Os sinais parecem bem menores, mas essa ilusão é

porque a escala desta figura está diferente da anterior. Isso aconteceu porque o filtro tem um atraso e por isso a primeira saída é zero.

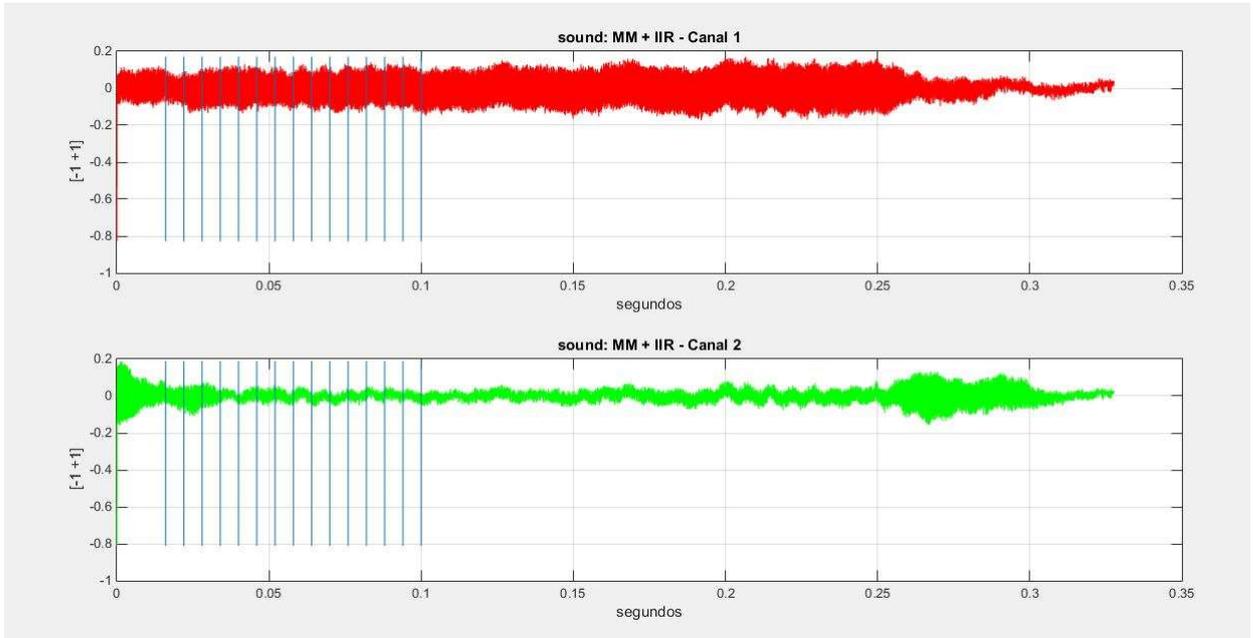


**Figura 5-2: Sinais sonoros filtrados**

As linhas azuis indicadas nas figuras 5-1 e 5-2 representam o início das janelas de tempo que serão usadas para a estimativa dos atrasos. Na versão deste ensaio eram 15 janelas, o que resulta em 15 estimativas de DOA.

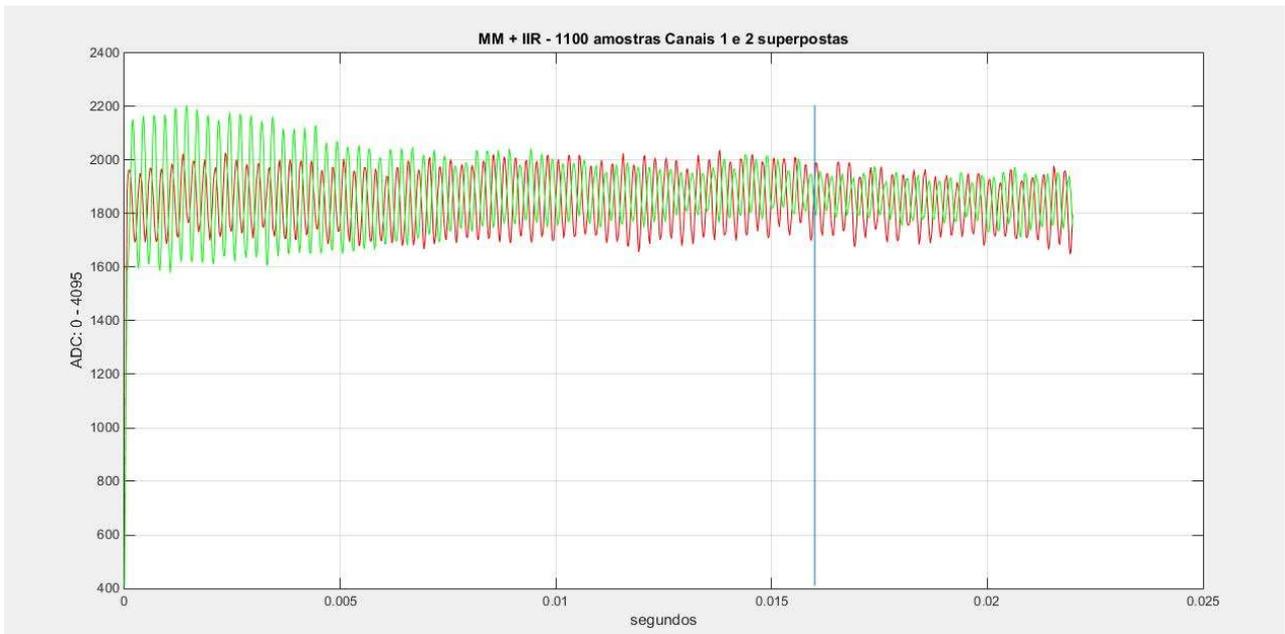
Na próxima etapa, o script MATLAB faz um escalonamento dos dois sinais. Ele programa calcula a média do sinal, a subtrai de todos os dados obtidos. Assim, o sinal resultante tem média 0. Após isso divide os sinais pelo maior valor absoluto de cada um deles. A intenção é a de limitar os sinais na faixa de +/- 1. É de se notar que o valor inicial continua a forçar a escala das figuras.

É importante indicar que tudo isso é feito apenas no MATLAB, para facilitar a visualização dos dados. O código que roda no MSP inicia a estimação dos atrasos logo após os sinais passarem pelos filtros.



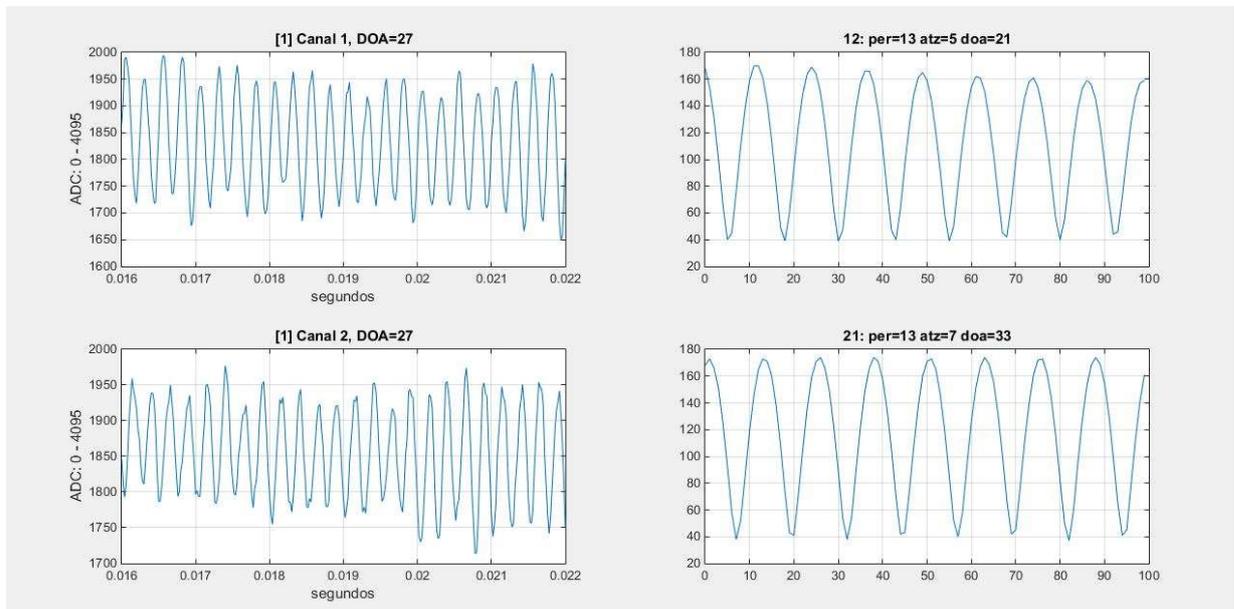
**Figura 5-3: Canais filtrados escalonados em amplitude**

A figura 5-4 apresenta um pequeno trecho da superposição dos sinais dos dois microfones. Não é tão fácil de perceber visualmente, mas o sinal verde está atraso em relação ao vermelho.



**Figura 5-4: Canais filtrados superpostos, mostrando as primeiras 1.100 amostras.**

Em seguida o MATLAB apresenta as 15 diferentes janelas temporais usadas nas 15 estimativas de DOA. Para cada janela, o script apresenta a janela com os dois sinais e e faz dois gráficos do erro médio quadrático, o primeiro é o canal 1 menos o canal 2, o segundo gráfico considera o canal 2 menos o canal 1. Dentre os dois gráficos, escolhe-se o que possui menor atraso, pois além de determinar o valor de  $k$ , ainda permite saber se o sinal vem do lado direito, ou esquerdo. Sendo assim, o resultado da primeira estimativa desse exemplo foi de  $21^\circ$ . Esses resultados são mostrados na figura 5-5. Todas as 15 janelas de estimação são assim apresentadas.



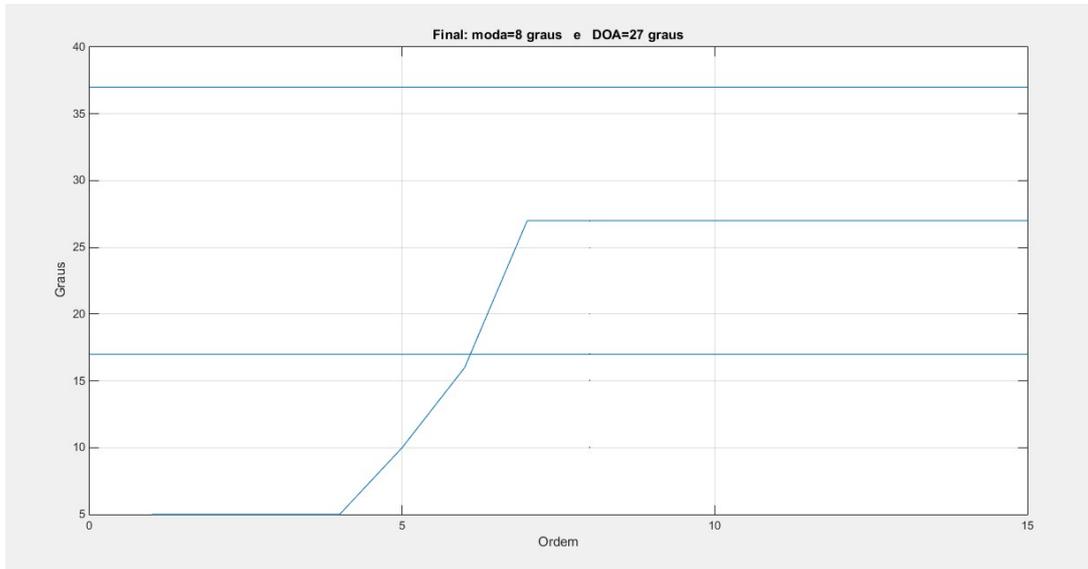
**Figura 5-5: Defasagem entre os sinais dos dois microfones em uma aquisição.**

Por fim, o script MATLAB compara as quinze estimativas com o Juiz. As quinze estimativas realizadas neste ensaio são ordenadas, ficando da forma mostrada abaixo. Algumas, evidentemente estão erradas. A figura 5-6 apresenta de forma gráfica essas 15 estimativas.

**+5 +5 +5 +5 +10 +16 +27 +27 +27 +27 +27 +27 +27 +27 +27**

Nesta figura, temos no eixo horizontal a ordem das amostras e na vertical seus valores em graus. As duas linhas horizontais representam os limiares superiores e inferiores. Neste caso, o limiar foi considerado aceitável. A função juiz tem a finalidade de descartar medidas

extremadas. Nesta versão usa-se como DOA a moda das estimativas que nesse exemplo, foi 27°.



**Figura 5-6: Gráfico após o ordenamento das estimativas.**

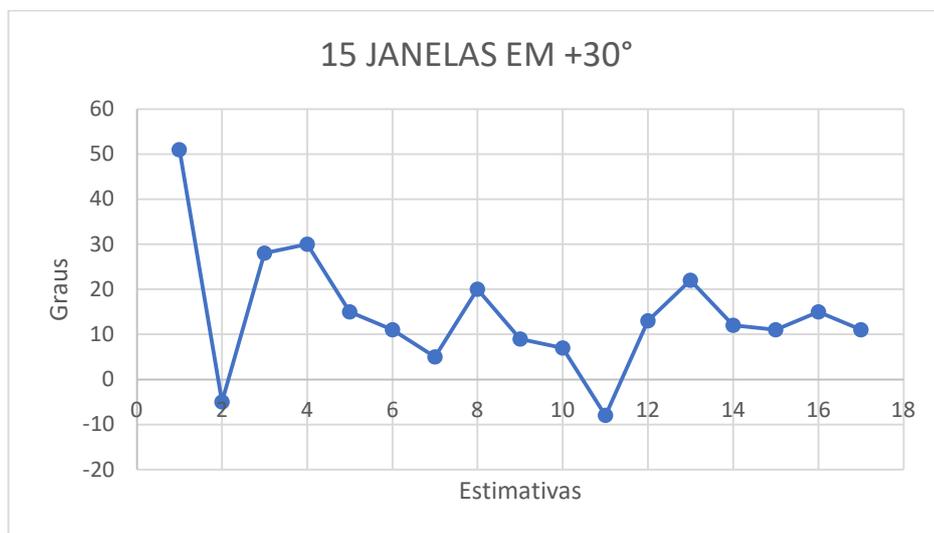
## 5.2 Resultados Iniciais

Nesta seção abordar-se-á a precisão do estimador, bem como a influência dos demais fatores que compõem o sistema. Para um resultado ser considerado acertado o ângulo estimado deve estar a  $\pm 5^\circ$  do azimute da fonte sonora. Um resultado será considerado errado quando o valor estimado estiver a  $\pm 10^\circ$  do azimute da fonte sonora. Serão considerados como erros graves aqueles em que a estimativa tiver ângulo oposto ao do azimute, por exemplo: azimute em  $+30^\circ$  e estimativa em  $-10^\circ$ . Quando o DOA retorna  $\pm 99^\circ$  (na figura aparece como a linha chegando em -100) significa que o programa não foi capaz de realizar a estimativas. Em todos os testes a serem apresentados a fonte sonora foi posicionada em uma distância de aproximadamente 1,20 metro dos microfones. As especificações dos demais parâmetros alterados durante os testes estarão presentes em cada gráfico apresentado.

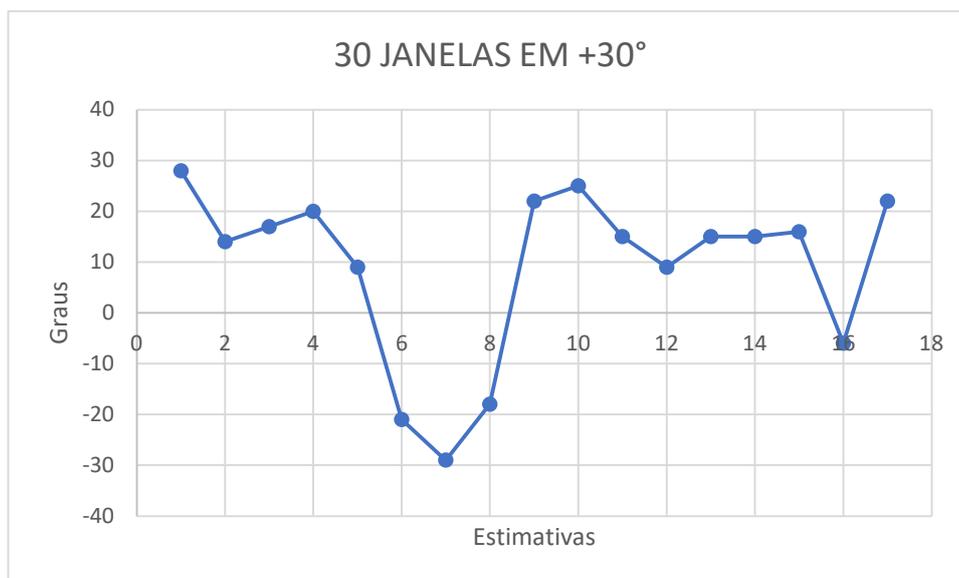
### 5.2.1 Testes em Ambiente sem isolamento acústico

Os dados que serão apresentados a seguir foram realizados tomando  $d = 15\text{ cm}$  e  $f = 2\text{ kHz}$ . Inicialmente alterou-se a quantidade de janelas de tempo selecionadas pelo programa para realizar estimativa e averiguar se isso afetaria a precisão. Os testes foram

feitos com 15 (quinze) e 30 (trinta) janelas de tempo para as estimativas. A fonte sonora foi posicionada em  $\pm 30^\circ$ , as figuras a seguir mostram o resultado.



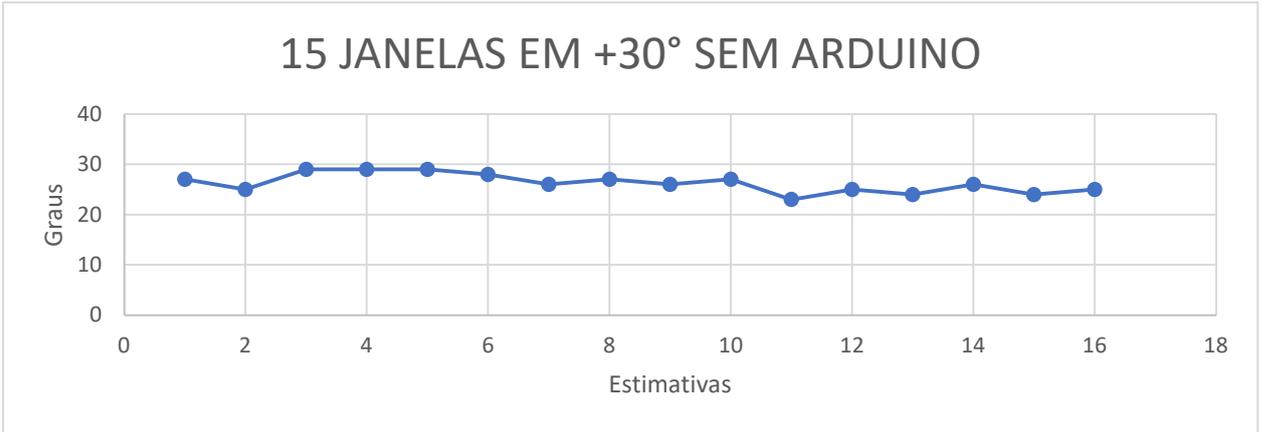
**Figura 5-7: Gráfico das estimativas em  $30^\circ$  com 15 janelas**



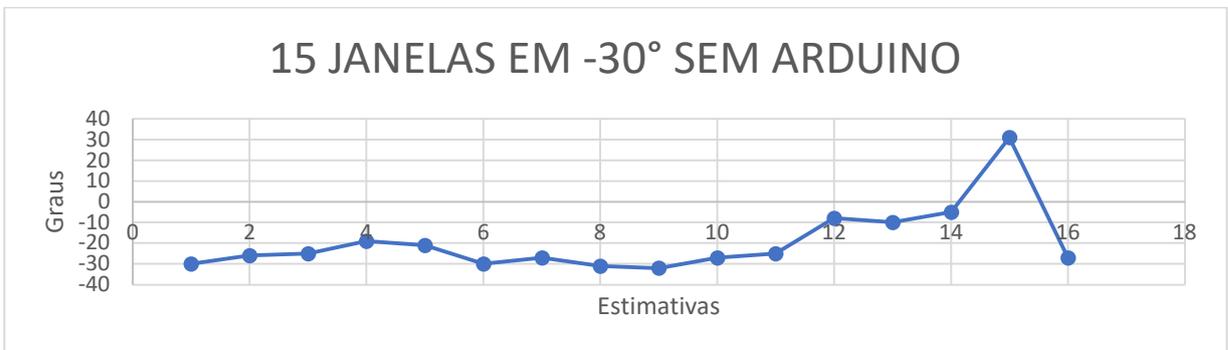
**Figura 5-8: Gráfico das estimativas em  $30^\circ$  com 30 janelas**

Considerando a faixa de erro descrita na seção 5.2, tem-se que apenas 2 das 17 estimativas foram corretas em ambos os casos. Ou seja, uma taxa de acerto de aproximadamente 11,76%.

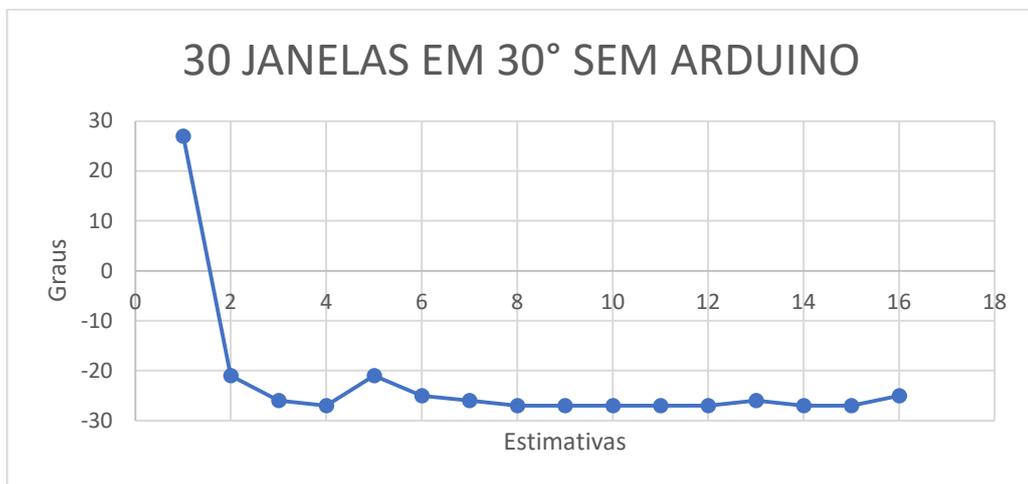
Em seguida, os testes foram refeitos. Porém, desta vez, a conexão do MSP 430 com o Arduino foi desfeita. Os resultados das estimativas eram observados em um painel LCD ligado ao MSP 430. Os resultados obtidos foram:



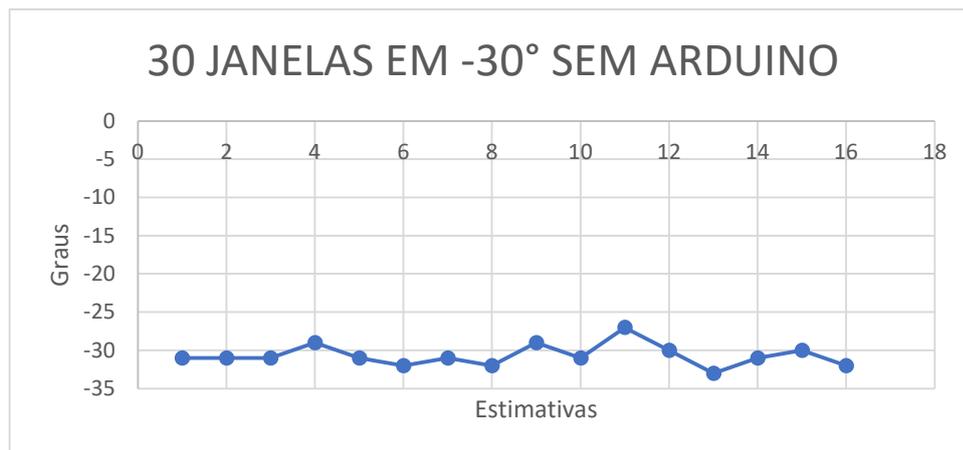
**Figura 5-9: Gráfico das estimativas em 30° com 15 janelas, sem Arduino**



**Figura 5-10: Gráfico das estimativas em -30° com 15 janelas, sem Arduino**



**Figura 5-11: Gráfico das estimativas em 30° com 30 janelas, sem Arduino**

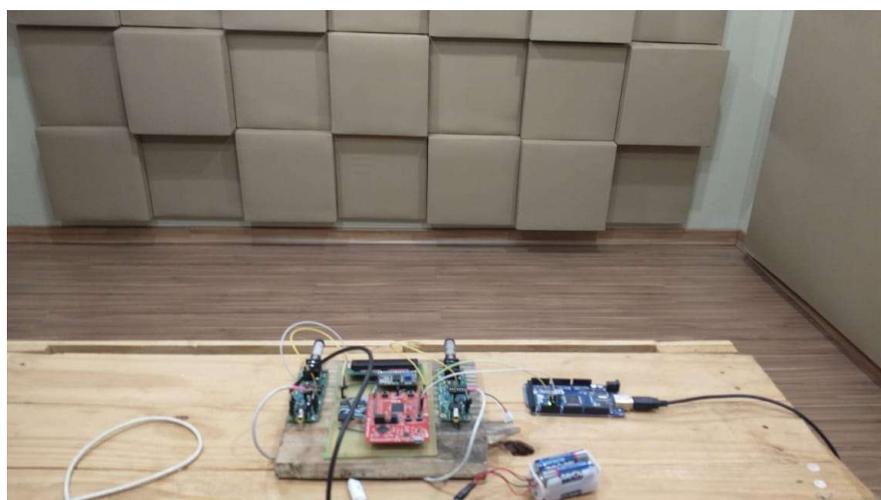


**Figura 5-12: Gráfico das estimativas em 30° com 30 janelas, sem Arduino**

Ao compararmos a figura 5-7 com a 5-9, nota-se que uma vez desfeita a ligação do MSP 430 com o Arduino, os dados aparentemente passam a ser um pouco mais precisos. Contudo, mais testes são necessários para confirmar isso. Pelas figuras 5-9 e 5-10 tem-se uma taxa de acerto de 93,33% em +30° e 73,33% em -30°. Para 30 janelas (figuras 5-11 e 5-12), nota-se que tanto em +30° quanto em -30° o DOA resultava em um ângulo negativo. Com base nos dados coletados, é difícil dizer se o número de janelas para as estimativas contribuiu ativamente com os erros, ou não, mais testes são necessários.

### 5.2.2 Testes em Estúdio com isolamento acústico

Os testes a seguir foram realizados em um estúdio musical com intuito de averiguar possíveis efeitos como diminuição do ruído externo e a possível ressonância da fonte captada pelo sistema.



**Figura 5-13: Montagem do sistema em um estúdio musical.**

Os testes foram feitos com  $d = 15\text{cm}$ ; 15 janelas; frequência da fonte em  $1\text{ kHz}$  e  $2\text{ kHz}$ . Inicialmente os testes foram feitos com o Arduino conectado, mais à frente a conexão será desfeita para averiguar se há, de fato, influência nos resultados.

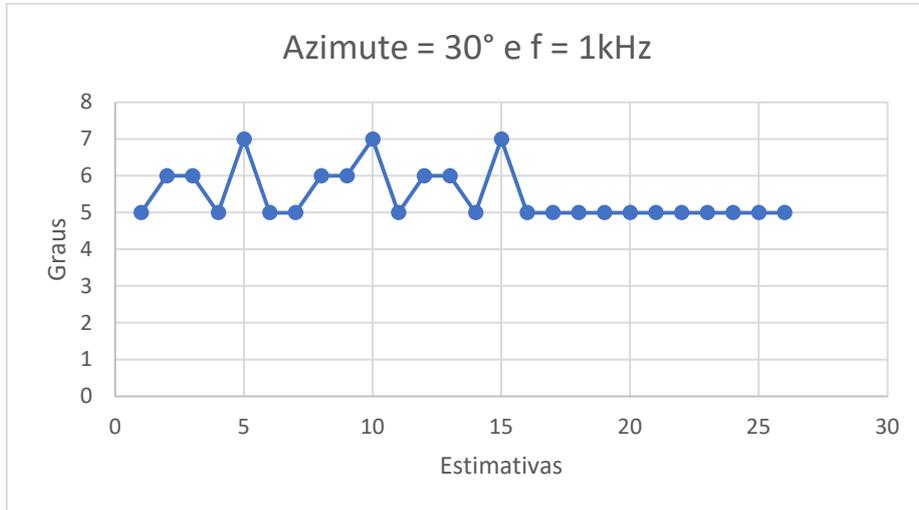


Figura 5-14: Gráfico das estimativas em 30° no estúdio.

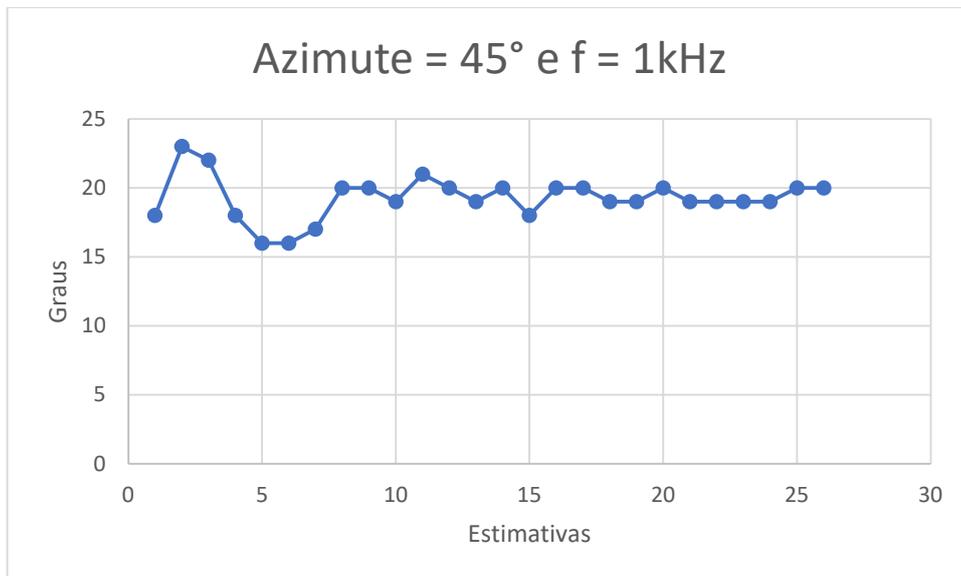
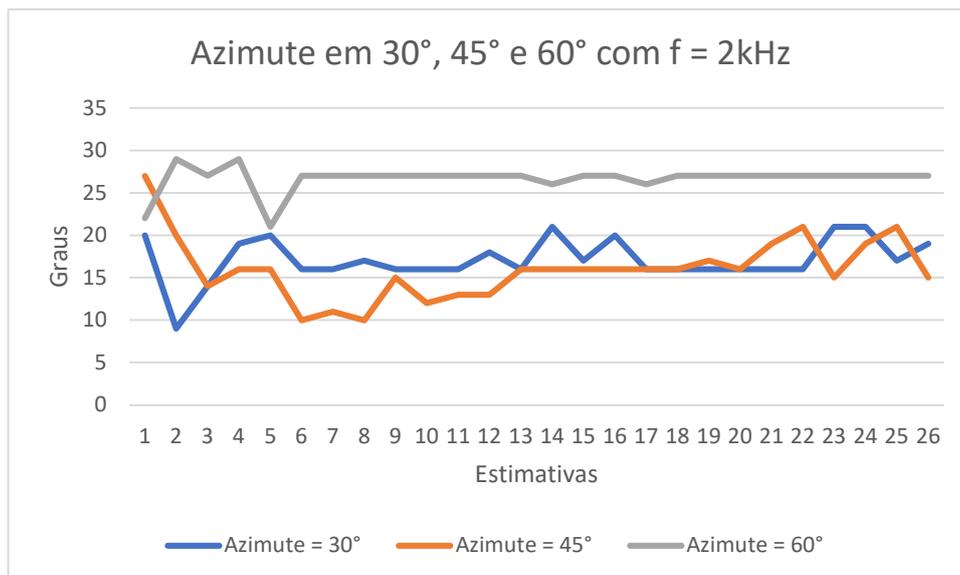


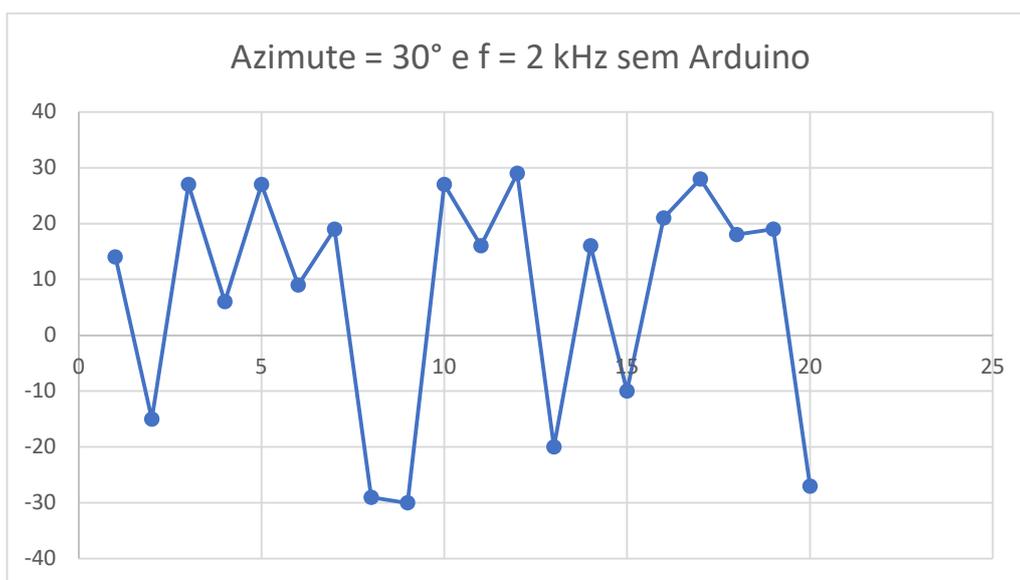
Figura 5-15 Gráfico das estimativas em 45° no estúdio.



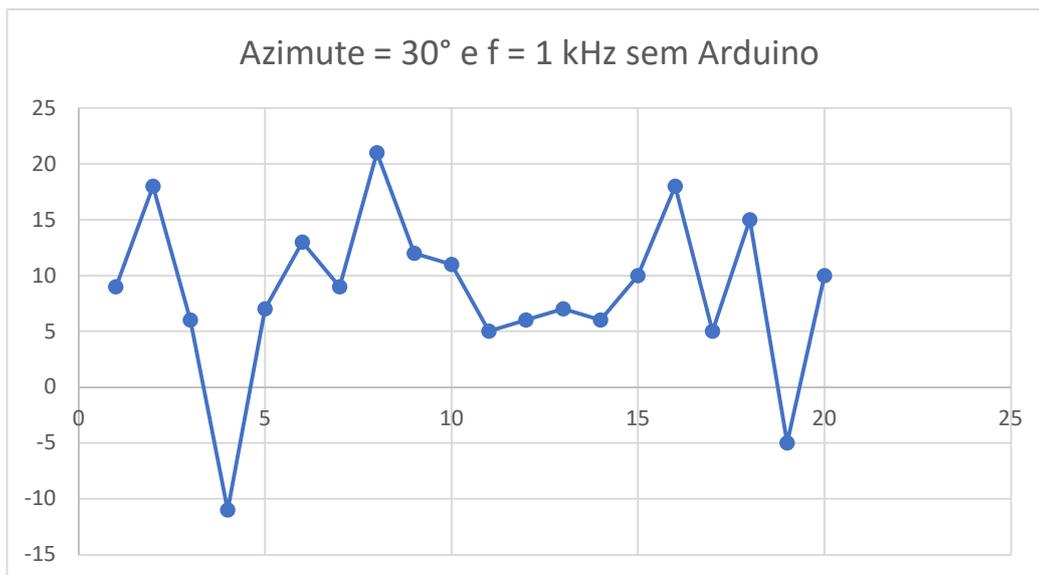
**Figura 5-16: Gráfico das estimativas em para azimutes variado em 2 kHz no estúdio.**

Os testes iniciais, tanto para  $f = 1\text{ kHz}$ , quanto para  $2\text{ kHz}$  foram ruins. Contudo, nota-se que as estimativas feitas em  $30^\circ$  e  $45^\circ$  no ambiente não isolado acusticamente retornavam ângulos negativos, ao passo que no ambiente isolado acusticamente, todas as estimativas foram positivas.

Fez-se a medição no estúdio para as frequências de  $1\text{ kHz}$  e  $2\text{ kHz}$ , com o azimute em  $30^\circ$ , com o Arduino desconectado. Os resultados são mostrados a seguir.



**Figura 5-17: Gráfico das estimativas em  $30^\circ$ , 2 kHz, sem Arduino no estúdio.**



**Figura 5-18:** Gráfico das estimativas em 30°, 1 kHz, sem Arduino no estúdio.

Curiosamente, na frequência de 2 kHz as estimativas chegaram mais perto do valor correto, contudo teve mais erros graves. Já em 1 kHz, as medições ficaram mais distantes do ângulo correto, contudo houve poucos erros negativos.

Por fim, a conexão com o Arduino foi refeita, então fez-se as estimativas usando um assvio não normalizado como fonte sonora, com azimutes diferentes.



**Figura 5-19:** Gráfico das estimativas em 30° do Assvio no estúdio.

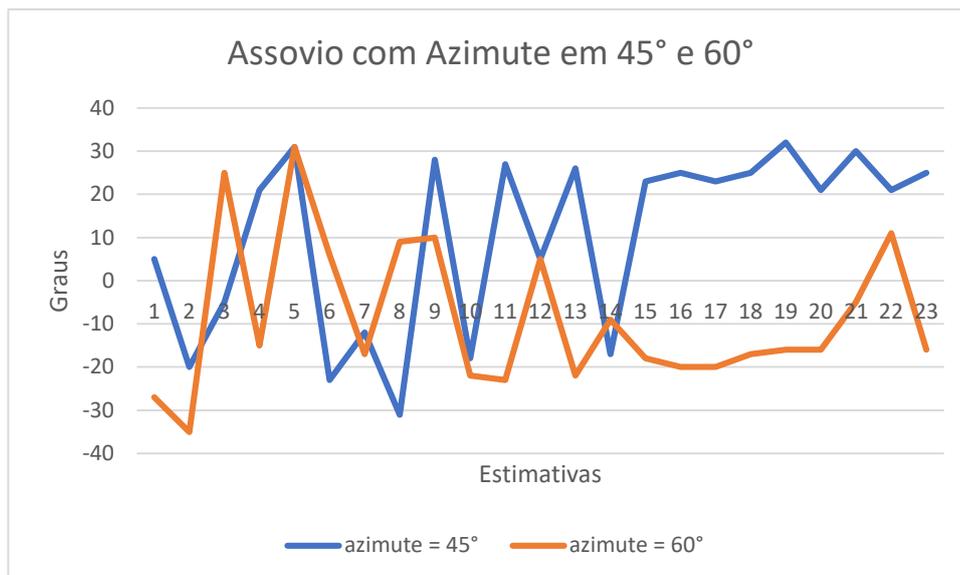


Figura 5-20: Gráfico das estimativas em em 45° e 60° do Assovio no estúdio.

Nota-se que no azimute de 30° teve apenas um erro negativo e algumas poucas medidas chegaram perto do ângulo correto. Entretanto, para as medidas em 45° e 60° houve muitos erros graves.

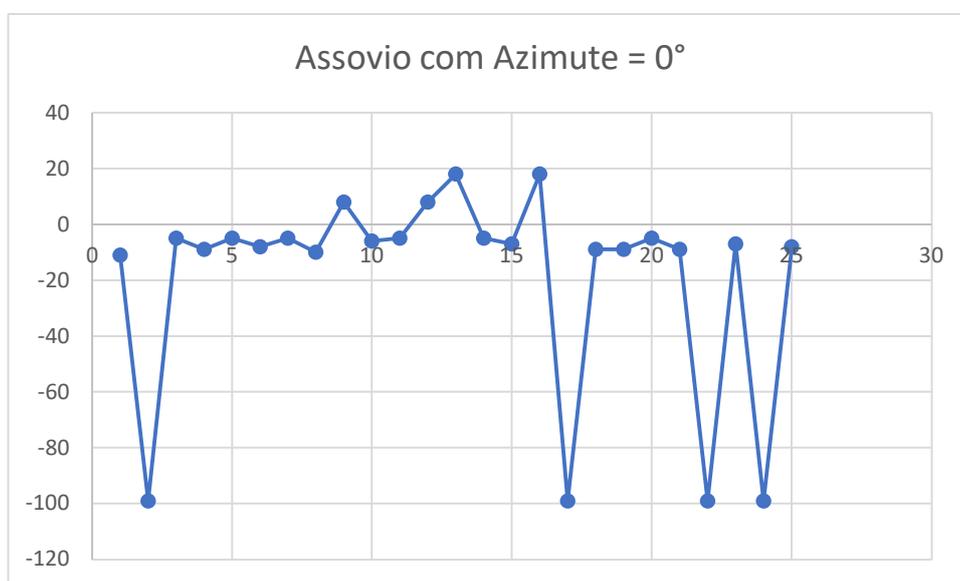


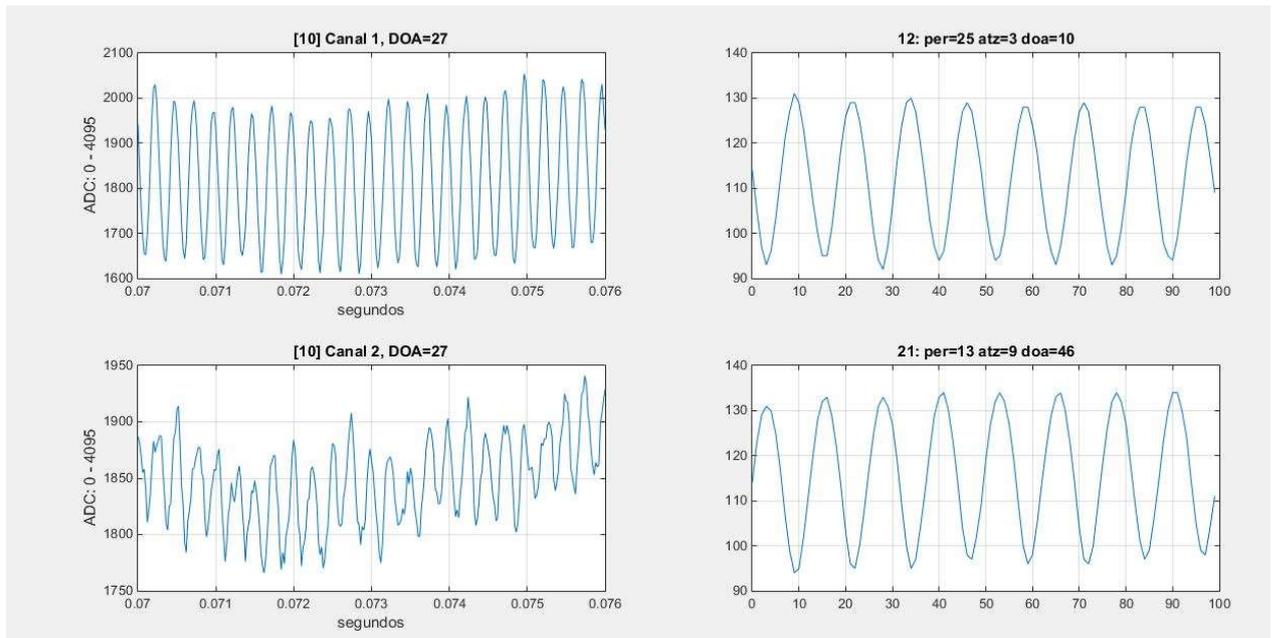
Figura 5-21: Gráfico das estimativas em 0° do Assovio no estúdio.

Com base em bibliografias e nos trabalhos anteriores, sabe-se que 0° costuma ser um ângulo difícil para o sistema estimar, uma vez que em tese as ondas captadas pelos microfones não deveriam sofrer nenhum atraso, resultando em  $k = 0$ . Porém, qualquer distorção nos sinais digitalizados levam o sistema a estimar outro valor. Levando em conta

a resolução atual do sistema, estimativas como  $-5^\circ$ ,  $5^\circ$  podem ser consideradas como boas. É possível ver que não foi possível realizar quatro estimativas na figura 5-21.

### 5.3 Avaliação de resultados

Apesar de diversas hipóteses terem sido levantadas acerca das possíveis razões para a inconstância do estimador, uma prevalece: a má qualidade do sinal no momento da medição. Tomando como exemplo uma das estimativas feitas na seção 5.1, tem-se:



**Figura 5-22: Uma estimativa ruim do exemplo mostrado na seção 5.1.**

A figura 5-22 revela que a janela de tempo do canal 2 usada para a estimativa estava em péssimas condições. Sucede-se que ao calcular o erro quadrático médio ponto-a-ponto para determinar o valor de  $k$  tem-se um erro grande. O primeiro gráfico da diferença resultou em  $10^\circ$ , enquanto o segundo em  $46^\circ$ . Então, para esta janela, a estimativa foi de  $10^\circ$  quando a moda indicava  $27^\circ$ .

Foi possível constatar que é frequente o surgimento de janelas com sinal de baixa qualidade. Acredita-se que esse seja o principal motivo das inconsistências apresentadas pelo sistema e que foram apresentadas nesse trabalho. Logo, faz-se necessário desenvolver um método para descartar tais janelas que trazem sinais ruins para a estimativa. Uma possível solução seria calcular a FFT de cada janela e verificar se elas têm um espectro semelhante. Uma outra forma, talvez fosse possível com o cálculo da variância do sinal em cada janela.

Outros fatores como posição e frequência da fonte sonora, frequência de amostragem, quantidade de estimativas, distância entre os microfones, ecos e reflexões, revelam algum nível de influência nas medições. Por exemplo, as medidas feitas no estúdio, no geral, erraram menos a direção do sinal – se era negativo, ou positivo -, ao passo que em dados momentos as medidas feitas fora de um ambiente acusticamente preparado foram consideravelmente mais acuradas. Contudo, dado a instabilidade do sistema atual, é difícil determinar o real impacto de cada um dos fatores citados.

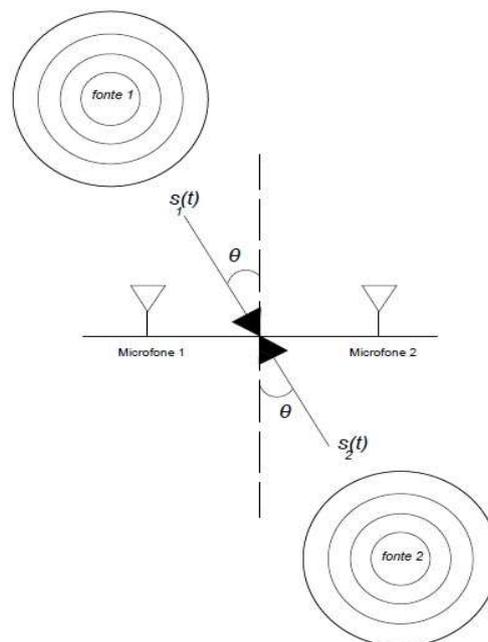
Nas estimativas do azimute com  $\theta = 0^\circ$ , 4 estimativas nem sequer puderam ser feitas. Fato que não ocorreu em nenhum outro momento. Isso mostra a necessidade de aprimoramento do sistema para esse caso.

Uma possível explicação para os erros encontrados nas medidas realizadas com a ligação com o Arduino pode estar na introdução de um ruído extra. O pré-amplificador tem um ganho em torno de 300, por isso ele é muito vulnerável ao ruído. No presente projeto, esses pré-amplificadores são alimentados com bateria. Entretanto, a linha de terra é comum. A porta USB do PC é bastante ruidosa, e esse ruído é injetado através da alimentação na placa com o MSP. Por sua vez, o MSP é um processador que também injeta ruído. A conexão do Arduino introduziu mais um ruído adicional, que não possui correlação com os demais. Há que se investigar, mas esta é uma possibilidade a ser analisada.

Para averiguar essa hipótese, os testes podem ser refeitos com o MSP430 conectado a um *Power Bank*, assim seria possível observar se as medições seriam, de fato, melhores. Medições feitas com o Arduino, sendo alimentado por outro *Power Bank*, conectado e desconectado do MSP430. Dessa forma, será possível determinar se há de fato ruído descorrelacionado entre os microprocessadores e se de fato isso interfere na estimativa DOA. Uma boa alternativa, além do *Power Bank* é usar a transmissão via *Bluetooth*, já que só acontece após a coleta de dados. Outra possibilidade que deveria ser empregada é ativar o modo baixo consumo do MSP430 durante todo o período de aquisição.

## 5.4 Limitações do projeto

Uma das principais limitações do projeto é a de que o arranjo dos microfones não consegue identificar se a fonte sonora está posicionada a frente ou atrás dos mesmos. Um dos fatores disso ocorrer é porque os microfones que são sensores omnidirecionais, que detectam sinais sonoros vindos de todas as direções, dificultado a detecção do sentido do sinal.



**Figura 5-23: Demonstração de como o mesmo ângulo pode valer para duas fontes diferentes [10]**

Manter os microfones alinhados e espaçados na distância correta foi difícil, já que os fios do próprio sistema puxavam a placa dos microfones e como a distância  $d$  é fundamental para a estimativa DOA, foi necessário fixar os circuitos dos microfones em uma placa de madeira. Porém, isso dificultou parte do experimento, uma vez que não era possível alterar a distância  $d$  com facilidade.

## 6 CONCLUSÃO

Este projeto teve por objetivo determinar o principal fator de erro das medições do estimador DOA. Com base nos resultados, é possível afirmar que o algoritmo de estimação de direção de chegada funciona de forma razoável. Entretanto, o método de seleção dos instantes de tempo para a medição deve ser reavaliado. Como visto na seção 5.3 as discrepâncias encontradas durante o processo de estimação de chegada do sinal sonoro revelam que o fator preponderante para os erros é a baixa qualidade encontrada em certas janelas de tempo dos canais. Uma vez que tal problema seja resolvido, será possível observar melhor o impacto dos demais parâmetros na precisão do sistema.

Aumentar a taxa de amostragem para  $50\text{ kHz}$  para cada microfone é uma alternativa viável para aumentar a resolução do sistema e garantir medidas mais próximas do azimute real da fonte sonora.

Sugere-se para trabalhos futuros que o método de verificação da qualidade das faixas dos sinais seja modificado. Por exemplo, é possível aplicar a Transformada Rápida de Fourier (FFT) no sistema e comparar a distribuição de energia entre as duas janelas. Outro projeto possível consiste em construir um sistema de medições contínuas enquanto a fonte sonora se movimenta de forma linear, ou seja, o sistema realiza várias DOAs e compara os azimutes finais para verificar se de fato os ângulos seguem uma ordem contínua.

Esse projeto pode ser incorporado a outros projetos que utilizem uma plataforma móvel, como por exemplo, o de detecção de obstáculos, podendo realizar as duas funções, melhorando as ações da plataforma. O que pode levar a sistemas com aplicações práticas muito importantes, como por exemplo: um robô que procura por pedidos de socorro advindo de sobreviventes de desmoronamentos.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] ARAÚJO, ANDRÉ MEDEIROS (2018). Plataforma móvel guiada pelo som. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2018, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.
- [2] A. T. Gontijo, Estimação da Direção de Chegada em Tempo Real por Arranjo de Microfones. Tese de Mestrado, UnB, Brasília, ago. 2010.
- [3] Chapman, S. (2005). *Electric Machinery Fundamentals* (4<sup>a</sup> ed.). New York: McGraw-Hill.
- [4] Datasheet MSP430-F5529. Disponível em <https://www.ti.com/product/MSP430F5529>
- [5] Datasheet Microchip 23LC1024. Disponível em <https://www.microchip.com/wwwproducts/en/23LC1024>.
- [6] LIMA, LUCAS; SOARES, MATHEUS (2017). Plataforma móvel com detecção de obstáculos. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2017, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.
- [7] NXP *Semiconductors*. (2015). PCA9865. Disponível em [http://www.nxp.com/documents/data\\_sheet/PCA9685.pdf](http://www.nxp.com/documents/data_sheet/PCA9685.pdf).
- [8] OLIVEIRA, IGOR; OLIVEIRA, RENATO (2016). Plataforma móvel com detecção de obstáculos. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2016, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.
- [9] REIS, YURI SOUZA (2019). Sistema de detecção de ondas sonoras. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2019, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.
- [10] SCOLA, CARLOS FERNÁNDEZ, ORTEGA, MARÍA, D, B, “*Direction of Arrival estimation – A two microphones approach*”. *Master Thesis, Blekinge Institute of Technology*, set. 2010.

- [11] Sen, P. C. (1997). *Principles of Electric Machines and Power Electronics* (2ª ed.). Kingston: John Wiley & Sons
- [12] TORRES, GUILHERME DE SOUSA (2018). Estimação de Direção de Chegada de Sinais Sonoros Utilizando Arranjo de Sensores. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2018, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.
- [13] Toshiba. (s.d.). TB6612FNG. Disponível em [https://cdnshop.adafruit.com/datasheets/TB6612FNG\\_datasheet\\_en\\_20121101.pdf](https://cdnshop.adafruit.com/datasheets/TB6612FNG_datasheet_en_20121101.pdf).