



**ESTIMADOR DE DIREÇÃO DE CHEGADA
DE SINAIS SONOROS**

BRENO DE ALMEIDA MENEZES

**TRABALHO DE CONCLUSÃO DE CURSO EM ENGENHARIA ELÉTRICA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

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

**ESTIMADOR DE DIREÇÃO DE CHEGADA
DE SINAIS SONOROS**

BRENO DE ALMEIDA MENEZES

ORIENTADOR: PROF. RICARDO ZELENOVSKY, ENE/UNB

TRABALHO DE CONCLUSÃO DE CURSO EM ENGENHARIA ELÉTRICA

BRASÍLIA-DF, 01 DE JUNHO DE 2021.

ORIENTADOR: RICARDO ZELENOVSKY

FICHA CATALOGRÁFICA

MENEZES, BRENO DE ALMEIDA

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. Estimção DOA

I. ENE/FT/UnB II.

REFERÊNCIA BIBLIOGRÁFICA

MENEZES, BRENO DE ALMEIDA(2021). Plataforma estimadora de sinais sonoros. Trabalho de Conclusão de Curso em Engenharia Elétrica, 2021, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF

RESUMO

Neste trabalho foi incorporado um microcontrolador MSP430-F5529 a um sistema de captação de sinais e implementado um sistema para avaliação de um estimador de direção de chegada de um sinal sonoro em um ambiente fechado. Iniciamos as atividades trabalhando com o sistema completo, utilizando o código previamente feito em trabalhos anteriores e remodelando para que a comunicação entre microcontrolador e computador seja feita através da comunicação UART. O sistema de captação dos sinais é composto por dois microfones que captam um sinal sonoro em um período previamente estipulado e através do cálculo da correlação temporal, erro quadrático médio e o atraso entre as amostras é possível estimar o azimute da fonte sonora. Este trabalho prossegue com o estudo para entender e mitigar os erros de estimação identificados pelo sistema.

ABSTRACT

In this work, a microcontroller MSP430-F5529 was incorporated, a signal capture system and implemented a system for evaluating an estimator of the direction of arrival of a sound signal in a closed environment. We started the activities working with the complete system, using the code previously made in previous works and remodeling so that the communication between microcontroller and computer is done through UART communication. The signal capture system is composed of two microphones, previously built by students and teachers, which capture a sound signal in a previously stipulated period and by calculating the temporal correlation, mean square error and the delay between them, it is possible to estimate the azimuth of the sound source. This work continues with the study to understand and mitigate the evaluation errors identified by the system.

LISTA DE QUADROS

Quadro 4.1 - Modos do programa	29
Quadro 4.2: Valores reais de ângulos do projeto	33
Quadro 4.3: Valores de ângulos usados no programa	34
Quadro 5.1: Primeiros 16 números do sinal quadrado da memória 0 e 1.....	52
Quadro 5.2: Primeiros 16 números do sinal filtrado da memória 0 e 1.....	53
Quadro 5.3: Ângulos resultantes para teste da estimativa A1 adiantado de A0.....	58
Quadro 5.4: Ângulos resultantes para teste da estimativa A0 adiantado de A1	60
Quadro 6.1: Primeira análise	69
Quadro 6.2: Segunda análise	69

LISTAS DE FIGURAS

Figura 1.1: Hardware do estimador de direção de chegada.....	13
Figura 2.1: Etapas da captação/conversão do sinal	15
Figura 2.2: Arranjo entre os microfones.....	16
Figura 2.3: Frente de onda bidimensional incidente nos microfones	17
Figura 2.4: Situação em que o atraso é detectado e é adequado.....	19
Figura 2.5: Situação em que o atraso detectado não é adequado	20
Figura 2.6: Região com/sem a presença do Aliasing Espacial	20
Figura 3.1: MSP430F5529 com suas características	21
Figura 3.2: MSP430F5529	22
Figura 3.3: Microfone Omnidimensional e seu pré-amplificador	22
Figura 3.4: Curva de resposta em frequência do microfone.....	23
Figura 3.5: Circuito do estágio de pré-amplificação	23
Figura 3.6: Esquema de conexão entre baterias, pré-amplificador e MSP430 apenas com um microfone.....	24
Figura 3.7: Esquema enfatizando a alimentação dos pré-amplificadores e suas conexões com o MSP430	25
Figura 3.8: CHIP 23LC1024 - Memória externa.....	26
Figura 3.9: Conexões entre as duas memórias externas e o MSP430	26
Figura 3.10: Módulo Bluetooth HC-05	27
Figura 3.11: Conexão física do HC-05 ao MSP430F5529.....	28
Figura 4.1: Etapas da transferência dos dados da memória externa para o computador.....	30
Figura 4.2: Ângulos possíveis para estimativa com dois microfones	31
Figura 4.3: Diagrama de bloco do sistema	35
Figura 4.4: Representação de um deslocamento para se calcular a diferença entre os dois sinais.	37
Figura 4.5: Senoides sintetizadas pelo MATLAB.....	38
Figura 4.6: Gráfico da correlação	38
Figura 5.1: Decomposição de um sinal PWM em uma componente DC e uma onda retangular de média zero.....	41

Figura 5.2: Filtro interligado ao MSP430F5529	42
Figura 5.3: Canais A0 e A1 do sinal triangular	43
Figura 5.4: Gráficos gerados no MATLAB, frutos dos dados da conversão analógico-digital	43
Figura 5.5: FFT dos canais A0 e A1	44
Figura 5.6: Tensão no pino P6.1 com referência à terra do MSP430.....	45
Figura 5.7: Sinal entregue pelo ADC quando os microfones estão em silêncio	45
Figura 5.8: Código usado no MATLAB para testar o sinal gerado pelo aplicativo do celular	46
Figura 5.9: Resultado da análise do sinal gerado pelo aplicativo do celular. Se vê a forma senoidal é confiável e, na direita, que a FFT indica o valor correto de frequência.	47
Figura 5.10: Resultado da conversão ADC, plotado pelo Matlab	47
Figura 5.11: FFT do sinal digitalizado que foi entregue ao Matlab	48
Figura 5.12: Circuito equivalente de entrada analógica	49
Figura 5.13: Sinal puro recebido entregue pelo ADC com o emprego de um divisor resistivo com resistores de 1 k Ω	51
Figura 5.14: Sinal quadrado recebido e plotado pelo MATLAB	52
Figura 5.15: Sinal A0 e A1 filtrado pelo média móvel	54
Figura 5.16: Faixa adequada para realizar a estimativa.....	55
Figura 5.17: Referência utilizada para os ângulos, definindo então lados direito e esquerdo..	56
Figura 5.18: Defasagem de 10 posições para A1	57
Figura 5.19: Defasagem de 9 posições para A1	57
Figura 5.20: Defasagem de 8 posições para A1	57
Figura 5.21: Defasagem de 7 posições para A1	57
Figura 5.22: Defasagem de 6 posições para A1	57
Figura 5.23: Defasagem de 5 posições para A1	57
Figura 5.24: Defasagem de 4 posições para A1	58
Figura 5.25: Defasagem de 3 posições para A1	58
Figura 5.26: Defasagem de 2 posições para A1	58
Figura 5.27: Defasagem de 1 posições para A1	58
Figura 5.28: Defasagem de 10 posições para A0	59
Figura 5.29: Defasagem de 9 posições para A0	59
Figura 5.30: Defasagem de 8 posições para A0	59
Figura 5.31: Defasagem de 7 posições para A0	59
Figura 5.32: Defasagem de 6 posições para A0	59

Figura 5.33: Defasagem de 5 posições para A0	59
Figura 5.34: Defasagem de 4 posições para A0	60
Figura 5.35: Defasagem de 3 posições para A0	60
Figura 5.36: Defasagem de 2 posições para A0	60
Figura 5.37: Defasagem de 1 posições para A0	60
Figura 6.1: Hardware do DOA/fonte sonora/transferidor utilizado nos ensaios	62
Figura 6.2: Resultado dos ensaios em 5°, 10° e em 15°	63
Figura 6.3: Resultado dos ensaios em 20°, 25° e em 30°	64
Figura 6.4: Resultado dos ensaios em 35°, 40° e em 45°	64
Figura 6.5: Resultado dos ensaios em 50°, 55° e em 60°	65
Figura 6.6: Resultado dos ensaios em -5°, -10° e em -15°	66
Figura 6.7: Resultado dos ensaios em -20°, -25° e em -30°	66
Figura 6.8: Resultado dos ensaios em -35°, -40° e em -45°	67
Figura 6.9: Resultado dos ensaios em -50°, -55° e em -60°	67

LISTAS DE TERMOS E SIGLAS

UART	Universal Asynchronous Receiver/Transmitter
DMA	Direct Memory Access
ADC	Analogic to digital converter
A/D	Analógico para digital
DOA	Direction of arrival
FFT	Fast Fourier Transform
MATLAB	Marca Registrada da MathWorks, Inc
SPI	Serial Peripheral Interface
LCD	Liquid Cristal Display
GPIO	General Purpose Input/Output

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVO	12
1.2	ORGANIZAÇÃO DO TRABALHO	13
1.2.1	Ponto de Partida	14
2	EMBASAMENTO TEÓRICO	15
2.1	CARACTERIZAÇÃO DO SINAL SONORO E SUA CAPTAÇÃO/CONVERSÃO 15	
2.2	ARRANJO DOS MICROFONES	16
2.3	DISTÂNCIA ENTRE OS MICROFONES	18
2.4	ALIASING	19
3	HARDWARE	21
3.1	MSP430F5529	21
3.2	MICROFONE E CIRCUITO DE PRÉ-AMPLIFICAÇÃO	22
3.3	MEMÓRIA EXTERNA	25
3.4	BLUETOOTH	27
3.4.1	Configuração do HC-05	28
4	SOFTWARE PARA ESTIMATIVA DOA	29
4.1	FIRMWARE	29
4.2	MÉTODO DE ESTIMAÇÃO DA DOA	31
4.2.1	Faixa de ângulo suportado	33
4.2.2	Etapas da estimativa	34
4.2.2.1	Primeira etapa: filtro média móvel	35
4.2.2.2	Segunda etapa: identificação da faixa de maior energia	36
4.2.2.3	Terceira etapa: Estimativa	36
4.3	APERFEIÇOAMENTOS INICIAIS DO SISTEMA	39
5	TESTES PRÁTICOS	41
5.1	TESTE DO CONVERSOR ANALÓGICO-DIGITAL (ADC)	41
5.1.1	Teste usando um sinal criado pelo próprio MSP430	41

5.1.2	Teste usando sinais recebidos pelos microfones	44
5.2	TESTE DA CAPTAÇÃO DOS MICROFONES	45
5.2.1	Modificação do divisor resistivo.....	48
5.3	TESTE DO ENVIO DOS DADOS PELO HC-05	51
5.4	TESTE DO FILTRO MÉDIA MÓVEL	52
5.5	TESTE IDENTIFICAÇÃO DE UMA DA REGIÃO ADEQUADA PARA ESTIMATIVA DO DOA	54
5.6	TESTE DA ESTIMATIVA DO ATRASO	55
6	ENSAIOS DA DIREÇÃO DE CHEGADA DO SINAL SONORO	62
6.1	ENSAIOS PARA ÂNGULOS DE REFERÊNCIA POSITIVA	63
6.2	ENSAIOS PARA ÂNGULOS DE REFERÊNCIA NEGATIVA.....	65
6.3	CONCEITOS APRENDIDOS DURANTE OS ENSAIOS	68
6.4	AVALIAÇÃO DOS RESULTADOS OBTIDOS	68
7	CONCLUSÕES E PROPOSTAS PARA FUTURAS ANÁLISES.....	71
	REFERÊNCIA BIBLIOGRÁFICAS.....	73

1 INTRODUÇÃO

O desenvolvimento de sistemas de estimação de direção de chegada (DOA) de sinais faz parte dos desejos da sociedade há anos. Apesar de ainda ser tratada como ficção científica na maioria dos casos, já está presente em diversas aplicações do cotidiano e continua sendo alvo de investimentos tecnológicos. Podemos destacar como exemplo sistemas de redes de antenas que usam essa técnica para estimar a direção de chegada de sinais incidentes.

Estes sistemas de redes inteligentes foram desenvolvidos para fins tecnológicos nos sistemas de comunicação móvel de diversas gerações. Com o passar do tempo ocorreu a inevitável difusão da tecnologia para todo meio social, assim diversas outras aplicações surgiram como o auxílio em resgates no caso de acidente e durante competições esportivas. Além disso, as pesquisas continuam apontando para novos usos da tecnologia. O artigo¹, por exemplo, desenvolveu no contexto dos problemas de localização de atiradores pelos sinais sonoros associados ao disparo de uma arma de fogo.

Tendo em vista tais sistemas, as aplicações em captações de sinais sonoros por robôs se mostram prometedoras. Em que por meio desses robôs, será possível enviá-los para procurar vítimas em lugares que o ser humano não consegue alcançar e sem causar riscos a sua saúde física, ou seja, em meio aos escombros ou perto de vulcões em erupção.

1.1 OBJETIVO

Este trabalho é uma continuação das atividades desenvolvidas em [1] e [2] que posteriormente serão mais discutidas. O presente trabalho de conclusão de curso pretende possibilitar que o estimador de direção de direção de chegada trabalhe de forma que ao receber um sinal sonoro pelos dois microfones, responda com a estimativa do ângulo de direção de chegada. Este trabalho iniciou com o projeto existente, que apresentava erros de estimação muito além do que seria esperado. O firmware disponível estava muito retalhado e muitas vezes conflitante. Por isso, se fez necessário uma verificação completa do hardware, testes para comprovar seu funcionamento e reescrituração de todo firmware.

¹ Fonte: <https://www.bibliotecas.sbrt.org.br/articles/236>

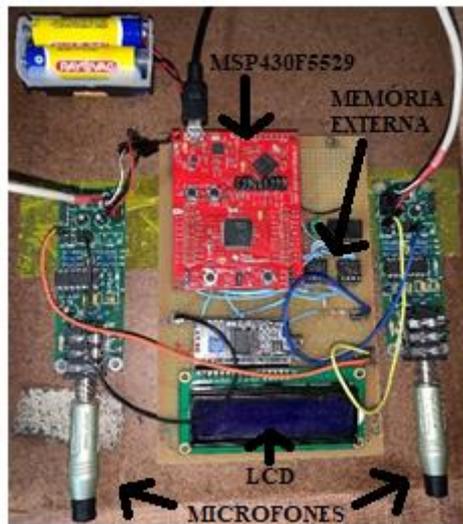


Figura 1.1: Hardware do estimador de direção de chegada

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho foi dividido em três partes.

A primeira parte composta pelos capítulos 2, 3 e 4 trata sobre o embasamento teórico, o hardware e algoritmo utilizado no projeto. Mais especificamente, o capítulo 2 fala sobre o Aliasing, o problema da distância entre os microfones e o equacionamento do arranjo dos microfones. O capítulo 3 aborda o microcontrolador MSP430-F5529, a memória externa que funciona como auxiliar à memória presente no microcontrolador, o microfone com seu circuito pré-amplificador e o Bluetooth HC-05 que foi inserido ao projeto. Por fim, o capítulo 4 que trata sobre o algoritmo utilizado e explicará assuntos como firmware, método de estimação de DOA, etapas de estimativa e resolução do estimador. Como já foi afirmado, este trabalho é uma continuação de projetos previamente desenvolvidos, assim, grande parte do embasamento teórico aqui apresentado é semelhante ao que se encontra nos trabalhos [1] e [2].

A segunda parte composta pelo capítulo 5 e 6 trata sobre os resultados dos ensaios realizados e as simulações feitas com a ajuda do MATLAB para verificar a qualidade do sistema final.

A última parte composta pelo capítulo 7 trata sobre as conclusões obtidas com este projeto e discute também a possibilidade de trabalhos futuros, no que se pode melhorar para tornar o sistema mais eficiente e preciso.

1.2.1 Ponto de Partida

Este trabalho é a continuação de dois trabalhos de conclusão de curso. O primeiro trabalho [2] foi mais um teste preliminar das estimativas que o sistema calculava de acordo com o sinal recebido. O segundo trabalho [1] foi uma investigação das causas que poderiam afetar o sistema para não estimar de forma correta, chegando à conclusão de que o meio que era utilizado para troca de informação com o computador, era o possível causador de distúrbios no sistema.

O meu trabalho consiste em continuar a tarefa de entender o que se passa no sistema, passando por todas as funções e remodelando-as , além de, verificar os componentes físicos de todo o sistema e ter a conclusão se algo está prejudicando a estimativa. Mas, primeiramente, foi feito dois aperfeiçoamentos iniciais para aumentar a velocidade do processo de estimativa que tomavam um tempo que não era utilizado de fato para as estimativas.

2 EMBASAMENTO TEÓRICO

2.1 CARACTERIZAÇÃO DO SINAL SONORO E SUA CAPTAÇÃO/CONVERSÃO

O sinal capturado na entrada do sistema tem características de uma onda mecânica longitudinal e periódica que se propaga por vibração no ar. Desse modo, a fim de, simplificar para descobrir onde há falhas no sistema, a fonte sonora será gerada por um aplicativo no celular que emite uma onda senoidal à escolha do usuário, o meio de transmissão é o ar e o receptor são dois microfones.

Essa onda sonora quando chega aos microfones é convertida em um sinal elétrico analógico dentro da faixa de 0 até 6 volts (isto será melhor explicado mais adiante). Para o armazenamento e processamento do sinal elétrico dos microfones é necessário mais uma conversão denominada de conversão analógico-digital, na faixa de 0 até 3,3 volts, feita pelo microcontrolador que então gera um sinal quantificado em tempo discreto. O armazenamento desses dados convertidos é feito numa unidade externa ao microcontrolador na forma de vetores, e somente a partir desses dados em vetores que é iniciada a filtragem e cálculos numéricos para estimar a direção de chegada.

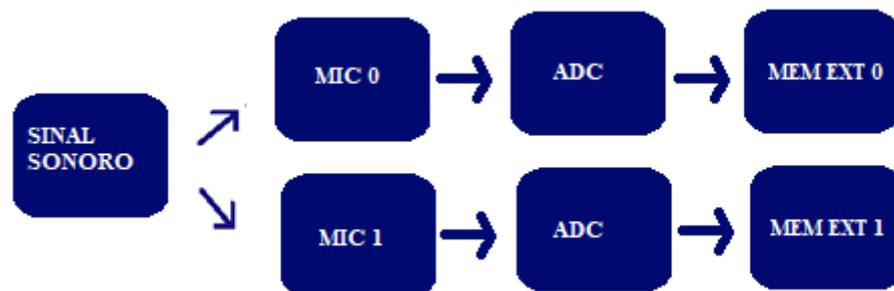


Figura 2.1: Etapas da captação/conversão do sinal

As duas memórias externas são do tipo SRAM e cada uma tem o tamanho de 128 KB, o que permite o armazenamento de até 64K valores inteiros em cada, totalizando 128KB valores inteiros nas memórias externas.

2.2 ARRANJO DOS MICROFONES

Para captar o sinal sonoro e estimar sua direção é utilizado no mínimo dois sensores sonoros organizados de forma adequada pelo fato do método que será usado nesse trabalho depender do atraso do som captado em relação aos microfones entre si.

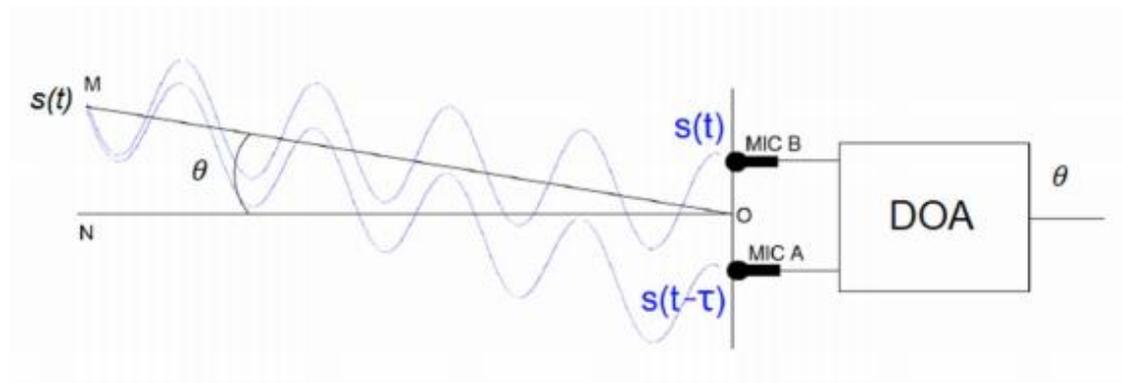


Figura 2.2: Arranjo entre os microfones

Este trabalho define que a direção de chegada é dada pelo ângulo ‘ θ ’. Na Figura 2.2 a direção de chegada do sinal é definida pelo ângulo formado pela reta ‘N’ que é perpendicular à linha que une os dois microfones e o caminho formado entre o centro dos microfones até a fonte sonora representado por ‘ $s(t)$ ’.

Olhando a captação dos sinais sonoros recebidos pelos microfones de uma forma bidimensional, o mesmo som é captado em tempos distintos, ou seja, de forma defasada, e de forma totalmente dependente do ângulo em que está posicionada a fonte sonora. A distância da defasagem ‘D’, mostrado na equação 2.1, do sinal sonoro ‘ $s(t)$ ’ do microfone ‘B’ em relação ao microfone “A” é a distância entre os microfones “d” vezes o seno do ângulo “ θ ” de incidência. A Figura 2.3 demonstra essas análises.

$$D = d * \text{sen}(\theta) \quad (2.1)$$

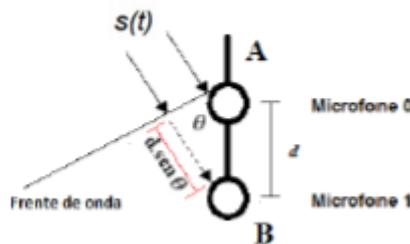


Figura 2.3: Frente de onda bidimensional incidente nos microfones

Outro ponto importante para se atentar é o atraso temporal ‘ τ ’ que é calculado pela razão entre distância da defasagem ‘ D ’ e a velocidade da onda sonora ‘ v ’, representado na equação 2.2 e quando multiplicamos esse atraso temporal pela frequência angular, temos a defasagem angular ‘ φ ’, representada na equação 2.3. A defasagem angular pode ser remodelada para que fique em função do comprimento de onda e assim surge a equação 2.4.

$$\tau = \frac{d * \text{sen}(\theta)}{v} \quad (2.2)$$

$$\varphi = \tau * \omega = 2\pi f * \frac{d * \text{sen}(\theta)}{v} \quad (2.3)$$

$$\varphi = \tau * \omega = 2\pi * \frac{d * \text{sen}(\theta)}{\lambda} \quad (2.4)$$

Entrando no assunto mais aprofundado sobre o atraso entre cada microfone e tomando o microfone A da Figura 2.3 como referência, entretanto, as equações são as mesmas no caso de usar o microfone B como referência. Com uma quantidade maior de microfones, por exemplo 4 microfones, pode-se empregar uma análise semelhante à feita para os dois microfones. Assim, a defasagem do terceiro microfone será o dobro da defasagem do segundo e a defasagem do quarto microfone será triplo da defasagem do segundo. É notado que por mais que adicione microfones ao sistema, um padrão é seguido e isso é originado pela forma como os microfones são dispostos, obedecendo sempre uma distância de ‘ d ’ cm entre si. Por esse padrão, é possível montar uma matriz, onde $x_i(t)$ é a saída do i -ésimo microfone, em um total de M microfones, sendo $x_0(t)$ a saída do microfone A que foi tomado como referência, representado na equação matricial 2.5.

$$\begin{bmatrix} x_0(t) \\ x_1(t) \\ x_2(t) \\ \vdots \\ x_{M-1} \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j\varphi} \\ e^{-2j\varphi} \\ \vdots \\ e^{-(M-1)j\varphi} \end{bmatrix} s(t) + \begin{bmatrix} n_0(t) \\ n_1(t) \\ n_2(t) \\ \vdots \\ n_{M-1} \end{bmatrix} \quad (2.5)$$

Na equação matricial 2.5 ainda é possível notar um termo adicional na saída de cada microfone representado por $n_i(t)$ que é o ruído, e por fim, toda essa matriz só é válida se o sinal sonoro ‘s(t)’ for um sinal limitado em banda estreita. O sinal sonoro ‘s(t)’ está representado matematicamente na equação 2.6.

$$s(t) = b(t)e^{j\omega t} \quad (2.6)$$

2.3 DISTÂNCIA ENTRE OS MICROFONES

Pelo modelamento do problema, como será calculado o arco seno para estimar o ângulo de incidência, é necessário que este ângulo seja menor que 180° . Por isso, o ângulo ‘ θ ’ da Figura 2.2 ser entre $-90^\circ \leq \theta \leq 90^\circ$. Como queremos descobrir a maior distância possível entre os microfones e tendo que a maior diferença de trajeto ocorre em ‘ θ ’ igual a 90° , usamos o ângulo de defasagem e o ângulo θ com valores de 180° e 90° , respectivamente. Com essas informações e usando a equação 2.4 como ponto de partida, temos:

$$\begin{aligned}
 \varphi &= 2\pi * \frac{d * \text{sen}(\theta)}{\lambda} \\
 180^\circ &= 2\pi * \frac{d * \text{sen}(90^\circ)}{\lambda} \\
 \pi &= 2\pi * \frac{d * 1}{\lambda} \\
 d &\leq \frac{\lambda}{2} \quad (2.7)
 \end{aligned}$$

Essa condição possibilita a estimação adequada do ângulo de chegada do sinal (DOA). Tal relação é tratada como versão espacial do Teorema da Amostragem de Nyquist.

Neste trabalho será usada uma onda senoidal de 1kHz que propaga pelo ar com uma velocidade de 343,4 m/s a 25 °C. Esses valores aplicados a equação 2.7, resultam que a distância que deve ser usada entre os microfones deverá ser menor que 17,17 cm. Assim, para não se trabalhar no limite e por praticidade, o valor adotado para a distância 'd' foi de 15 cm.

Com esse valor de 15 cm de afastamento entre os microfones, a maior frequência que o projeto pode operar sem apresentar erros na estimação do azimute é de 2,289 kHz.

2.4 ALIASING

Quando o Aliasing ocorre, provavelmente, resulta em uma estimação errada. Assim, esta é uma situação a ser evitada.

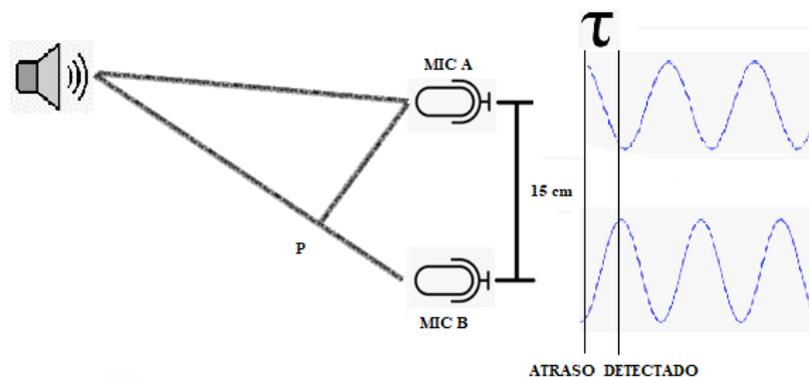


Figura 2.4: Situação em que o atraso é detectado e é adequado

A distância P-B dividida pela velocidade do som é o atraso temporal discutido anteriormente. Na ocasião da Figura 2.4, o mesmo som chega ao mesmo tempo em A e em P e a distância P-B depende do espaçamento entre os microfones 'd', tendo que se 'd' aumenta, P-B também aumentará partindo do preceito que P-B é igual a $d \cdot \sin(\theta)$.

A Figura 2.4 representa uma situação em que não ocorre o Aliasing e o atraso detectado é o atraso dentro da faixa esperada pelo fato da distância P-B ser menor que $\lambda/2$.

O problema do Aliasing surge quando essa distância P-B é maior ou igual a $\lambda/2$. Como é mostrado na Figura 2.5, a distância P-B é maior que $\lambda/2$, portanto, o sistema detecta atraso

errado e, é claro, faz uma estimativa equivocada. Na Figura 2.5, as condições foram exageradas de forma a caracterizar o problema. Vale ressaltar que para o caso do ângulo ' θ ' igual a zero em que o sinal recebido não tem atraso entre os microfones, o sistema enfrenta um grande problema e, em geral, fornece a estimativa errada. É necessário um outro método para estimar tais casos.

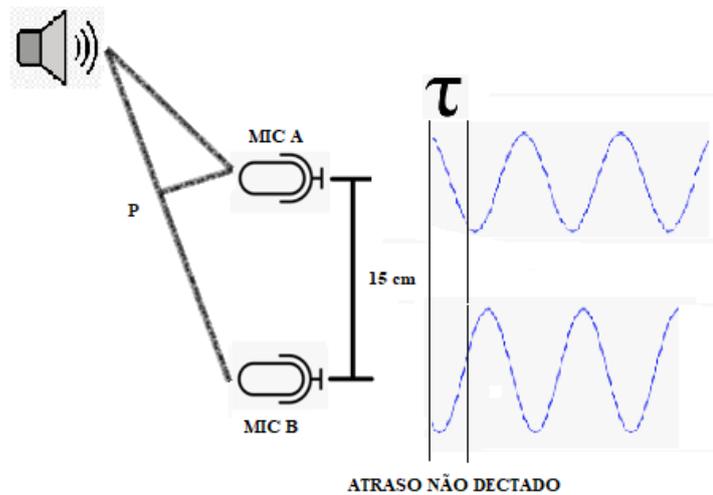


Figura 2.5: Situação em que o atraso detectado não é adequado

Vimos que a condição para não ocorrer o Aliasing é respeitar a distância P-B menor que $\lambda/2$ o que implica em limite para a frequência do sinal. Se este limite for ultrapassado, ainda se podem fazer estimativas corretas, desde que se limite o ângulo de incidência do sinal. Por exemplo, a Figura 2.6 ilustra este caso, onde se deve operar com a fonte sonora dentro da faixa marcada em amarelo.

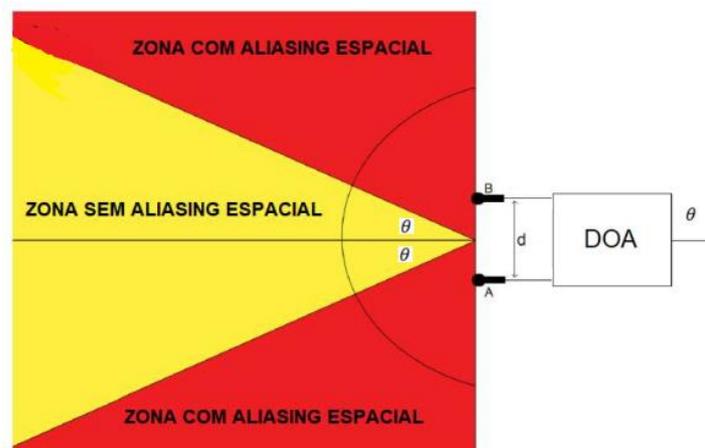


Figura 2.6: Região com/sem a presença do Aliasing Espacial

3 HARDWARE

O foco desse capítulo é a apresentação de toda parte física do projeto desde o sistema de captação, amplificação e processamento do sinal sonoro. Além de indicar como foi feito o envio dos dados do MSP430F5529 para o PC (Matlab).

3.1 MSP430F5529

Os MSP430 são microcontroladores RISC-like de 16 bits voltados para aplicações de baixo consumo de energia com conjuntos de periféricos direcionados para uma variedade de aplicações. Implementados em linguagem C, utilizando um software chamado Code Composer Studio que possibilita o usuário elaborar e executar diversos algoritmos. Tanto o MSP430 quanto o software Code Composer Studio são desenvolvidos pela Texas Instruments, sendo o software disponibilizado de forma gratuita. Como dito anteriormente, o microcontrolador possui uma CPU RISC-like de 16 bits, além de, cinco temporizadores de 16 bits, um conversor analógico-digital de 12 bits de alto desempenho (ADC), quatro USCIs, um multiplicador de hardware, um controlador para acesso direto à memória (DMA), um módulo Real-Time Clock (RTC), 63 pinos, 128KB de memória flash e 8KB de memória RAM. O microcontrolador está mostrado na Figura 3.1 de forma mais esquemática e na Figura 3.2 em sua forma física.

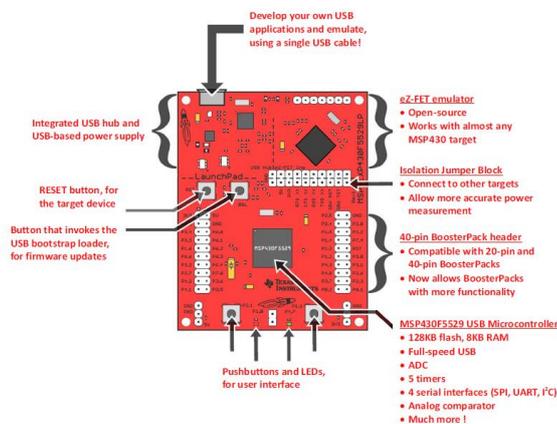


Figura 3.1: MSP430F5529 com suas características

Fonte: [MSP-EXP430F5529LP | Mouser](https://www.mouser.com/ProductDetail/Texas-Instruments/MSP-EXP430F5529LP)

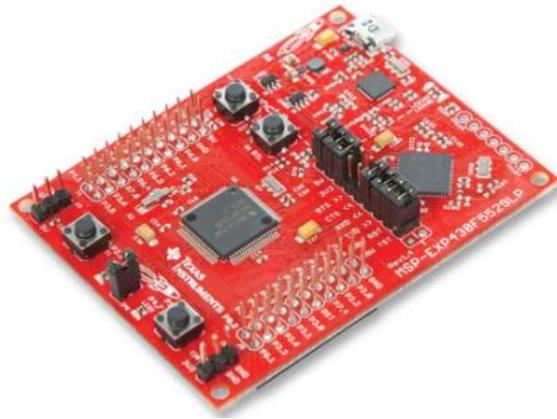


Figura 3.2: MSP430F5529

Fonte: [MSP-EXP430F5529LP | Farnell](https://www.farnell.com/datasheets/1664221.pdf)

3.2 MICROFONE E CIRCUITO DE PRÉ-AMPLIFICAÇÃO

Todo esse conjunto de hardware tem a função de captar o sinal sonoro e convertê-lo em um sinal elétrico. Os microfones utilizados foram previamente projetados em um trabalho anterior voltado para o estudo de métodos para a estimativa da direção de chegada de sinais sonoros.



Figura 3.3: Microfone Omnidimensional e seu pré-amplificador

O microfone omnidimensional da Figura 3.3 tem resposta linear adequada para o intervalo de frequência correspondente à faixa espectral da voz humana (50 a 3400 Hz) e uma relação de sinal – ruído de 62dB. A curva de resposta em frequência de 20 a 20 kHz para este microfone está representada na Figura 3.4.

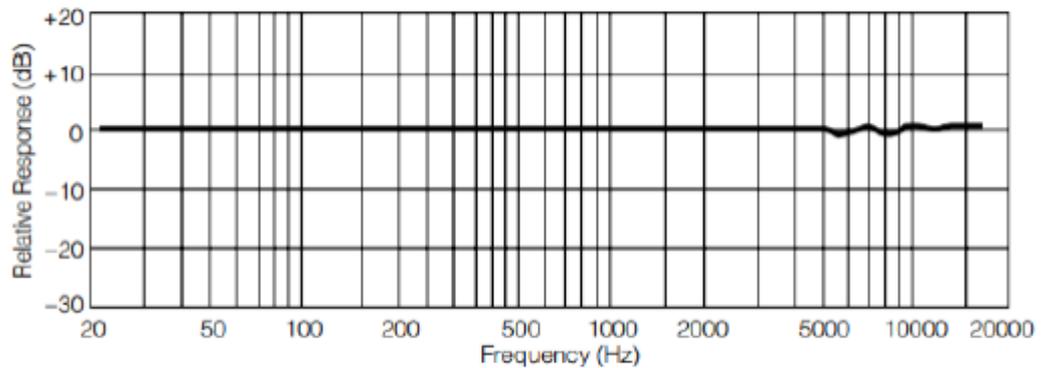


Figura 3.4: Curva de resposta em frequência do microfone

O microfone foi conectado a uma placa de circuito impresso que contém quatro amplificadores operacionais presentes no chip LME49740² que é responsável pelo primeiro tratamento do sinal captado. De acordo com o esquema apresentado na Figura 3.5, o sinal captado pelo microfone é entregue a um amplificador de instrumentação e depois passa por um filtro passa-baixas com frequência de corte de 4,378 kHz e que também tem a função de operar como buffer de saída.

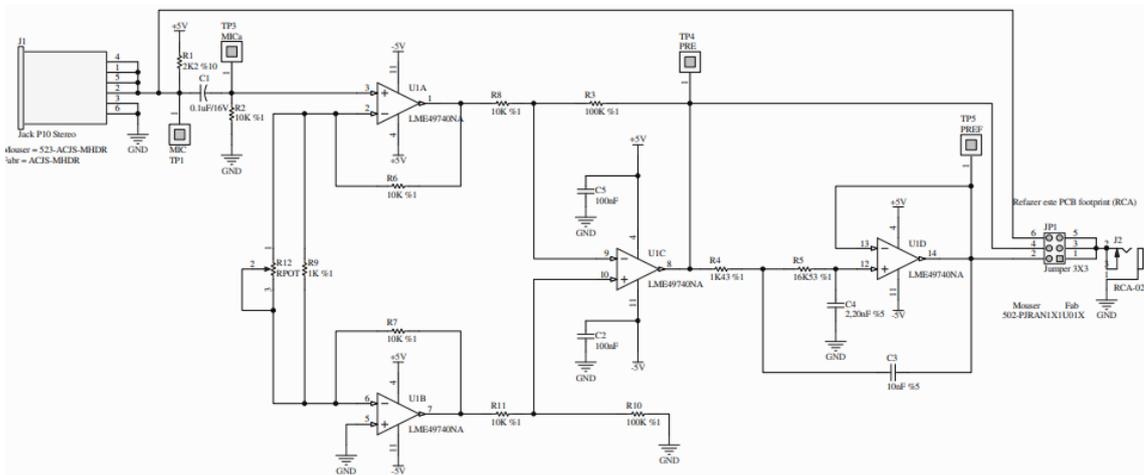


Figura 3.5: Circuito do estágio de pré-amplificação

Esse sinal da saída do buffer (pino 2 do bloco JP1 do circuito impresso) é conectado ao pino P6.0 ou P6.1 do MSP430 a depender do canal que o microfone ficará responsável por ser. A referência nomeada de “-5V” no circuito impresso é ligada à terra do MSP430F5529.

A conexão do pré-amplificador com o MSP merece uma explicação mais cuidadosa. O circuito usado como pré-amplificador foi aproveitado do hardware construído para uma tese de

² Fonte: [LME49740 Datasheet\(PDF\) - National Semiconductor \(TI\) \(alldatasheet.com\)](https://www.ti.com/lit/ds/lme49740)

mestrado. Este amplificador trabalha com alimentação simétrica, na faixa de 3V até 12 V. Em tal configuração, como era de se esperar, o sinal de saída tem o valor médio igual a zero e excursiona em valores positivos e negativos. Isto é um problema para o atual projeto porque o conversor ADC do MSP430 só opera com tensões positivas e na faixa de 0V até 3,3 V.

A solução adotada está ilustrada na Figura 3.6. A alimentação simétrica (+3V, -3V) é gerada a partir de 4 baterias AA 1,5V, garantindo então a alimentação do pré-amplificador. O uso de baterias em tal circuito tem a grande vantagem de garantir uma alimentação muito “limpa”. O terra do MSP430 é ligado aos -3V e o sinal de saída do pré-amplificador, que excursiona na faixa de 6V é dividido por 2 com o uso do divisor resistivo formado pelos dois resistores de valor ‘R’.

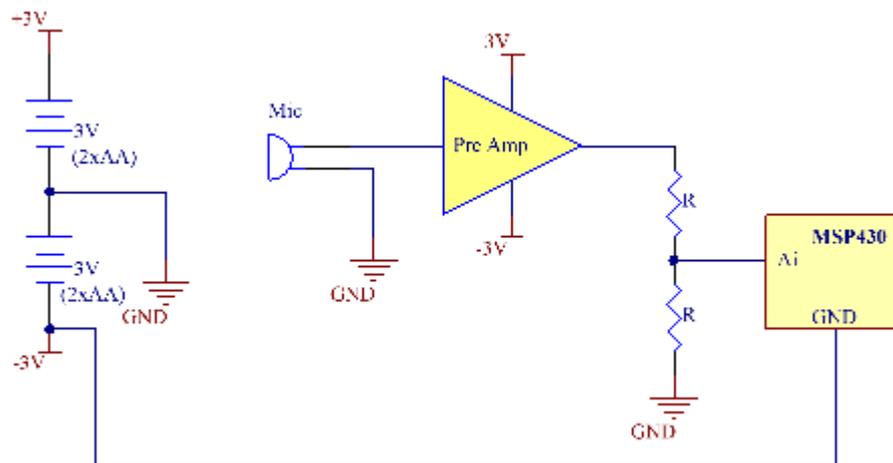


Figura 3.6: Esquema de conexão entre baterias, pré-amplificador e MSP430 apenas com um microfone

Em suma, o MSP430 recebe um sinal com valor médio perto de 1,5V e excursionando de 0V até 3V, o que garante o atendimento de suas especificações. Há um outro ponto que merece ser comentado que é a impedância de saída deste circuito. De forma bastante simplificada, ela pode ser considerada igual a $R/2$. Em trabalhos anteriores, o valor usado de ‘R’ era igual a 10 k Ω e após a mudança de ‘R’ para 1 k Ω , foi verificada uma diferença apreciável no desempenho, fato que será comentado mais adiante.

É ainda interessante expor que este esquema com o divisor resistivo trouxe outra vantagem. Um primeiro ensaio com o pré-amplificador alimentado apenas com 0V e 3V, para atender às exigências do MSP, oferecia uma faixa linear entre 1,0 V até 2,0 V, ou seja, apenas uma faixa útil de 1,0V. Uma grande perda, já que o ADC opera de 0V até 3,3V. Com a alimentação simétrica de +3V e -3V, a excursão do sinal ficou muito boa entre -2V e +2V. Após

o divisor resistivo, a saída excursiona com qualidade desde 0,5 V até 2,5, permitindo um melhor aproveitamento da faixa dinâmica do ADC.

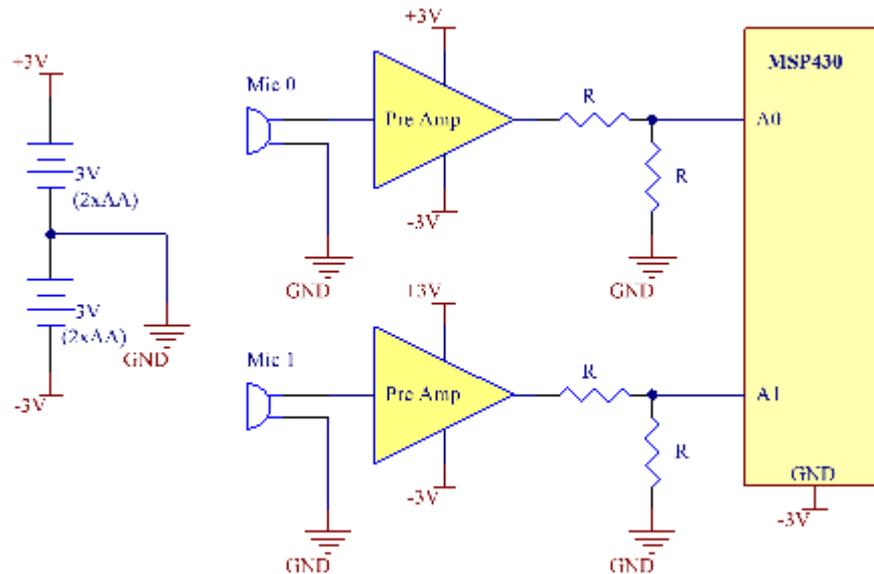


Figura 3.7: Esquema enfatizando a alimentação dos pré-amplificadores e suas conexões com o MSP430

3.3 MEMÓRIA EXTERNA

A memória interna do MSP430 é de apenas 8 KB, o que limita a armazenagem em 4 K valores inteiros. Para a taxa de aquisição que se opera (50 kHz), ela permitiria armazenar apenas 80 ms das aquisições provenientes dos microfones. Além disso, essa memória interna é usada para as variáveis do programa do usuário e para a pilha da CPU. Desde o início do projeto era óbvia a necessidade de uma memória maior. Por isso, se decidiu usar a memória externa para armazenar todos os dados captados pelos microfones e que foram convertidos pelo conversor analógico-digital. Foram utilizados dois chips 23LC1024³ que são produzidos pela Microchip, possuem memória de 128KB cada e atuarão como periféricos de memória SRAM. Para uma aquisição com taxa de amostragem de 25 kHz para cada microfone, o tempo de dados armazenados é equivalente a 2,6 segundos.

³ Fonte: [23LC1024 | Microchip Technology](https://www.microchip.com/pt-br/products/23LC1024)



Figura 3.8: CHIP 23LC1024 - Memória externa

Fonte: [23LC1024 | Microchip Technology](#)

A comunicação entre o MSP430F5529 e a memória externa é pelo barramento SPI. A comunicação SPI se baseia no processo Mestre e Escravo, onde o mestre é o que gerencia o barramento de comunicação, responsável por gerar a linha de sincronismo (SCK) além de inicializar e finalizar as transmissões, enquanto o escravo só atua quando acionado pela linha de sincronismo. A comunicação SPI é bidirecional, também chamada de full-duplex, enquanto o mestre envia os dados pela linha SIMO (Master Output Slave Input), o escravo envia dados para o mestre pela linha SOMI (Master Input Slave Output). Dessa forma, existem duas linhas de comunicação, uma linha para estabelecer o sincronismo da comunicação e outra linha para realizar a transmissão efetiva de dados entre mestre e escravo. As conexões estão apresentadas na Figura 3.9.

No projeto foram feitas funções de escrita e leitura na memória, além de configuração da SPI para facilitar o manuseio do programa.

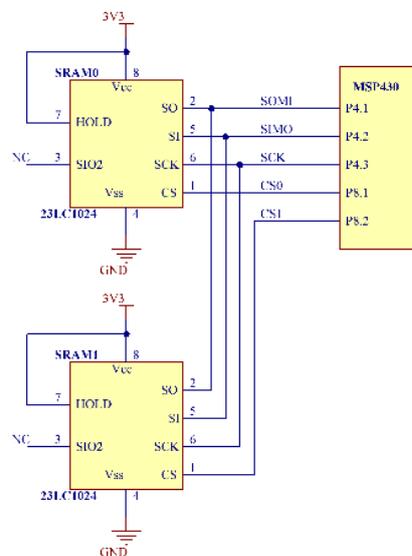


Figura 3.9: Conexões entre as duas memórias externas e o MSP430

3.4 BLUETOOTH

O dispositivo Bluetooth utilizado é o HC-05 que é de fácil utilização e projetado para conexão serial (UART). Isto dispensa a conexão física entre o sistema projetado e o computador. Sendo Bluetooth V2.0 + EDR com taxa máxima de transmissão de 1382400 bps. Ele usa Bluecore CSR 04 – Sistema Bluetooth de chip externo único com tecnologia CMOS e com AFH (Recurso Adaptativo de Salto de Frequência). Dimensões de 12.7mm por 27mm.

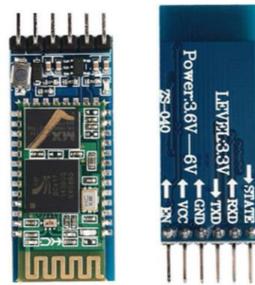


Figura 3.10: Módulo Bluetooth HC-05

Fonte: [Módulo Bluetooth Hc-05 -e](#)

O módulo bluetooth nesse projeto não faz parte da função principal de estimar a direção de chegada de sinais sonoros, mas é item principal para entender o que acontece com as funções escritas no Code Composer Studio, pelo fato de permitir a troca de informações entre o computador com o programa MATLAB e o microcontrolador. Desta forma se consegue validar as operações, usando os recursos gráficos do Matlab os dados adquiridos, o desempenho dos filtros etc. A Figura 3.11 demonstra como é feita as conexões físicas entre MSP430 e o HC-05.

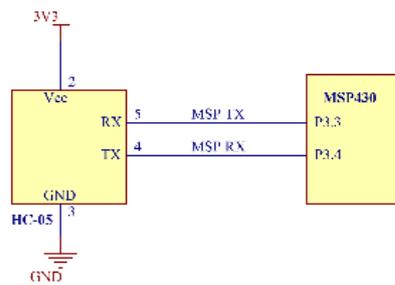


Figura 3.11: Conexão física do HC-05 ao MSP430F5529

3.4.1 Configuração do HC-05

Nesse projeto o HC-05 foi configurado para operar numa taxa de transmissão de 460800 bps e para isso algumas etapas são necessárias para que o módulo bluetooth funcione adequadamente. As configurações de fábrica do módulo são 9600 bps, 8bits, 1 bit de parada, sem paridade, por isso a necessidade de configuração.

Primeiramente foi pareado o HC-05 ao computador com a senha de conexão padrão de “0000” ou “1234”. Após isso, o módulo Bluetooth precisa ser energizado com o seu botão físico pressionado para que entre no modo Comandos AT (modo de configuração). Quando preparado para responder aos Comandos AT, taxa passa a ser de 38.400 bps. Usando tais comandos se pode alterar diversas configurações do HC-05, dentre elas a velocidade de comunicação que passou a ser de 460.800 bps (aproximadamente 0,5 Mbps).

A programação desta velocidade é feita por um programa isolado, escrito especialmente para realizar esta operação. Uma vez feito essa configuração, o HC-05 salva esses dados e toda vez que for religado, já estará pronto para uso com as últimas especificações.

4 SOFTWARE PARA ESTIMATIVA DOA

Neste capítulo será discutido as principais ideias por trás dos códigos implementados, bem como, a melhora no tempo mínimo necessário para a transferência de dados entre o MSP430F5529 e o computador por meio do HC-05.

4.1 FIRMWARE

Todo um processo de criação de funções e scripts foram necessários antes de chegar na função principal do código, além de, ser configurado elementos de controle tais como pinos de entrada e saída, botões, timers, interrupções, conversor analógico-digital (ADC), transmissão serial SPI com a memória externa e inicialização do LCD.

Para facilitar as etapas de testes de partes do programa principal foi criado um esquema para operação no que se chamou de modos. Ao todo, foram criados vinte modos vazios que no decorrer do tempo foram sendo preenchidos para testes. Isto permitiu, com o mesmo programa, a execução de vários ensaios sem que acontecesse interferência entre eles.

Quadro 4.1 - Modos do programa

		MODOS	
01	Livre	11	Chaves
02	DOA - Apresentação	12	HC-05 (Incompleto)
03	Livre	13	SRAM
04	Livre	14	Energia
05	Livre	15	ADC+DMA+Matlab+ Sinal Triangular
06	Livre	16	Seriais
07	Livre	17	Serial por DMA
08	Livre	18	Livre
09	Livre	19	Teste das funções de escrita na memória
10	Leds	20	DOA com MATLAB

Os modos são acessados por meio do envio de seu respectivo número pelo Bluetooth via terminal serial, no caso desse trabalho foi utilizado o terminal Tera Term⁴.

Para que o processo de conversão analógico-digital tenha maior velocidade, foi feito a configuração do conversor A/D de modo a usar 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 MSP. O controlador de DMA libera o barramento da CPU e permite armazenar diretamente o valor digitalizado na memória do MSP sem a necessidade de usar a CPU para essa tarefa. Além de configurarmos o modo de operação de cada um desses recursos, configuramos os timers TA0.1 para a operação do conversor A/D e do DMA.

Para a apresentação de etapas de processamento e estimativa final da direção de chegada do sinal sonoro, realizamos a configuração de um LCD. Tal periférico se comunica usando protocolo I²C com o MSP430F5529. Com o objetivo de ter uma maior velocidade na atualização do LCD, foi usado o DMA. Então, quando escrevemos no LCD, na verdade, escrevemos em um buffer de memória e toda vez que esse buffer é atualizado, o DMA é habilitado e envia as alterações para o LCD.

O envio do conteúdo da memória externa para o PC envolve duas operações: acesso SPI para ler as memórias externas e transmissão serial para o HC-05. Essas duas operações são realizadas simultaneamente com o uso de dois buffers de 128 bytes. Enquanto um buffer está sendo preenchido com os dados da memória externa (SPI) o outro está sendo transmitido pela porta serial para o HC-05 via DMA.



Figura 4.1: Etapas da transferência dos dados da memória externa para o computador

A transferência dos dados das duas memórias não ocorre ao mesmo tempo, sendo que primeiro é transmitido todos os dados de uma memória externa e depois todos os dados da outra memória externa. O tempo necessário para enviar todos os dados de uma memória externa ao computador está em torno de 6 segundos, totalizando 12 segundos para envio dos dados das duas memórias.

⁴ Fonte: [Tera Term - Download \(softonic.com.br\)](http://teraterm.sourceforge.net/)

Essa alteração do processo de transmissão dos dados foi de grande valia. Nos trabalhos anteriores, a taxa de transmissão utilizada era de 115200 bit/s, o processo de leitura e transmissão dos dados da memória externa ao computador não acontecia ao mesmo tempo, utilizava o Arduino para fazer a mediação do processo e levava em torno de um minuto e meio para transmitir os dados das duas memórias. Essa demora prejudicava no tempo gasto para entendimento da funcionalidade de algumas funções contidas no projeto. Com a alteração do processo de transmissão, as etapas de teste em 5 TESTES PRÁTICOS foram mais rápidas e práticas de serem realizadas.

O modo 2 que contém a programa final para estimar a direção de chegada do sinal sonoro que faz uso de 64 regiões do sinal adequadas para esta operação. O sistema estima a direção de chegada para cada região e salva em um vetor. Se ocorrer 64 regiões adequadas para a estimativa, é calculado a moda desse vetor, com as 64 estimativas, que resultará na estimativa final do ângulo de chegada. Esse resultado é mostrado no canto superior direito do LCD seguindo de uma exclamação (essa exclamação indica que o número que aparece foi o da atual estimativa). Se em uma futura aquisição, não for possível identificar as 64 regiões adequadas, o programa não calcula a moda do vetor que guarda as estimativas de cada região e no LCD aparece o resultado anterior que originou das 64 regiões adequadas, mas sem ser seguido de uma exclamação. Ou seja, para o programa resultar uma estimativa é necessário ter 64 regiões adequadas, caso contrário, ainda é mostrado no LCD o resultado da última estimativa que tiveram 64 regiões adequadas.

4.2 MÉTODO DE ESTIMAÇÃO DA DOA

O azimute da fonte sonora pode variar entre $-90^\circ \leq \theta \leq 90^\circ$. Neste trabalho não foi considerado os casos em que a fonte sonora está atrás dos microfones

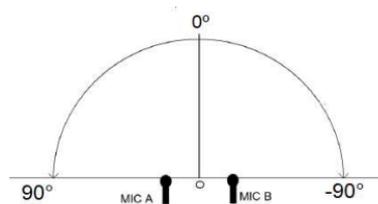


Figura 4.2: Ângulos possíveis para estimativa com dois microfones

Podemos equacionar o sinal recebido pelos microfones como uma função do sinal original multiplicado por um ganho oriundo dos amplificadores associados aos microfones e somados a um ruído.

O conversor analógico-digital trabalha a uma taxa de 50 kHz, sendo 25 kHz para cada microfone. Como a estimativa DOA é obtida através de um processo de correlação entre os sinais referentes a cada microfone, a estimativa se torna mais eficiente ao realizarmos este processo com um conjunto grande de valores.

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:

$$f_{ams} = \frac{\text{amostra}}{\text{tempo}} \quad (4.1)$$

Para se relacionar a k-ésima amostra em tempo discreto com o atraso temporal, uma análise dimensional pode ser feita como é mostrado na equação 4.2.

$$tk = \frac{k}{f_{ams}} = \frac{\text{amostra}}{\frac{\text{amostra}}{\text{tempo}}} = \text{tempo} = \frac{d \cdot \text{sen}(\theta)}{v} \quad (4.2)$$

Analisando a equação 4.2, podemos obter uma equação para determinar o ângulo ‘ θ ’ de chegada do sinal sonoro. Nela, temos que ‘ v ’ = 340 m/s é a velocidade do som, ‘ d ’ = 15 cm é a distância entre os microfones e ‘ f_{ams} ’ = 25 kHz é a frequência de amostragem do conversor A/D para cada microfone. A variável ‘ k ’, por sua vez, é o atraso relativo entre os sinais, calculados pela correlação temporal entre os sinais digitalizados. Assim, a direção de chegada é dada pela equação 4.3.

$$\theta = \arcsen\left(\frac{k \cdot v}{d \cdot f_{ams}}\right) \quad (4.3)$$

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.

$$f_{ams} \geq 2 \times f(\text{fonte sonora}) \quad (4.4)$$

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

A frequência utilizada tem seu valor de 1 kHz, logo o valor mínimo para a frequência de amostragem é 2 kHz. Entretanto, como pode ser visto na equação 4.3, o ângulo de azimuth é inversamente proporcional a frequência de amostragem, isso significa que quanto maior for 'fams', menor será a resolução da faixa de entre um ângulo e o outro e melhor será o ângulo máximo que o projeto poderá trabalhar. Logo uma maior frequência de amostragem é desejada. Por fim, optou-se por uma frequência de amostragem para cada microfone igual a 25 kHz e frequência de amostragem para o conversor analógico-digital igual a 50 kHz para converter os dois microfones.

4.2.1 Faixa de ângulo suportado

Utilizando a equação 4.3 e substituindo os valores, temos os seguintes ângulos para cada atraso indicado pela variável "k".

Quadro 4.2: Valores reais de ângulos do projeto

k	Ângulo
1	5,2015°
2	10,4474°
3	15,7833°
4	21,2640°
5	26,9775°
6	32,9563°
7	39,3952°
8	46,4965°
9	54,6863°
10	65,0940°

Dados esses valores e para poupar memória do MSP430, foram definidos valores inteiros para representar a estimativa do azimute calculado pelo programa. O Quadro 4.1 representa como foram relacionados os atrasos com as possíveis respostas.

Quadro 4.3: Valores de ângulos usados no programa

k	Ângulo
1	5°
2	10°
3	15°
4	21°
5	27°
6	33°
7	40°
8	47°
9	55°
10	66°

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 dobrar a quantidade de amostras com a interpolação spline cúbica, e por fim, é possível aumentar o valor de 'd' para melhorar a precisão, entretanto é necessário manter o critério de meio comprimento de onda.

4.2.2 Etapas da estimativa

A estimativa da direção de chegada do sinal recebido pelos microfones passa por 3 etapas, sendo primeira a aplicação do filtro digital (média móvel), a segunda a identificação de maior faixa de energia e a terceira e última, a estimativa de fato.

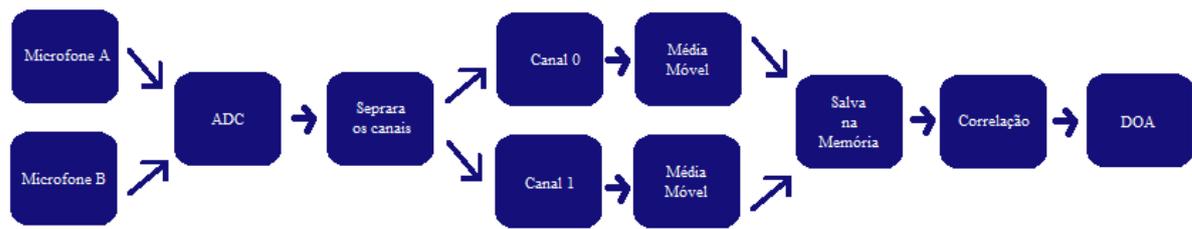


Figura 4.3: Diagrama de bloco do sistema

4.2.2.1 Primeira etapa: filtro média móvel

Os dados convertidos e salvos na forma de vetor na memória externa estão sujeitos a ruídos oriundos dos processos de conversão presentes na captação dos microfones e na digitalização do conversor A/D. Os ruídos adicionados ao sinal original representam um elevado risco para a eficiência da estimativa de DOA. Por isso, é recomendável a aplicação de filtros capazes de atenuar tais ruídos.

Neste projeto, é utilizado o filtro digital chamado de “Média Móvel”, um filtro de categoria FIR – Resposta Finita ao impulso, do inglês Finite Impulse Response, que se trata de um processo estatístico e suaviza flutuações curtas. Este filtro atua nos dados recém-convertidos e salvos na memória externa e determina o valor de cada amostra sendo igual à média entre ‘n’ valores adjacentes a ele como mostrado na equação 4.1. A quantidade de elementos adjacentes tomados para o cálculo determina a ordem do filtro e para esse projeto foi determinado que esse filtro teria uma ordem 4.

$$Média\ Móvel = \frac{1}{n} \sum_{i=1}^N p_i \quad (4.1)$$

4.2.2.2 Segunda etapa: identificação da faixa de maior energia

Com os dados da memória externa filtrados pelo filtro de Média Móvel, é iniciado a segunda etapa que identifica a faixa de maior energia captada pelos microfones, que passa a ser a de maior interesse para a estimativa da DOA.

Nessa etapa, a memória externa é varrida em blocos de 128 inteiros (16bits) e calculamos o desvio médio desse intervalo. O Desvio Médio simples, representado pela equação 4.2, é uma medida da dispersão dos dados em relação à média de uma sequência, o “afastamento” em relação a essa média. Esta medida representa a média das distâncias entre cada elemento da amostra a seu valor médio.

$$Desvio\ Médio = \frac{\sum_{i=1}^N |x_i - \bar{x}|}{N} \quad (4.2)$$

Na equação 4.2, ‘N’ representa a quantidade total valores inteiros e \bar{x} representa a média aritmética desses valores.

Para determinar se uma certa região de 128 inteiros é adequada para se fazer uma estimativa do ângulo de chegada foi estipulado um valor de limiar. Se o valor do desvio médio dessa região for maior que esse limiar, então a região é adequada para estimação. Isto permite recusar intervalos de silêncio ou de baixa amplitude do sinal.

4.2.2.3 Terceira etapa: Estimativa

O atraso temporal entre os sinais de cada microfone é equivalente ao instante em que o mesmo sinal deslocado temporalmente passa a ficar muito semelhante ao sinal original.

Para encontrar o atraso temporal, é feito a verificação da similaridade entre os dois sinais por meio da correlação entre eles.

A rotina que faz a correlação entre os sinais, calcula a diferença entre os elementos de uma mesma posição, aplica o valor absoluto e soma todos esses valores. Após isso, divide pelo número de elementos do mesmo vetor. Como é mostrado na equação 4.3

$$Erro = \frac{\sum_{i=1}^N |v_{1i} - v_{2i}|}{N} \quad (4.3)$$

Essa operação para achar a posição de menor diferença é com deslocamentos, iniciando em 1 e indo até um limite n . A cada operação, essa diferença é comparada ao resultado da diferença anterior e caso apresente um valor menor, a variável diferença mínima será atualizado para este novo valor encontrado e a variável k recebe o valor do deslocamento que representa quantas vezes transladou o sinal até se encontrar a menor diferença.

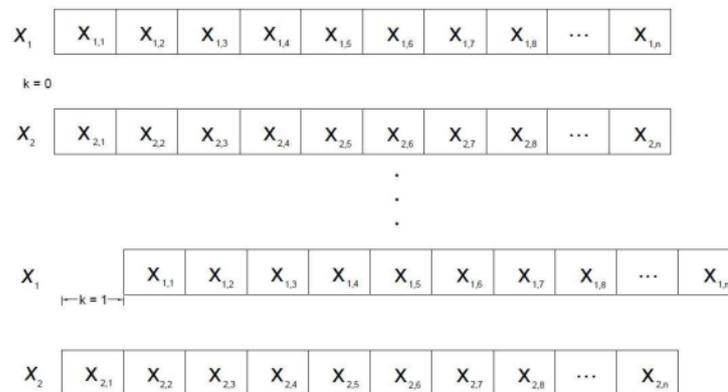


Figura 4.4: Representação de um deslocamento para se calcular a diferença entre os dois sinais.

Para fins de melhor entendimento, foi criado um programa⁵ no MATLAB que sintetiza dois sinais senoidais de mesma amplitude, mesma frequência de 1 kHz, com taxa de amostragem de 25 kHz e 100 amostras. O primeiro sinal denominado X_1 não tem defasagem e o segundo sinal denominado de X_2 apresenta uma defasagem de 5 amostras, como mostrado na Figura 4.5.

⁵ Fonte: [Seno - Google Drive](#)

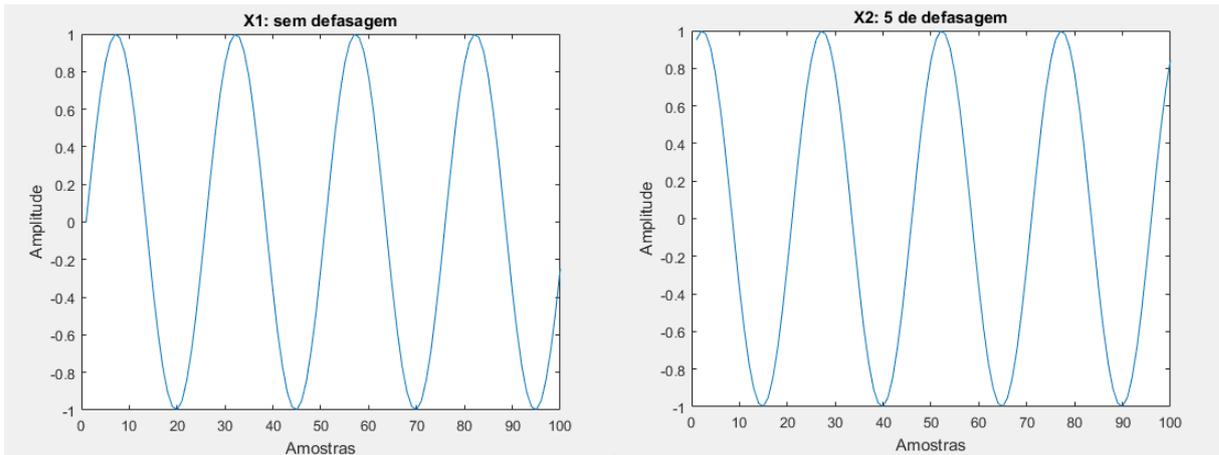


Figura 4.5: Senoides sintetizadas pelo MATLAB

Ao fazermos a operação de correlação entre as duas senoides, obtemos a Figura 4.6. Sendo que o eixo das abcissas representa a diferença indicada pela equação 4.3 para cada deslocamento da forma indicada pela Figura 4.4. A linha em azul indica as diferenças quando se desloca X1 em relação e X2 e a linha em vermelho as diferenças caso se desloque X2 em relação a X1. Temos dois valores candidatos: um indicado pela linha em azul e o outro pela linha em vermelho. Usa-se o menor deles, que no caso ficou em 5 amostras.

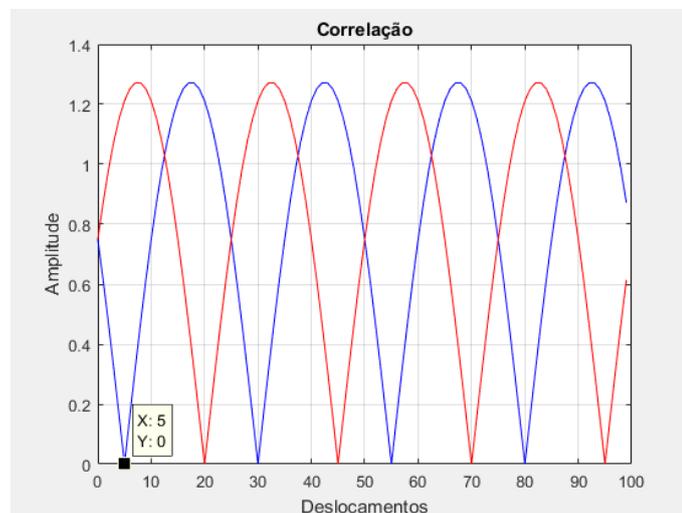


Figura 4.6: Gráfico da correlação

O tamanho do atraso entre os dois sinais é dado pelo menor valor do gráfico da correlação, que se repete periodicamente. Esse é o valor de 'k' necessário para determinar o azimuth da fonte sonora. Como na Figura 4.5, a diferença era de 5 amostras, na Figura 4.6, o menor valor do gráfico deu no quinto deslocamento.

Outro ponto que a Figura 4.6 trata é a direção que o sinal está chegando aos microfones. A correlação é feita nos dois sinais como referência, primeiro, o sinal X1 parado e o sinal X2 deslocando, depois, o sinal X2 parado e o sinal X1 deslocando. O valor da correlação que atinge primeiro o menor valor, indica que o sinal que ficou parado no cálculo é o sinal defasado, desse modo, caso o sinal X2 fosse um sinal captado pelo microfone da esquerda, denotaríamos que o sinal vem da direita. Referência da Figura 5.17.

4.3 APERFEIÇOAMENTOS INICIAIS DO SISTEMA

Antes de dar início à descrição deste trabalho, é importante deixar claro que antes de se analisar e reescrever o estimador, foram feitas diversas melhorias no projeto original. Este projeto anterior, por simplicidade e facilidade de programação, usava um Arduino para fazer a conexão serial (38.400 bps) entre o sistema e o PC. Na verdade, o Arduino substituía o módulo bluetooth previsto no projeto. O primeiro passo foi a remoção deste Arduino e a adição de uma placa HC-05 que ficou responsável pela comunicação Bluetooth. O canal serial que fica disponível com o MSP430 e o CCS passou a ser o canal serial básico e foi usado, principalmente, para debug. Em suma, passaram a existir dois canais de comunicação. O estabelecimento da conexão Bluetooth trouxe a grande vantagem de se isolar o circuito do PC, eliminando-se assim uma potencial fonte de ruído.

Um outro ponto melhorado foi o acesso ao LCD. O projeto anterior fazia este acesso via polling era muito lento para realizar esta operação. Em outras palavras, o acesso ao LCD consumia muito tempo e freava todo o sistema. Foi adotada uma nova solução com o uso de um buffer em RAM e a atualização do LCD é toda feita via DMA. As rotinas de acesso ao LCD simplesmente escrevem neste buffer e um timer, periodicamente, verifica se houve alteração deste buffer. Caso positivo, ele dispara o DMA que se encarrega de transferir todo o buffer para o LCD. Assim, com a liberdade de se fazer um certo exagero, podemos dizer que a atualização do LCD não impacta no desempenho do sistema.

Para se ter certeza do funcionamento das rotinas, colocou-se a meta de conferir com o Matlab, todos os cálculos e filtragens executados pelo sistema. Para que isso pudesse acontecer era necessária a rápida transferência de dados para o PC. O projeto anterior fazia essa operação de forma muito lenta. Isto será melhor explicado adiante. De forma breve, indicamos que cada

etapa da estimação envolve a quantidade de 256 KB. A transferência acontecia com os valores impressos em texto (ASCII). Então cada valor inteiro resulta em 6 caracteres (5 dígitos mais o espaço em branco). Tínhamos então a necessidade de enviar 786.432 caracteres (128K x 6). Transferir esta quantidade de bytes na taxa de 115.200 bps toma, cerca, de 70 segundos (o ensaios indicaram uma demora de 1,5 minuto). Um valor muito alto. Com os aperfeiçoamentos e uso do Bluetooth e o uso do formato binário, esta quantidade de dados está sendo transferida em 12 segundos. Depois de todas estas etapas concluídas é que se deu início à reescrituração de todo o firmware e a realização dos testes para comprovação do funcionamento.

5 TESTES PRÁTICOS

5.1 TESTE DO CONVERSOR ANALÓGICO-DIGITAL (ADC)

Quando é recebido o sinal elétrico dos microfones, é necessário que aconteça conversão analógico-digital para armazenamento dos dados na memória externa. Para se ter certeza de seu correto funcionamento, foram feitos testes para saber se essa conversão feita pelo MSP430F5529 estava funcionando perfeitamente.

5.1.1 Teste usando um sinal criado pelo próprio MSP430

A ideia central foi criar um sinal analógico pelo próprio microcontrolador, enviá-lo para o conversor analógico-digital (ADC) e conferir se os valores da conversão representavam o sinal original gerado.

O microprocessador MSP430F5529 não tem um conversor digital-analógico (DAC). Entretanto, tal dispositivo pode ser construído de forma simples usando um gerador PWM e um filtro passa-baixas. O sinal PWM pode ser visto como um sinal PWM com média igual a zero ao qual foi adicionado uma tensão contínua (DC). Essa tensão contínua é igual ao valor médio do sinal PWM, de forma gráfica na Figura 5.1.



Figura 5.1: Decomposição de um sinal PWM em uma componente DC e uma onda retangular de média zero

A ideia é aproveitar o valor DC (valor médio) do PWM para ser a saída do DAC. Para tanto, vamos usar um filtro passa-baixas para filtrar a onda quadrada e deixar passar apenas o valor médio. Conseguimos essa remoção usando um filtro muito simples que é o filtro formado por um resistor (R) e um capacitor (C). A frequência de corte de tal filtro é dada por $1/RC$ rad/s. A Figura 5.2, mostra o filtro com $R = 10 \text{ k}\Omega$ e $C = 100 \text{ nF}$, o que resulta na frequência de corte de 1.000 rad/s . Para obtermos a frequência de corte em Hz, dividimos 1.000 por 2π , e obtemos aproximadamente a 160 Hz ($159,16 \text{ Hz}$).

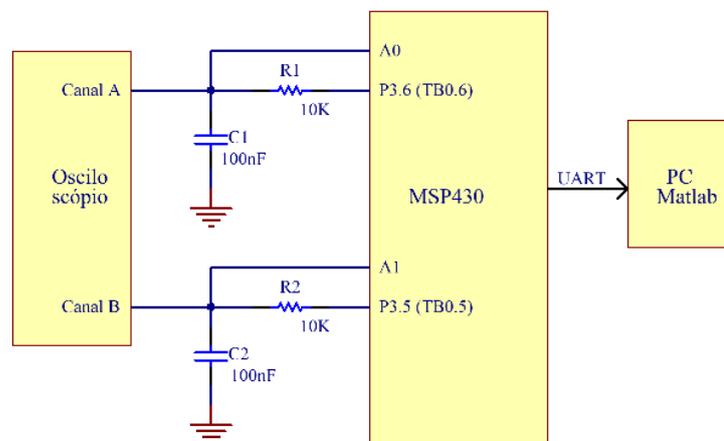


Figura 5.2: Filtro interligado ao MSP430F5529

Para os testes usamos dois sinais diferentes, sendo o canal A0, uma onda triangular de frequência igual a 10 Hz e o canal A1, uma onda triangular de frequência igual a 5 Hz . Esses sinais foram verificados por um osciloscópio e a imagem resultante na tela do osciloscópio está representado na Figura 5.3.

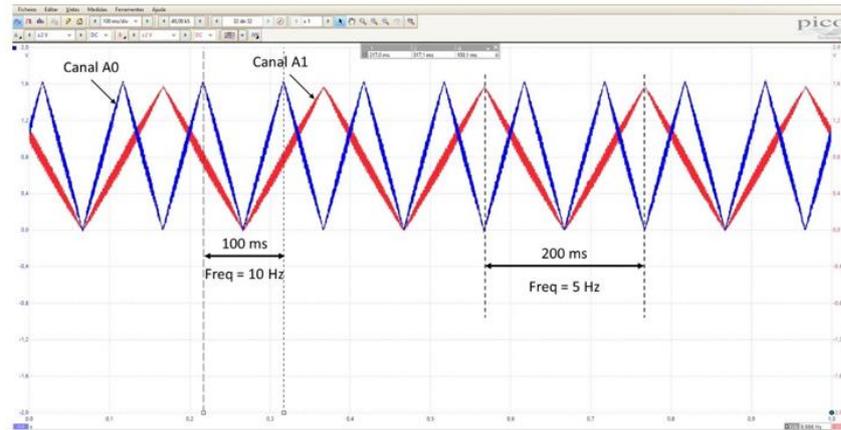


Figura 5.3: Canais A0 e A1 do sinal triangular

Esses sinais dos canais A0 e A1 são entregues ao ADC do microcontrolador e os dados dessa conversão são armazenados na memória externa do MSP430 e posteriormente enviados ao MATLAB para que sejam plotado seus respectivos gráficos. O resultado gerado pelo Matlab a partir dados recebidos está na Figura 5.4. Como se pode ver, os gráficos das Figura 5.3 e Figura 5.4 coincidem.

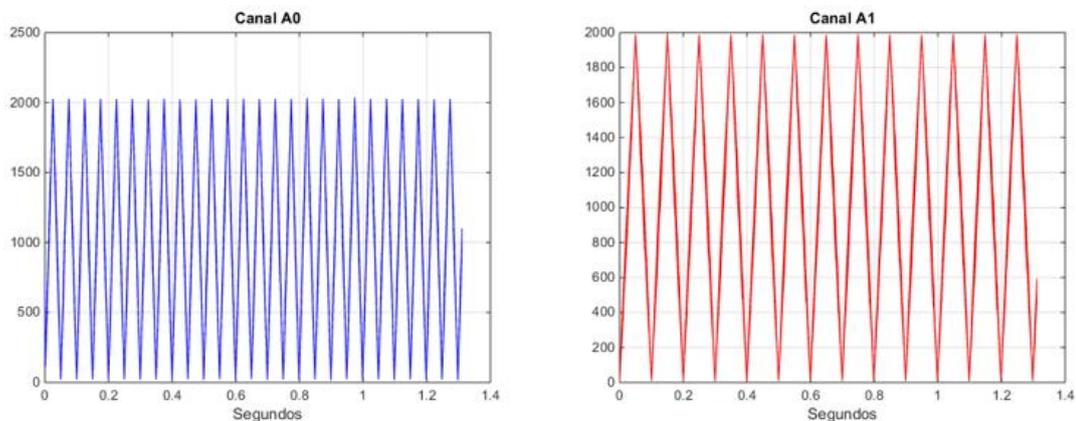


Figura 5.4: Gráficos gerados no MATLAB, frutos dos dados da conversão analógico-digital

Como é visto na Figura 5.4, o processo de conversão analógico-digital feita pelo MSP430F5529 está em perfeito funcionamento, já que a forma de onda e frequência de onda dos dois canais são as mesmas da Figura 5.3. E para que seja visto que a frequência do canal A0 é 10Hz e do canal A1 é de 5Hz foi feita a FFT que está mostrada na Figura 5.5.

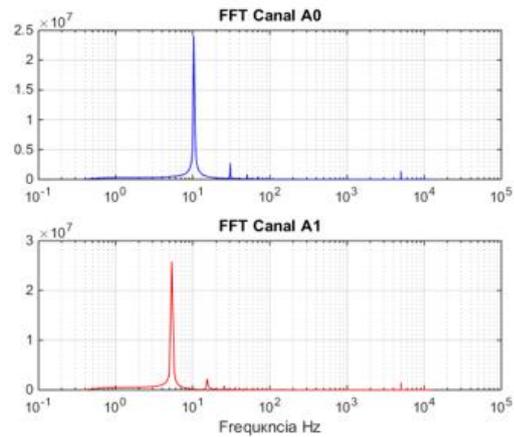


Figura 5.5: FFT dos canais A0 e A1

5.1.2 Teste usando sinais recebidos pelos microfones

Esse teste se baseou em enviar para o MATLAB, o sinal gerado quando não se apresentava sinal aos microfones, ou seja, com os microfones em silêncio.

Do modo como foi configurado o circuito de baterias e a forma de ligação aos microfones, a tensão de saída seria em torno de 3V. Vale ressaltar que para ter exatamente esses 3V, as baterias de 1,5V têm que estar novas. No presente caso, elas já estavam um pouco usadas, o que resultou em uma tensão um pouco mais baixa na saída, mas nada que tire a capacidade de realização e averiguação desse teste.

Sendo assim, ao passar pelo divisor resistivo, a tensão de saída do microfone deve ser em torno de 1,5V pelo fato dos dois resistores terem os mesmos valores. Na Figura 5.6, a tensão resultante foi de 1,33 V, fato comentado no parágrafo anterior.

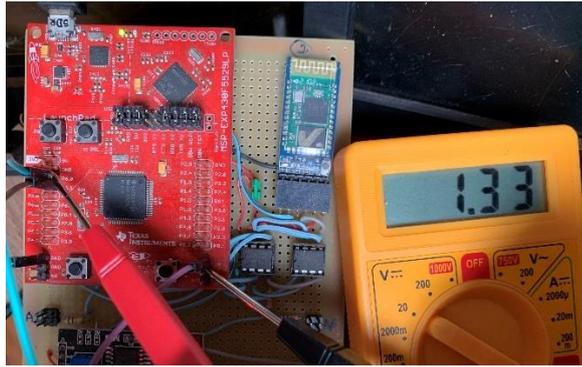


Figura 5.6: Tensão no pino P6.1 com referência à terra do MSP430

O conversor analógico digital está configurado para atuar com 12 bits, a faixa de valores será de 0 a 4095. Sendo 0 V referente a 0 e 3,3 V referente a 4095 para a conversão.

Com regra de 3 e utilizando o valor de 1,33 V como entrada, o valor resultante para a conversão será de 1650,4. Na Figura 5.7, vemos que o resultado entregue pelo ADC está muito próximo do que seria esperado. É claro, existe bastante ruído já que não se aplicou sinal aos microfones.

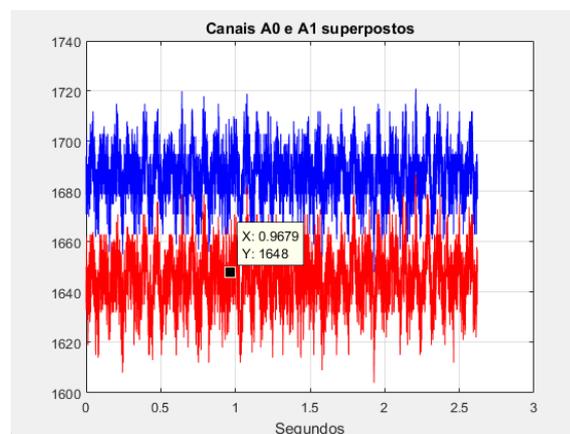


Figura 5.7: Sinal entregue pelo ADC quando os microfones estão em silêncio

5.2 TESTE DA CAPTAÇÃO DOS MICROFONES

Para o teste da captação dos microfones, foi usado um celular como fonte sonora rodando o aplicativo chamado *Tone Generator: Audio Sound Hz*⁶. Foi necessário a averiguação

⁶ Fonte: [Tone Generator: Audio Sound Hz on the App Store \(apple.com\)](https://www.apple.com/app-store/tone-generator/)

dessa fonte para saber se representava fielmente o sinal que estava configurado no aplicativo, e assim, por meio da funcionalidade do Matlab, gravou-se o sinal emitido pelo microfone do computador onde estava instalado o MATLAB.

```

Fs = 25000;
recObj = audiorecorder(Fs,8,1);
disp('Start speaking.')
recordblocking(recObj, 2);
disp('End of Recording. ');
y = getaudiodata(recObj);
figure(1);
plot(y);
title('Sinal Sonoro');

nfft = length(y);
nfft2 = 2^nextpow2(nfft);
ff = fft(y,nfft2);
fff=ff(1:nfft2/2);
xfft=Fs*(0:nfft2/2-1)/nfft2;
figure(2);
plot(xfft,abs(fff));
title('FFT');
xlabel('Frequencia Hz');

```

Figura 5.8: Código usado no MATLAB para testar o sinal gerado pelo aplicativo do celular

A configuração da fonte sonora utilizada foi um sinal senoidal com frequência fundamental de 1000 Hz e não foi necessário se preocupar com a amplitude, já que muda com a altura do som do celular e a distância entre celular e microfone. Pela Figura 5.8 é notável que a frequência de amostragem configurada para a captação do áudio pelo Matlab foi de 25000 Hz, valor este que será o mesmo valor para cada microfone que será usado no trabalho final.

A Figura 5.9 é resultado da captação realizada pelo Matlab e podemos confirmar que a envoltória da onda segue fielmente a de uma onda senoidal e que pela FFT do sinal, também representada pela Figura 5.9, de que a frequência fundamental selecionada no aplicativo foi efetivada de forma correta.

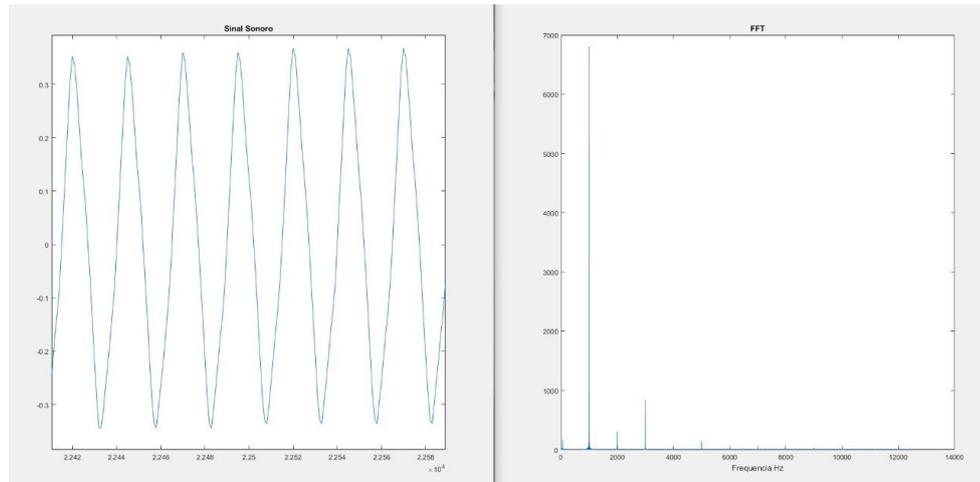


Figura 5.9: Resultado da análise do sinal gerado pelo aplicativo do celular. Se vê a forma senoidal é confiável e, na direita, que a FFT indica o valor correto de frequência.

Com o entendimento que a fonte sonora é uma fonte de qualidade boa, partimos para a análise do teste da captação dos microfones. O teste se baseou em emitir um sinal sonoro pelo celular (celular foi posicionado no centro da distância entre os dois microfones, com a finalidade de obter um sinal sem defasagem entre o canal A0 e canal A1), e usar o microfone para captar esse sinal e enviá-lo para o microcontrolador que faz a conversão analógico-digital e guarda essas informações na memória externa. Após isso, os dados que foram armazenados na memória externa são enviados para o Matlab para que seja feita a FFT do sinal e plotagem dos dados. A FFT está representada na Figura 5.11 e a plotagem dos dados representada na Figura 5.10.

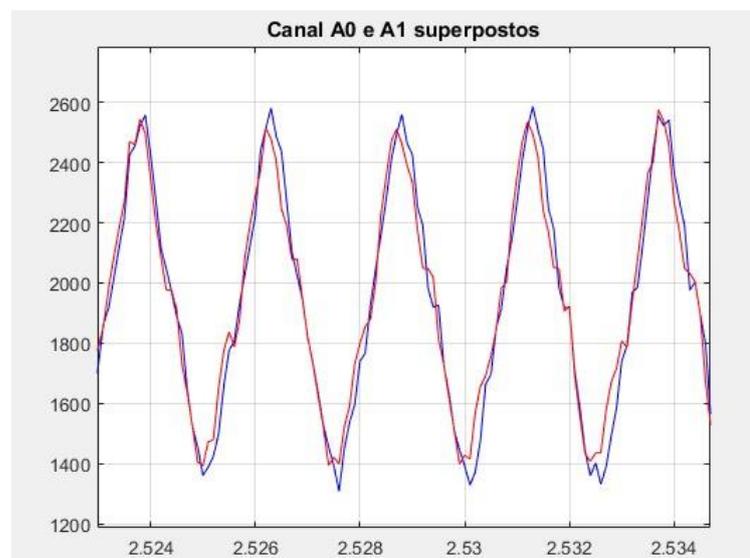


Figura 5.10: Resultado da conversão ADC, plotado pelo Matlab

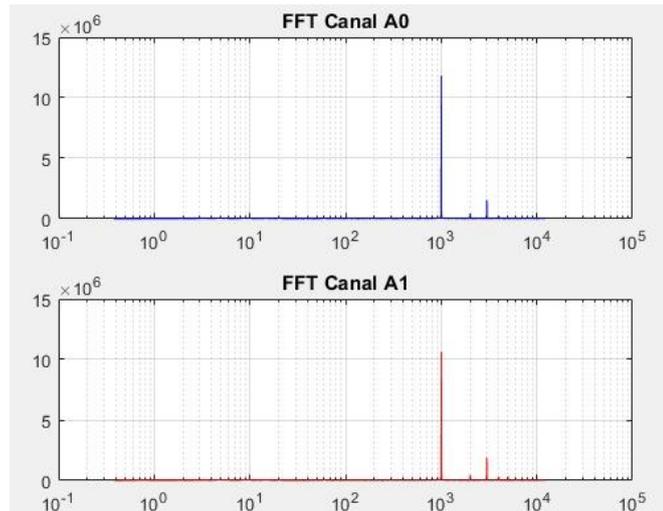


Figura 5.11: FFT do sinal digitalizado que foi entregue ao Matlab

Pelas Figura 5.10 e Figura 5.11, podemos confirmar a correta funcionabilidade dos sensores sonoros e da digitalização que foram capazes de captar o sinal recebido de forma adequada. Na Figura 5.10 nota-se alguma distorção na envoltória dos sinais. Este fato será discutido no item 5.2.1. Apesar disso, se pode dizer que foi recebida uma senoide muito próxima da esperada. É interessante notar que os dois sinais (azul e vermelho) estão praticamente coincidentes, já que o sinal foi gerado usando DOA igual a zero graus.

A Figura 5.11 apresenta a FFT de ambos os sinais e se pode ver que a quase totalidade da energia está concentrada em 1.000 Hz. Nota-se um pouco de energia em 2.000 Hz e em 3.000 Hz, provavelmente resultante da pequena distorção notada na figura anterior.

Ainda vale ressaltar que os dados enviados e plotados pelo Matlab não passaram por nenhum tipo de filtro que retirasse os ruídos presentes no mesmo.

5.2.1 Modificação do divisor resistivo

O MSP430 oferece um temporizador para controlar o tempo de amostragem do sinal analógico antes de iniciar a conversão. Ele é denominado de Sample/Hold Timer. O tempo de amostragem pode ser configurado e toma como base o relógio do ADC (ADC12CLK). Os valores possíveis vão de 4 até 1024 períodos do ADC12CLK, que no caso desta proposta está em 20 MHz. A conversão, de acordo com o manual demora 13 períodos do SMCLK.

Nos primeiros experimentos, foi usado para disparo um sinal quadrado de 50 kHz, período de 20µs, ou seja, 10µs para amostragem e 10µs para conversão. Como o relógio do ADC está em 20 MHz, são 200 períodos para amostragem e 200 períodos para conversão. Existe uma grande folga nesta temporização e então foram feitos vários testes com diferentes configurações de amostragem e conversão. Iniciou-se com tempo de amostragem acima de 4 ciclos, mas nenhuma das opções melhorou a qualidade da conversão, então foi definido que seriam 4 ciclos para a amostragem, valor esse que já era utilizado em trabalhos anteriores. A Figura 5.10 foi feita utilizando esses parâmetros.

O capacitor do amostrador/retentor (sample/holder) do conversor precisa de um tempo para se carregar de tal forma que a tensão de entrada menos a tensão do capacitor esteja abaixo a faixa de tensão equivalente a meio bit de ‘Vs’, mostrado na equação 5.1

$$V_s - V_c < \frac{1}{2} \text{ bit } V_s \quad (5.1)$$

O manual oferece um esquema bem prático para o cálculo do tempo de amostragem. Iniciamos com a Figura 5.12 que mostra um modelo com os valores de resistência e capacitância do circuito de entrada do ADC do MSP430. A resistência ‘Rs’, a resistência de saída do circuito do microfone. Essa resistência ‘Rs’, é dado pelos dois resistores do divisor resistivo da Figura 3.6 em paralelo, já que a saída do circuito do microfone é um amplificador operacional (buffer de ganho 1, como visto na Figura 3.5), tendo sua impedância de saída em malha aberta de 16 Ω e em malha fechada de 0,1 Ω.

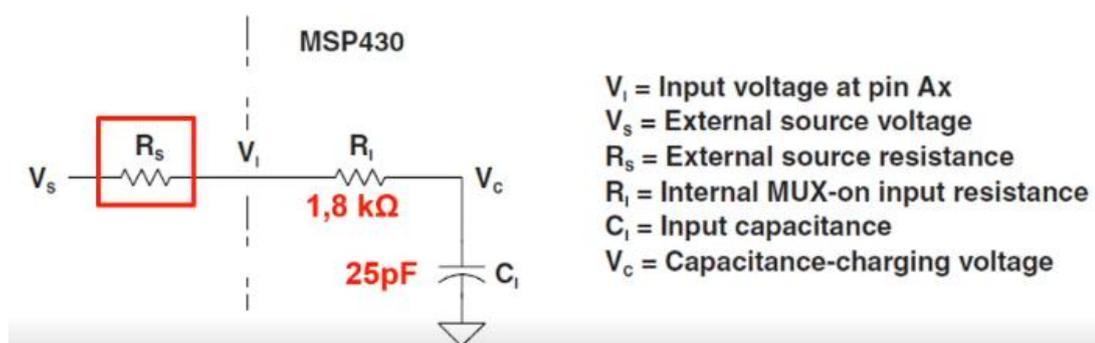


Figura 5.12: Circuito equivalente de entrada analógica

Para saber o tempo mínimo de amostragem, o manual⁷ do MSP430F5529 apresenta a equação 5.2 que depende de “Rs”.

$$T_{sample} > (R_s + R_i) \times \ln(2^{n+1}) \times C_i + 800 \text{ ns} \quad (5.2)$$

Substituindo “Rs” por 5 kΩ e “n” por 12 que é a resolução do ADC, temos o tempo mínimo de amostragem, necessitada nos trabalhos anteriores, de 2,33 μs. Com o Clock do microcontrolador em 20 MHz, tínhamos que seriam necessários 46,6 ciclos de Clock para a amostragem. Valor muito acima do selecionado que foi de escolhido de 4 ciclos. Porém, nada de apreciável foi notado com este intervalo de amostragem.

A modificação feita foi trocar os resistores de 10 kΩ por resistores de 1 kΩ, com o intuito de diminuir a impedância de saída do amplificador. Com a mudança dos valores dos resistores que são usados para fazer o divisor resistivo, temos que o novo valor de “Rs” é de 500 Ω. Então, agora, são necessários 26,2 ciclos de Clock, pois o tempo mínimo de amostragem passou para 1,31 μs.

Ainda assim, 4 ciclos estão bem abaixo dos 26,2 ciclos calculados, mas está bem mais perto do que 46,6 ciclos, representando uma redução de 43,78%.

A Figura 5.13 representa o sinal captado nas mesmas condições realizadas na Figura 5.10. Como podemos observar, agora o sinal está representando a sua envoltória de forma mais adequada, com menor presença de ruído em sua composição. Os picos que antes tinham muitas imperfeições, agora estão quase que perfeitos. Vale ressaltar que o sinal está puro, sem nenhum tratamento via MSP430, apenas como o filtro passa-baixas analógico (anti-aliasing) presente no pré-amplificador microfone.

⁷ Fonte: [User Guide - Google Drive](#)

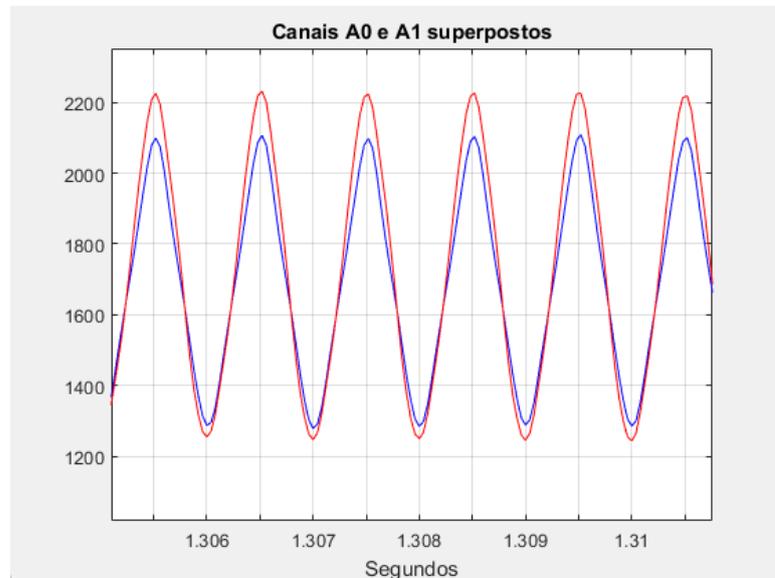


Figura 5.13: Sinal puro recebido entregue pelo ADC com o emprego de um divisor resistivo com resistores de $1\text{ k}\Omega$

5.3 TESTE DO ENVIO DOS DADOS PELO HC-05

Para testar a veracidade da programação que envia os dados presentes na memória externa ao MATLAB foi feito um teste preenchendo toda a memória externa com dados conhecidos e, em seguida, enviando estes dados ao computador por meio do HC-05 e plotando os dados pelo MATLAB.

Nesse teste foi criada uma função para o MSP que escreve uma onda quadrada na memória externa. Nessa função, alguns parâmetros são responsáveis por definir como será esta onda. No caso desse teste, a amplitude foi definida como 4 e o período como 8192 posições na memória, ou seja, a cada 8192 números percorridos pela memória, ocorre um período do sinal.

Na Figura 5.14, podemos observar o sinal sintetizado nas memórias externas pelo MSP430 que foram enviados pelo HC-05. Nota-se que essa parte do programa funciona de forma adequada.

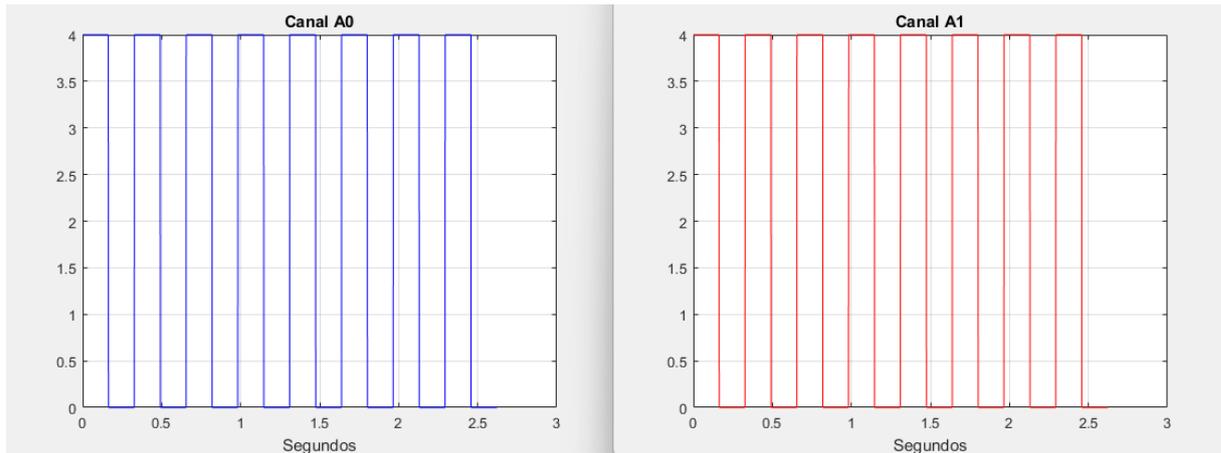


Figura 5.14: Sinal quadrado recebido e plotado pelo MATLAB

5.4 TESTE DO FILTRO MÉDIA MÓVEL

Para averiguar se o filtro de média móvel está com a programação correta, foi usado a mesma função do item - 5.3 (Teste do envio dos dados pelo HC-05) que escreve uma onda quadrada nas memórias externas. Dessa vez, apenas o período da onda quadrada foi modificado. Assim, a amplitude continuou 4, porém o período foi modificado para 8 posições na memória.

O filtro de média móvel foi programado para ser de ordem 4, por isso a escolha do período da onda quadrada deve ser 8, pois, desse modo, o sinal resultante da passagem pelo filtro de média móvel terá que ser um sinal triangular.

Quadro 5.1: Primeiros 16 números do sinal quadrado da memória 0 e 1

Posição do vetor	Valor
1	4
2	4
3	4
4	4
5	0
6	0
7	0
8	0
9	4

10	4
11	4
12	4
13	0
14	0
15	0
16	0

O Quadro 5.1 representa as primeiras 16 posições do vetor que foi recebido pelo MATLAB do sinal escrito nas memórias antes do filtro, esse sinal é o que está representado na Figura 5.15, mas com o período muito reduzido. Esse vetor tem 65536 posições, mas só foram apresentadas as 16 primeiras posições, pois esse padrão segue por todo resto das memórias.

Quadro 5.2: Primeiros 16 números do sinal filtrado da memória 0 e 1

Posição do vetor	Valor
1	1
2	2
3	3
4	4
5	3
6	2
7	1
8	0
9	1
10	2
11	3
12	4
13	3
14	2
15	1
16	0

O Quadro 5.2 representa os dados presentes na memória 0 e memória 1, após a passagem do filtro média móvel pelo sinal presente nas duas memórias. Novamente, esse vetor tem 65536 posições, mas só foi colocado as 16 primeiras posições, pois esse padrão segue por todo resto das memórias.

Pelo teste é identificado que o filtro de média móvel atua corretamente nas duas memórias externas. A Figura 5.15, representa graficamente o sinal quadrado após a passagem pelo filtro média móvel, resultando num sinal triangular. Como os dois sinais são iguais, eles estão sobrepostos.

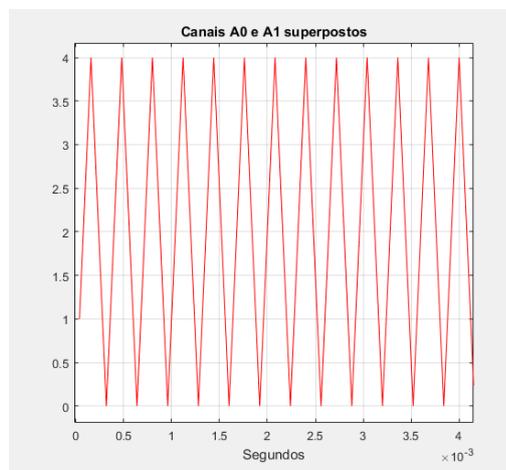


Figura 5.15: Sinal A0 e A1 filtrado pelo média móvel

5.5 TESTE IDENTIFICAÇÃO DE UMA DA REGIÃO ADEQUADA PARA ESTIMATIVA DO DOA

Foi criada uma função que identifica regiões adequadas para realizar a estimativa e posteriormente realizado testes que comprovam a sua funcionalidade. Uma região é adequada para a estimação possui bastante “energia”. Em termos práticos foi usado um limiar, assim, “energia” acima deste limiar indica uma boa região para estimação.

Essa função que calcula a “energia” se baseia no desvio médio dos valores contidos na memória 0 e memória 1. A função trabalha calculando a “energia” de blocos de 128 valores inteiros. Se os blocos de ambas as memórias indicarem “energia” acima do limiar especificado,

então este bloco de 128 inteiros é adequado para a estimação do DOA. A “energia” é calculada com base no desvio médio, como mostrado abaixo.

$$\text{Desvio médio} = \frac{|X_1 - \bar{x}| + |X_2 - \bar{x}| + \dots + |X_n - \bar{x}|}{n} \quad (5.2)$$

Sendo, ‘ \bar{x} ’ igual a média aritmética da amostra e ‘n’ a quantidade de números da amostra.

Para testar este detector, foi realizada uma aquisição especial na qual a fonte sonora foi ligada e desligada algumas vezes. O ponto de partida para o detector iniciar foi definido com o endereço 2048 das memórias (mesmo ponto inicial para fazer as futuras estimativas) que representa começar a partir de 1024 números armazenados na memória. Foi programado para que fosse entregue 8 regiões boas para estimativa a partir desse endereço posição de 2048.

A Figura 5.16 mostra a eficiência dessa abordagem do desvio médio, sendo que a linha tracejada na vertical representa a posição de partida para fazer a estimativa. Como é observado, a região que parou de receber o sinal da fonte sonora não foi utilizado para a estimativa.

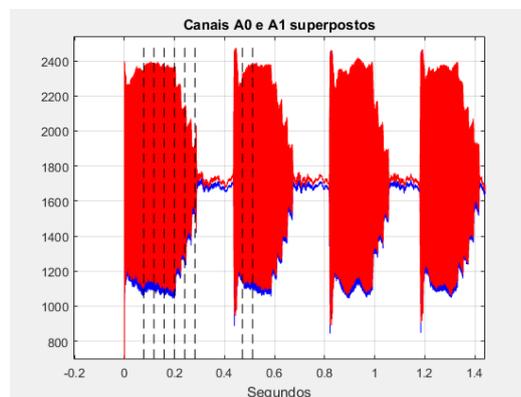


Figura 5.16: Faixa adequada para realizar a estimativa

5.6 TESTE DA ESTIMATIVA DO ATRASO

Nesse teste da estimativa do atraso, a principal pergunta é se a função que estima o atraso de um sinal em relação ao outro está correto, e se, a direção que o sinal está chegando

(lado direito ou lado esquerdo) procede, sendo sinal vindo pela esquerda retorna ângulos negativos e sinal vindo pela direita retorna ângulos positivos. Essa referência é tomada em relação ao microfone apontando para o locutor.

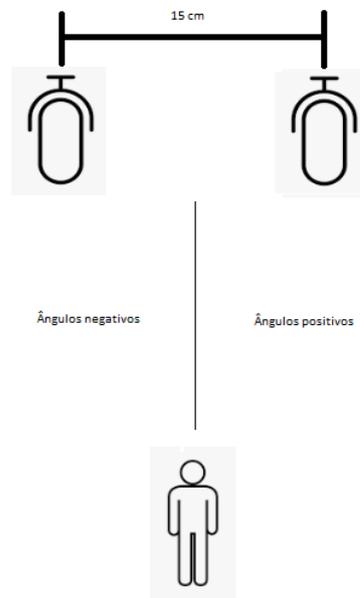


Figura 5.17: Referência utilizada para os ângulos, definindo então lados direito e esquerdo.

Para esse teste, foi criada uma função que sintetiza um sinal senoidal nas duas memórias com a mesma frequência que será utilizada na fonte sonora de 1 kHz. Os outros parâmetros foram: amplitude igual 1000, ponto médio igual a 2000, frequência de amostragem igual a 25000 kHz.

Não foram realizados testes para defasagem de 0 posições nos dois canais e acima de 10 posições para os canais A0 e A1, pois defasagens acima de 66 graus (situação de 15 cm entre microfones e taxa de amostragem de 25 kHz) apresentaram resultados ruins, por isso, se considerou que a faixa útil deveria ficar entre -66 e + 66. A defasagem de 0 posições para os dois canais entra no modo de resposta errada, pois não existe defasagem para o programa detectar. É preciso encontrar uma outra solução para estimar tal caso.

Para a primeira parte do teste foi configurado para o sinal A0 ficar fixo e o sinal A1 variar sua defasagem de 10 até 1 posições à frente do sinal A0. Desse modo, terá que resultar em ângulos negativos. Da Figura 5.18 a Figura 5.27, apresentam os sinais no tempo e permite visualizar a defasagem inserida. O Quadro 5.3 apresenta o resultado do estimador de atrasos para as diversas situações e, como se pode ver, ele sempre acertou.

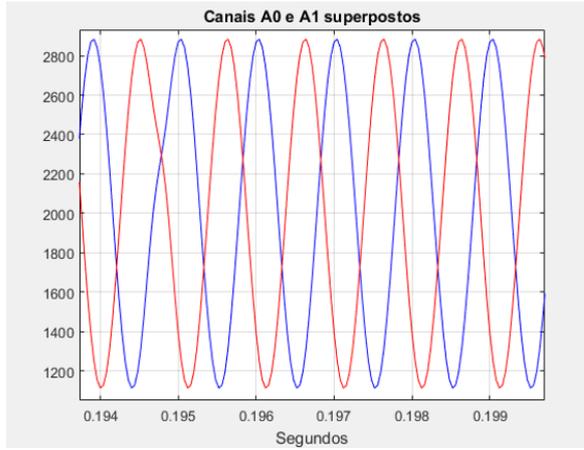


Figura 5.18: Defasagem de 10 posições para A1

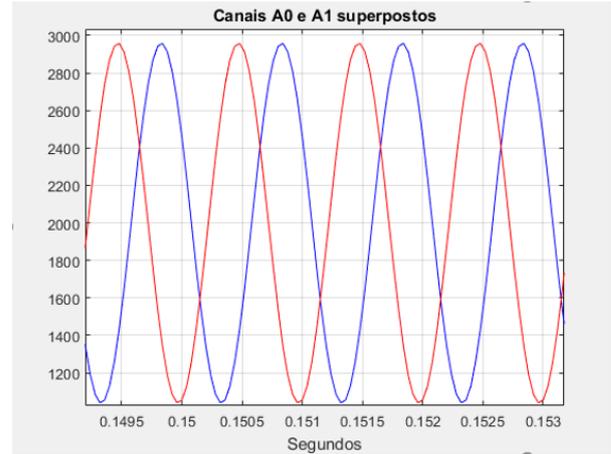


Figura 5.19: Defasagem de 9 posições para A1

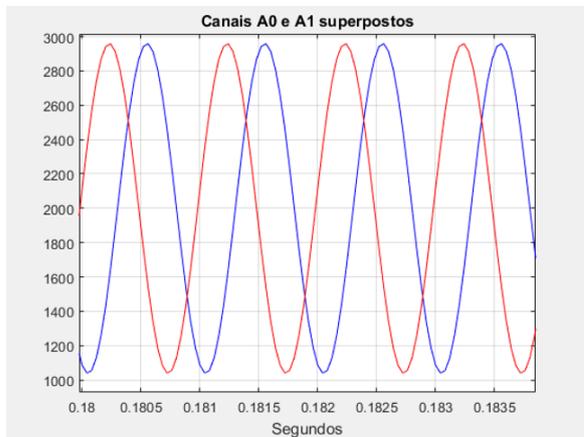


Figura 5.20: Defasagem de 8 posições para A1

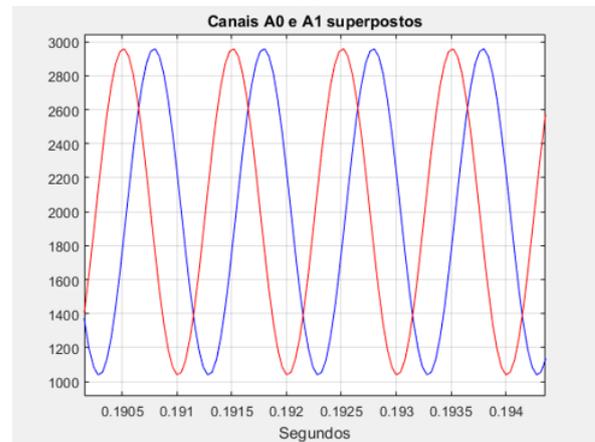


Figura 5.21: Defasagem de 7 posições para A1

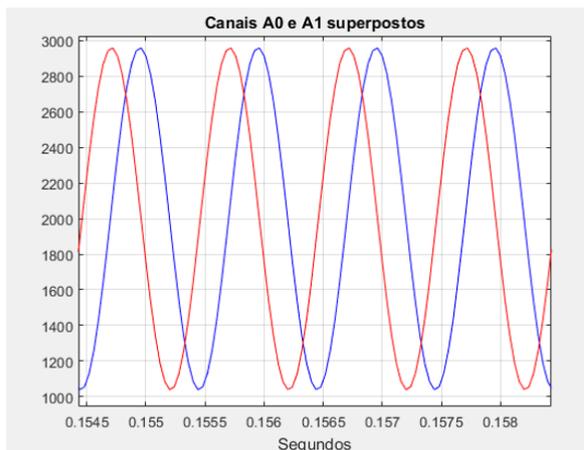


Figura 5.22: Defasagem de 6 posições para A1

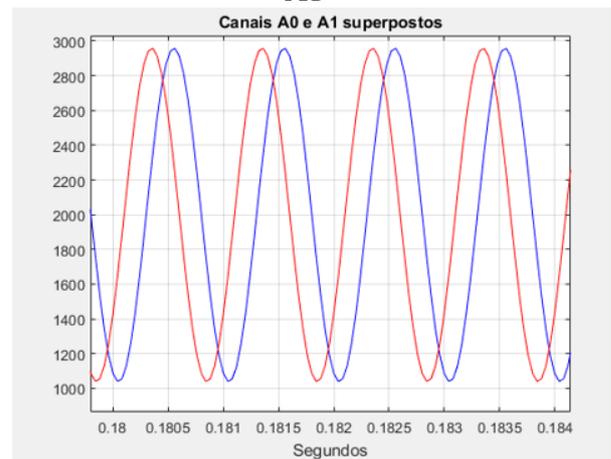


Figura 5.23: Defasagem de 5 posições para A1

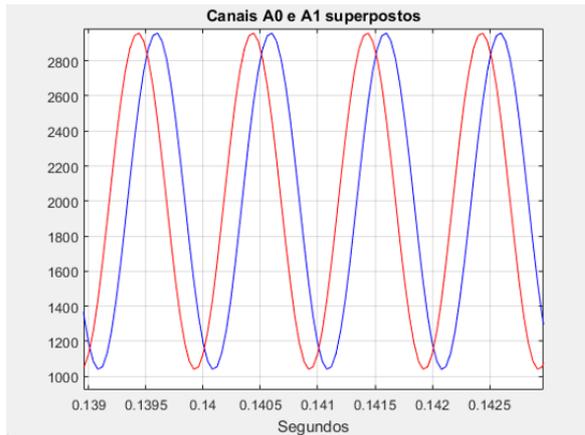


Figura 5.24: Defasagem de 4 posições para A1

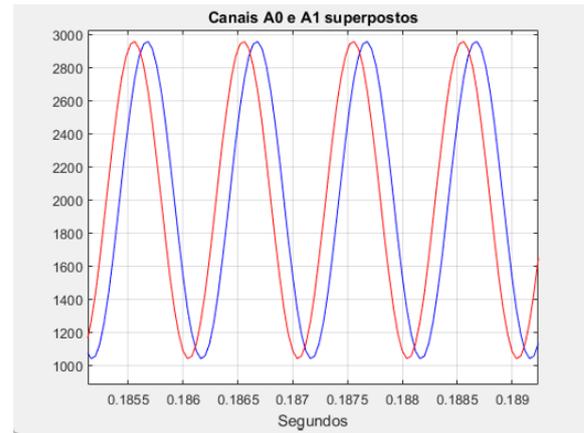


Figura 5.25: Defasagem de 3 posições para A1

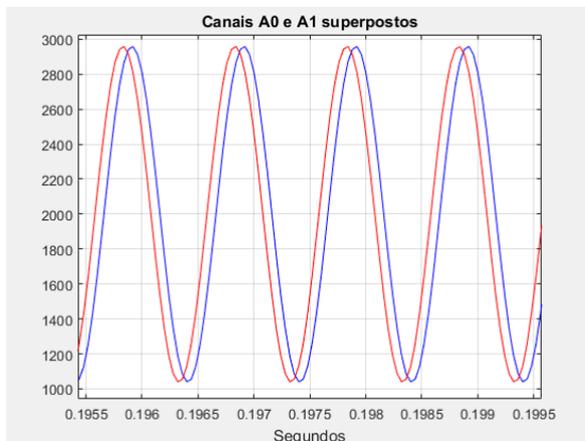


Figura 5.26: Defasagem de 2 posições para A1

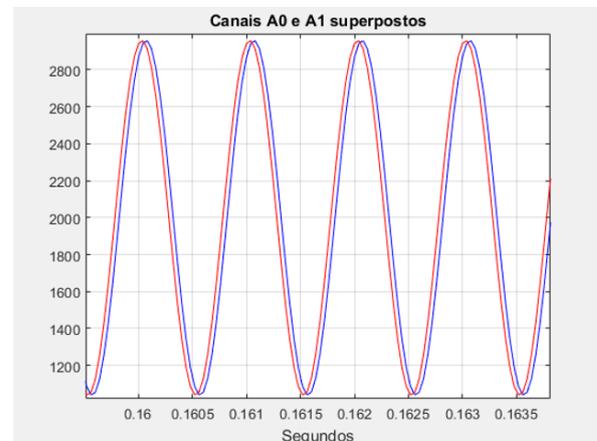


Figura 5.27: Defasagem de 1 posições para A1

Quadro 5.3: Ângulos resultantes para teste da estimativa A1 adiantado de A0

Figura	Ângulo resultante pelo código
Figura 5.18	-66°
Figura 5.19	-55°
Figura 5.20	-47°
Figura 5.21	-40°
Figura 5.22	-33°
Figura 5.23	-27°
Figura 5.24	-21°
Figura 5.25	-16°
Figura 5.26	-10°
Figura 5.27	-5°

Para a segunda parte do teste foi configurado para o sinal A1 ficar fixo e o sinal A0 variar sua defasagem de 10 até 1 posições à frente do sinal A1. Desse modo, terá que resultar em ângulos positivos.

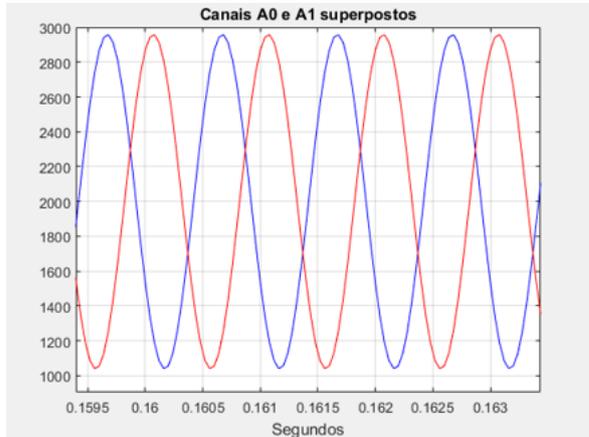


Figura 5.28: Defasagem de 10 posições para A0

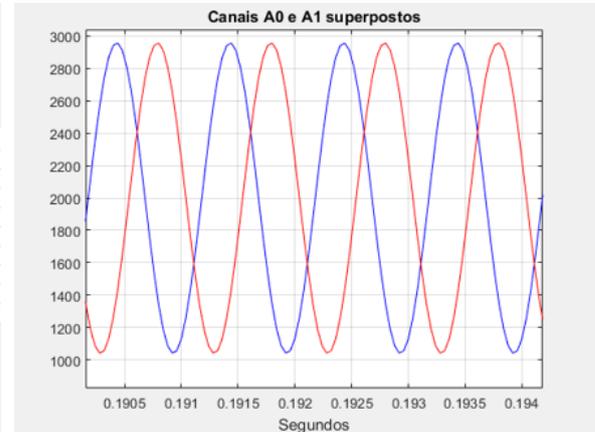


Figura 5.29: Defasagem de 9 posições para A0

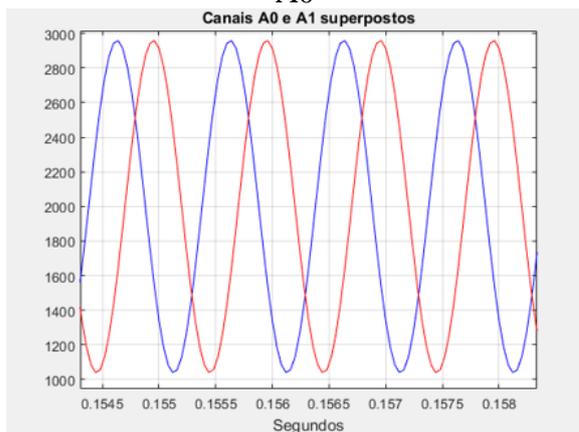


Figura 5.30: Defasagem de 8 posições para A0

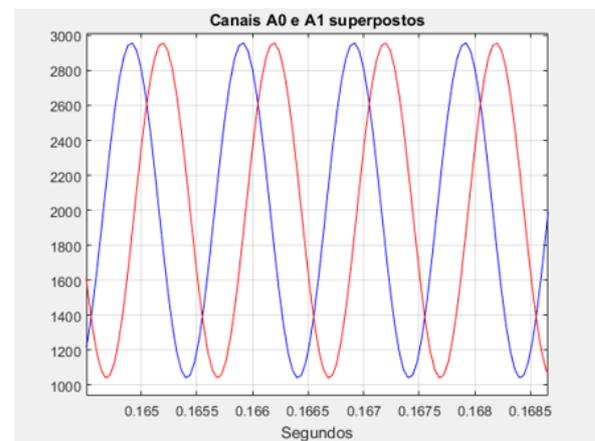


Figura 5.31: Defasagem de 7 posições para A0

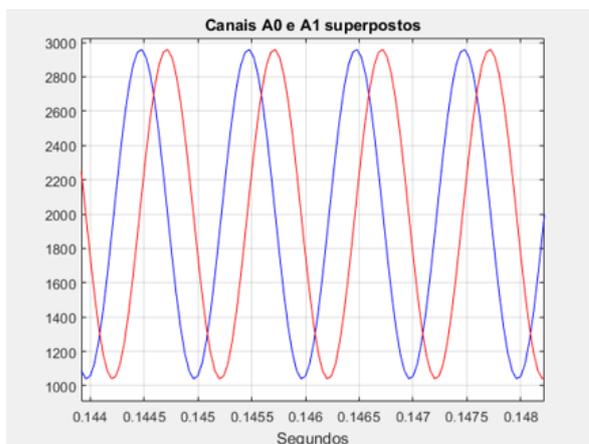


Figura 5.32: Defasagem de 6 posições para A0

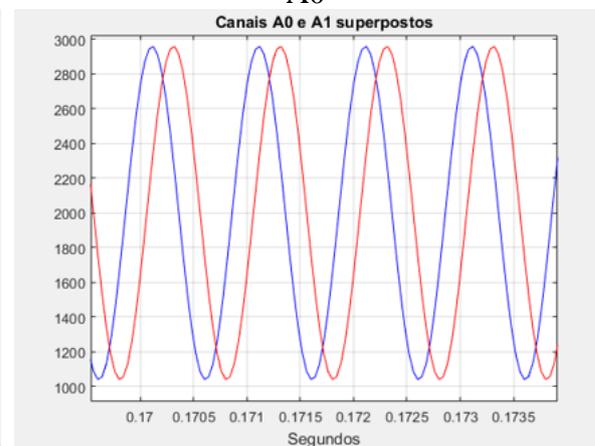


Figura 5.33: Defasagem de 5 posições para A0

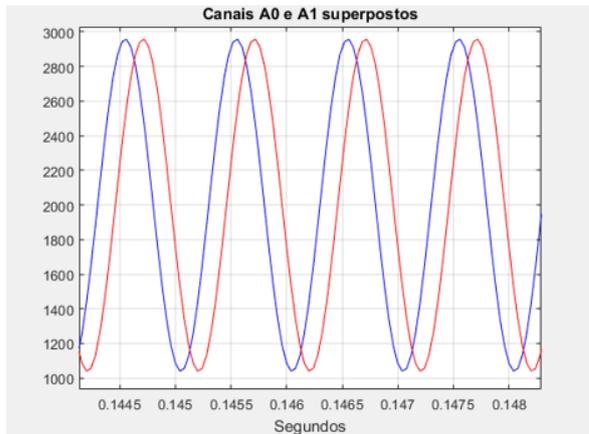


Figura 5.34: Defasagem de 4 posições para A0

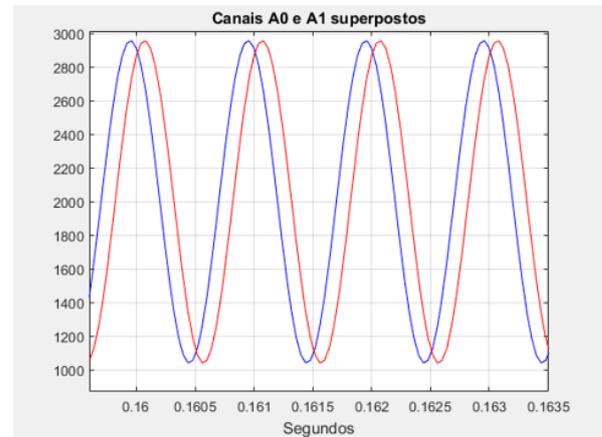


Figura 5.35: Defasagem de 3 posições para A0

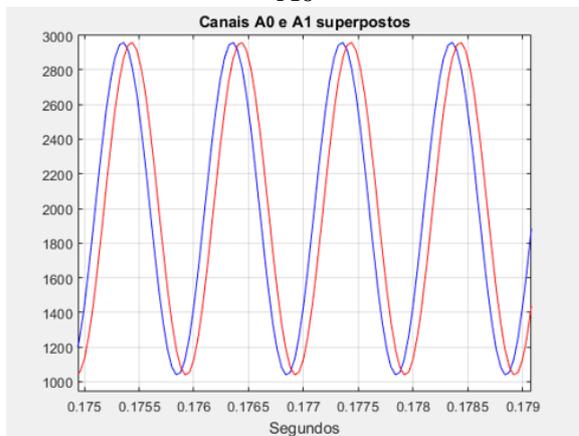


Figura 5.36: Defasagem de 2 posições para A0

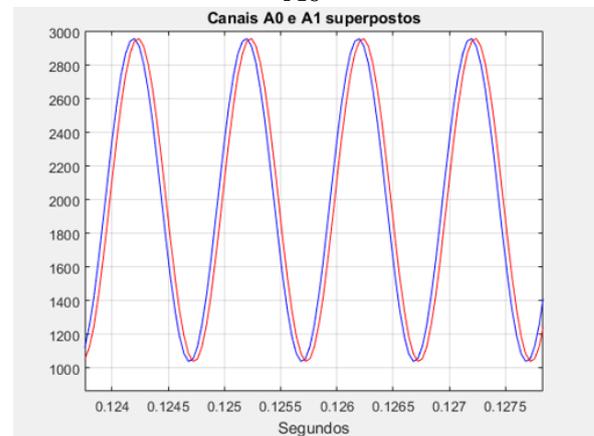


Figura 5.37: Defasagem de 1 posição para A0

Quadro 5.4: Ângulos resultantes para teste da estimativa A0 adiantado de A1

Figura	Ângulo resultante pelo código
Figura 5.28	66°
Figura 5.29	55°
Figura 5.30	47°
Figura 5.31	40°
Figura 5.32	33°
Figura 5.33	27°
Figura 5.34	21°
Figura 5.35	16°
Figura 5.36	10°
Figura 5.37	5°

De todas essas 20 imagens anteriores, o sinal vermelho representa o canal A0 que se fosse um sinal captado pelo microfone seria o microfone da direita e o sinal azul representa o canal A1 que se fosse um sinal captado pelo microfone seria o microfone da esquerda.

Pelas análises dessas 20 imagens e pelas respostas da estimativas do sinal de chegada referentes a cada figura, podemos afirmar que o algoritmo empregado na MSP430F5529 para estimativa da direção de chegada está funcionando da maneira correta.

6 ENSAIOS DA DIREÇÃO DE CHEGADA DO SINAL SONORO

Nessa parte final do projeto foram feitos ensaios para estimar a direção de chegada do sinal, quando todas as funções do programa⁸ atuam de forma conjunta. a direção de chegada dos sinais sonoros e se esses ângulos resultantes estão de acordo com o esperado.

Os ensaios abrangeram ângulos de 5° a 60° tanto para o lado direito (referência positiva) quanto para o lado esquerdo (referência negativa). A faixa útil de 5° a 60° foi varrida em passos de 5°. Foram feitas 30 estimativas para cada ângulo. Em alguns ensaios tiveram respostas do ângulo de chegada de 99° ou -99°, esses valores foram configurados para representar erro na estimativa.

Vale lembrar que o resultados das estimativas dos ângulos estão usando os dados do Quadro 4.3.

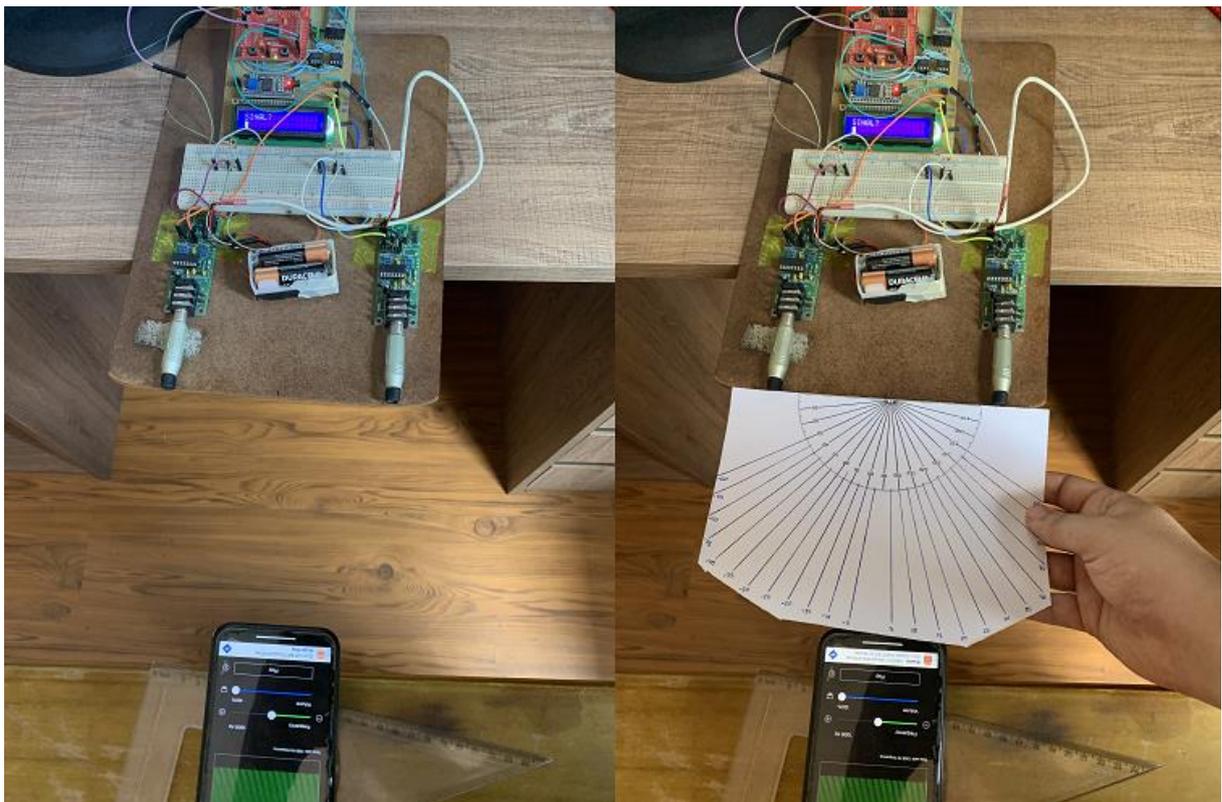


Figura 6.1: Hardware do DOA/fonte sonora/transferidor utilizado nos ensaios

⁸ Fonte: [DOA_FINAL - Google Drive](#)

6.1 ENSAIOS PARA ÂNGULOS DE REFERÊNCIA POSITIVA

Nesse teste a fonte sonora foi posicionada na parte direita da Figura 5.17. As 4 figuras a seguir representam os resultados dos ensaios para ângulos positivos, ou seja, o microfone do lado direito recebe o sinal primeiro que o microfone do lado esquerdo. As diversas figuras apresentam resultados indicados por linhas coloridas, uma para cada direção de chegada. Para cada ‘tentativa’ são realizadas 64 estimativas, feitas a partir de 64 regiões adequadas (com “energia” suficiente) identificadas pelo algoritmo já estudado. O sistema usa esses 64 valores para determinar a estimativa final (esse valor é o que está plotado nos gráficos), usando a moda. Então, cada ‘tentativa’ representa uma aquisição completa, resultando como resposta a moda de 64 estimativa de regiões adequadas do sinal.

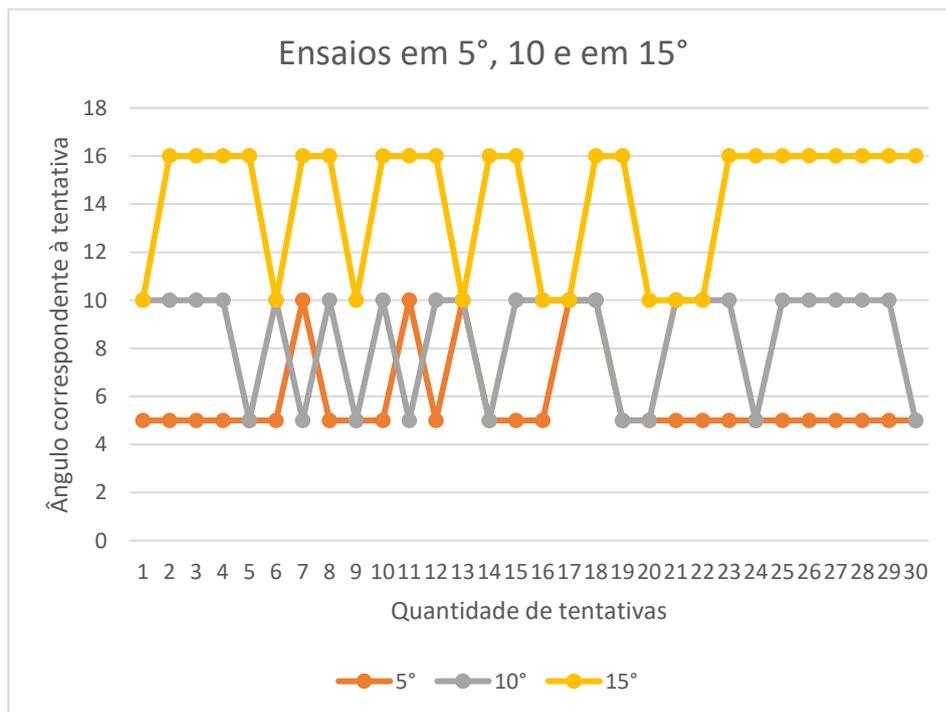


Figura 6.2: Resultado dos ensaios em 5°, 10° e em 15°

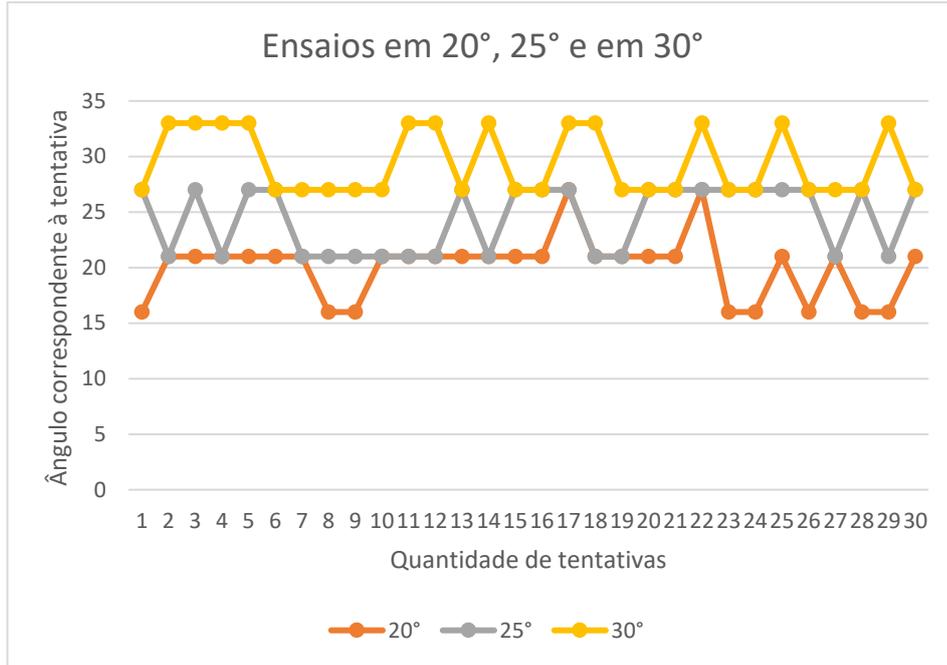


Figura 6.3: Resultado dos ensaios em 20°, 25° e em 30°

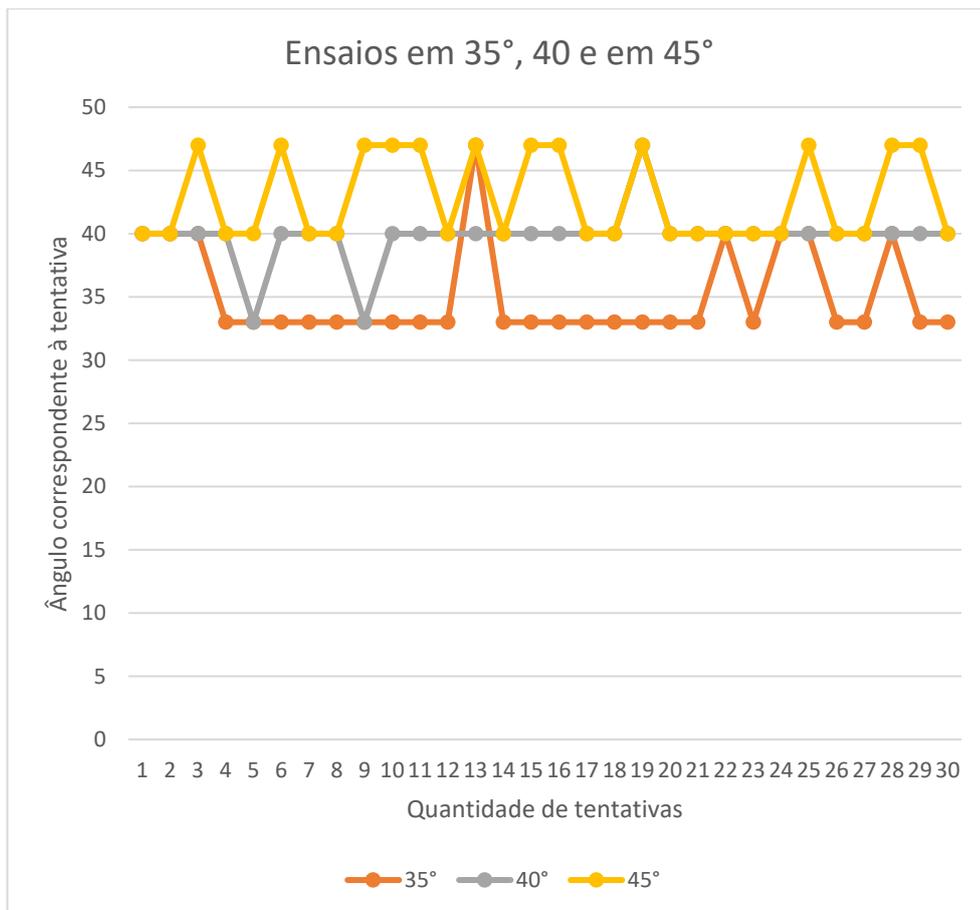


Figura 6.4: Resultado dos ensaios em 35°, 40° e em 45°

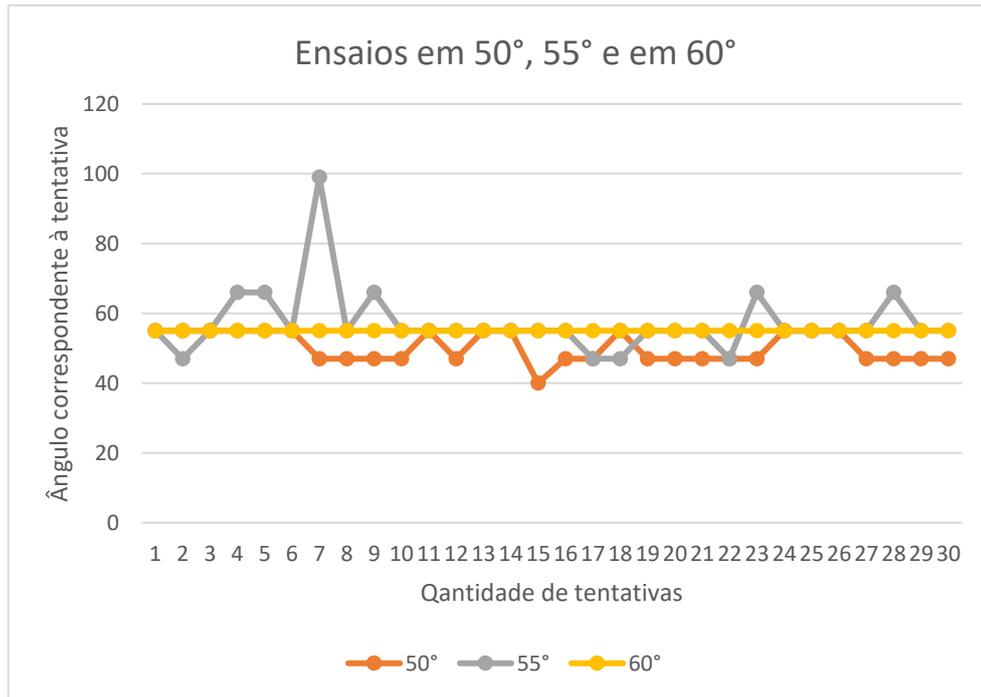


Figura 6.5: Resultado dos ensaios em 50°, 55° e em 60°

6.2 ENSAIOS PARA ÂNGULOS DE REFERÊNCIA NEGATIVA

Nesse teste a fonte sonora foi posicionada na parte esquerda, segundo referência indicada na Figura 5.17. As 4 figuras a seguir representam os resultados dos ensaios para ângulos negativos, ou seja, o microfone do lado esquerdo recebe o sinal primeiro que o microfone do lado direito.

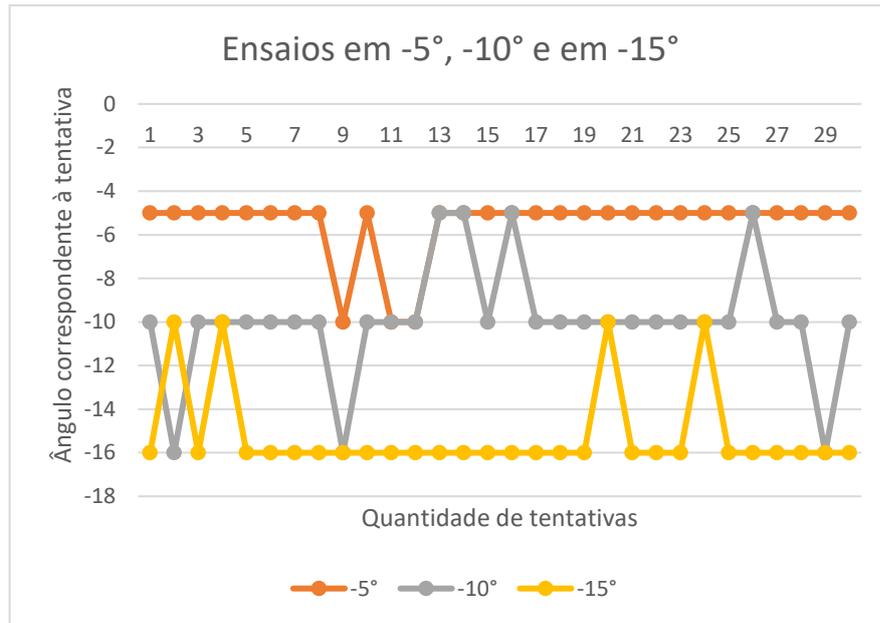


Figura 6.6: Resultado dos ensaios em -5°, -10° e em -15°

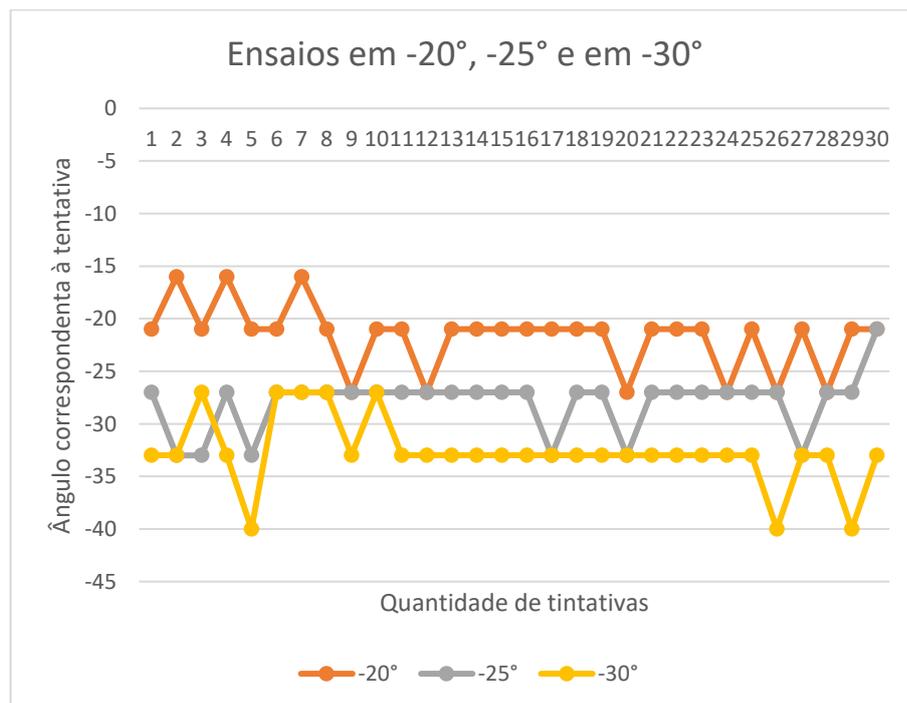


Figura 6.7: Resultado dos ensaios em -20°, -25° e em -30°

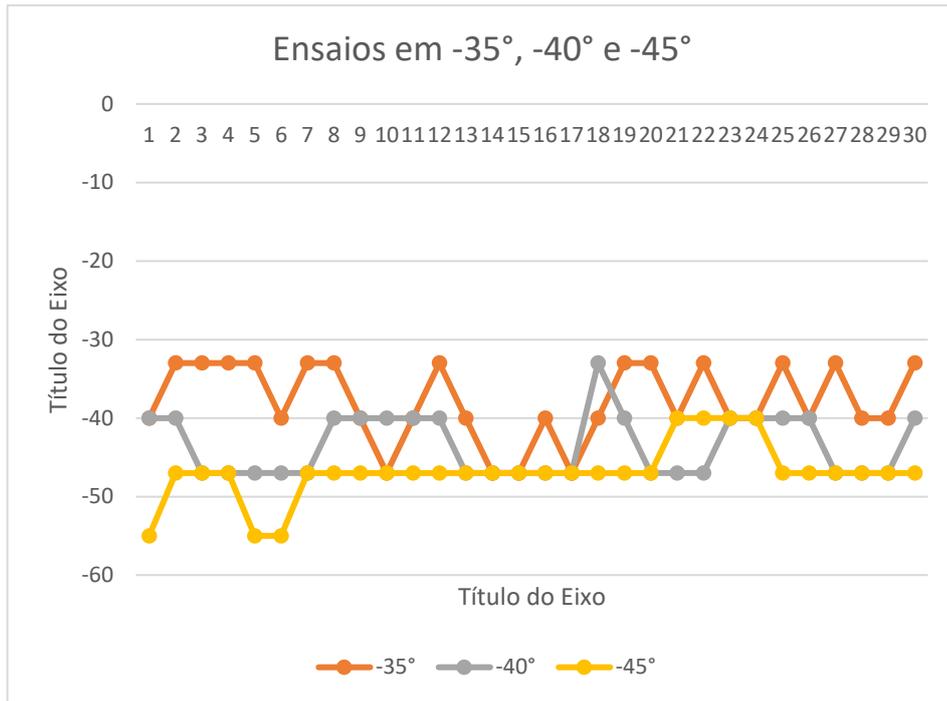


Figura 6.8: Resultado dos ensaios em -35°, -40° e em -45°

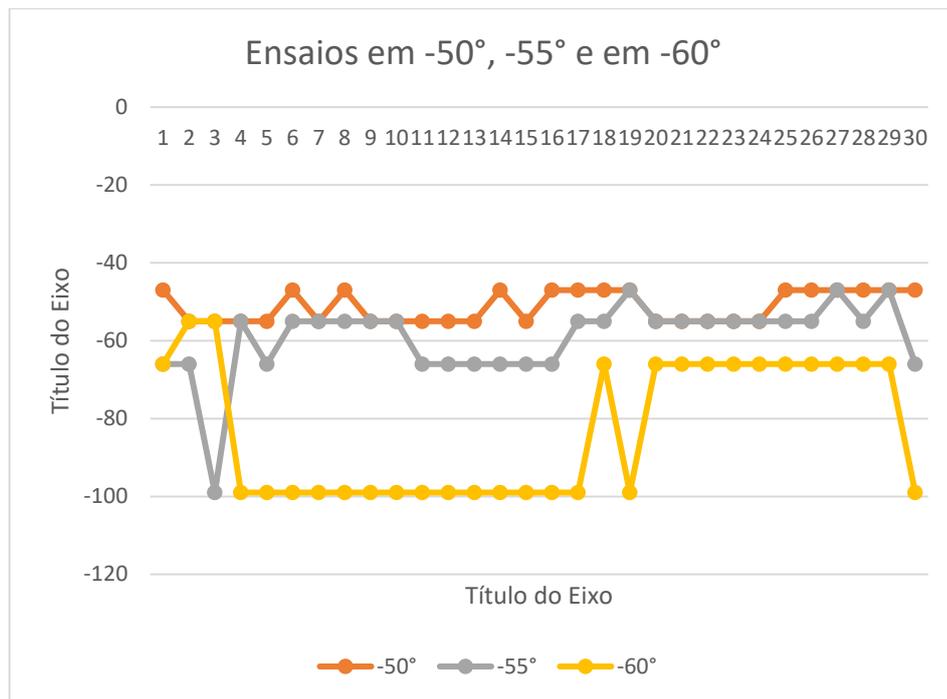


Figura 6.9: Resultado dos ensaios em -50°, -55° e em -60°

6.3 CONCEITOS APRENDIDOS DURANTE OS ENSAIOS

Para que o programa retorne estimativas de DOA aproveitáveis, é necessário que nenhuma interferência ocorra ao sinal que é emitido da fonte sonora. Por exemplo, no início dos ensaios tanto o MSP430 com os microfones e a fonte sonora colocados em cima de uma mesa, levava a vários resultados totalmente que errados, que não faziam sentido. A realização de mais alguns ensaios indicou que isso acontecia por causa da interferência nas ondas, seja sela construtiva ou destrutiva, gerado principalmente por reflexão sobre a mesa e ecos.

Dito isso, foi modificado a forma como foi feita os ensaios. A fonte sonora ficou em uma mesa e o MSP430 com os microfones ficaram em uma outra mesa, sem ter nada por perto entre os dois. Essas mesas estavam niveladas para evitar qualquer tipo de interferência que o sinal poderia sofrer.

Os ensaios de ângulo $-55^{\circ}/55^{\circ}$ e $-60^{\circ}/60^{\circ}$ foram difíceis de serem executados. Tendo que a posição que a fonte sonora fica é muito perto de um microfone e muito longe do outro microfone, nisso tive que ir testando intensidade do sinal sonoro e distância da fonte sonora ao centro dos microfones. Para um sinal de intensidade sonora alta e nessa condição de posição, o microfone mais perto recebia o sinal e fazia com que seu o amplificador, entrasse na região de saturação, prejudicando na conversão analógico-digital. Para uma intensidade do sinal sonora razoável, o microfone mais perto da fonte sonora funciona adequadamente, mas o microfone mais distante da fonte sonora, não entregava um sinal adequado com uma envoltória de onda senoidal. Diante disso, tive que encontrar uma relação entre distância e intensidade sonora para que o sistema entregasse um valor de ângulo mais correto.

6.4 AVALIAÇÃO DOS RESULTADOS OBTIDOS

Para essa avaliação foram feitas duas análises, a primeira para a resposta o mais perto do valor correto e a segunda para um valor de resposta aceitando variações de 1 de defasagem para cima ou para baixo, pelo fato dito no parágrafo anterior. Por exemplo, para o caso de 30 graus, aceita-se como correto os valores 27 e 33 graus. Isto se justifica porque o sistema de estimação, por ser discreto, não é capaz de estimar a direção de 30 graus, e por isso oscila entre

os dois valores mais próximos que são 27 e 33 graus. Em cada análise, será dividido em ângulos que tive ou não dificuldade para adquirir, sendo que os que eu tive dificuldade foram os casos acima de 55°.

Ao todo, foram feitas 720 aquisições. Para os ângulos que não tive dificuldade, foram 600 aquisições e os ângulos com dificuldade foram 120 aquisições. Todos os dados estão salvos em um arquivo Excel⁹.

Quadro 6.1: Primeira análise

Primeira Análise		Acertos
Faixa de 5° a 50°	Valor correto	76,83333333%
Faixa de 5° a 50°, com adicional da faixa de 55 a 60°	Valor correto	69,02777778%

Quadro 6.2: Segunda análise

Segunda Análise		Acertos
Faixa de 5° a 50°	Aceitando variação de 1 de defasagem	99,6666667%
Faixa de 5° a 50°, com adicional da faixa de 55 a 60°	Aceitando variação de 1 de defasagem	97,2222222%

Pelo Quadro 6.1, é mostrado a primeira análise que é uma pouco mais severa. O programa acertou quase 80% dos ensaios para casos na faixa de 5° a 50° e acertou quase 70% quando adicionas na conta da faixa de 5° a 50°, as tentativas dos ensaios em -55°/55° e -60°/60°.

Pelo Quadro 6.2, é mostrado a segunda análise que é um pouco mais abrangente, pois aceita como resposta correta valores de ângulos com defasagem um acima/abaixo do correto. O programa acertou 99,6% dos ensaios para casos na faixa de 5° a 50° e acertou 97.22% quando adicionas na conta da faixa de 5° a 50°, as tentativas dos ensaios em -55°/55° e -60°/60°.

É importante explicar como de fato acontece a conversão analógico digital dos sinais captados pelos microfones. Como foi dito em 4.2, a taxa de conversão é de 50 kHz para o ADC do MSP430. Pela forma como funciona o ADC, a conversão de cada microfone não acontece

⁹ Fonte: [Resultado Final.xlsx - Planilhas Google](#)

ao mesmo tempo, mas sim de modo intercalado, um seguido do outro, por isso a taxa de amostragem de cada microfone fica a metade da taxa de conversão do conversor. Assim, o tempo entre uma conversão de um microfone para o outro é de 0,02 ms, o que já é um atraso introduzido pela amostragem, mas que tem sido ignorado sem grandes prejuízos.

7 CONCLUSÕES E PROPOSTAS PARA FUTURAS ANÁLISES

Esse projeto teve como objetivo a verificação de toda a programação feita no MSP430F5529. Partindo da simplificação do sinal que seria captado pelos sensores sonoros, que antes era um assovio e nesse projeto foi usado um sinal senoidal de 1 kHz. Os testes de todas as partes principais foram realizados em separado, tendo algumas dessas partes aprimoradas e/ou corrigidas, garantido assim que todo o firmware está funcional.

Saindo da parte de programação e tentando entender mais a parte física de todo o projeto criado, foi entendido a relação tempo do amostrador do ADC x impedância de saída dos microfones. Sendo que os dois devem ser o menor possível para melhor digitalização do sinal. Dentro desse assunto, como o tempo de amostragem já foi determinado para ser o menor possível que o microcontrolador pode ser configurado, trabalhos futuros deverão entender qual a maior corrente possível que pode fluir por essa resistência de saída (com maior diminuição da resistência de saída), a fim de melhorar ainda mais a relação dita anteriormente.

O método da correlação para fazer a estimativa do ângulo de chegada dos sinais sonoros, é muito sensível ao formato da onda captada pelos microfones e as interferências construtivas e/ou destrutivas no sinal sonoro ocasionam erros na estimativa do ângulo de chegada, por isso com a mudança nos resistores responsáveis pelo divisor resistivo e não deixando nada perto entre a fonte sonora e os microfones, garantiu uma porcentagem maior de respostas corretas. Ainda avaliando a porcentagem e pensando em trabalhos futuros, é necessário pensar numa forma de os dois microfones captarem o sinal sonoro ao mesmo tempo, visto que isso iria aumentar ainda mais a porcentagem de acerto, provavelmente, saindo de 76% para 99% de acerto.

Outro ponto que tenderia a melhorar a porcentagem de acertos é aumentar a taxa de amostragem de 25 kHz para 50 kHz em cada microfone, resultando numa taxa de 100 kHz para o conversor analógico-digital do MSP430. Isto vai dobrar a resolução do estimador e assim, mais se aproximar do valor correto do DOA. Essa função está praticamente pronta dentro da programação e para utilizá-la basta alterar uma única constante que todo o sistema se adapta. Vale ressaltar que com essa mudança o tempo do sinal sonoro que poderá ser armazenado nas memórias externas cairá pela metade.

As baterias que são usadas para fornecer energia aos microfones também podem ser objeto de estudos, pois o método que é atualmente usado é utilizar 4 pilhas comuns de tamanho

AA que ao serem usadas vão perdendo aos poucos a sua eficiência, ou seja, diminuem o valor da tensão que fornecem ao longo do tempo. E isso, diminui a faixa máxima de excursão para a conversão do sinal analógico, além de, aumentar as chances de entrar na região de saturação nos amplificadores presentes nos circuitos.

Uma das principais limitações do projeto é 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 devido aos microfones que são sensores omnidirecionais, que detectam sinais sonoros vindos de todas as direções, dificultado a detecção do sentido do sinal. No decorrer do projeto, não foi posicionado a fonte sonora nessa circunstância para realizar ensaios práticos. Outra limitação do projeto, como dito anteriormente, é não conseguir estimar o azimute para 0° , sendo assim, necessário outra solução para estimar tal caso.

REFERÊNCIA BIBLIOGRÁFICAS

- [1] LIMA, G. S (2021). Estimador de direção de chegada. Faculdade de Tecnologia, Universidade de Brasília, 2021.
- [2] REIS, Y. S (2019). Sistema de detecção para ondas sonoras. Faculdade de Tecnologia, Universidade de Brasília, 2019.
- [3] 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.
- [4] DORF, RICHARD C. Introdução aos circuitos elétricos (8ª ed.).
- [5] FEIJOO, ANA M. L. C. Parte I – Estatística descritiva. Disponível em: [feijoo-9788579820489-06.pdf \(scielo.org\)](#)
- [6] Chapman, S. (2005). Electric Machinery Fundamentals (4ª ed.). New York: McGraw-Hill
- [7] Alan V. Oppenheim e Ronald W. Schaffer (2009). Discrete-Time Signal Processing (3ª ed.)
- [8] MSP430x5xx and MSP430x6xx Family – User Guide. Disponível em: [MSP430x5xx and MSP430x6xx Family User's Guide \(Rev. Q\) \(ti.com\)](#)
- [9] HC-05 - Bluetooth to Serial Port Module. Disponível em: [HC-05 Datasheet.pdf \(components101.com\)](#)