

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA  
CATARINA  
CAMPUS FLORIANÓPOLIS  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
CURSO SUPERIOR DE TECNOLOGIA EM ELETRÔNICA INDUSTRIAL**

**GUILHERME WILM GARCIA**

**ESTUDO E IMPLEMENTAÇÃO DE CANCELAMENTO ATIVO DE  
RUÍDO USANDO UM SISTEMA MICROCONTROLADO**

**FLORIANÓPOLIS, JULHO DE 2018.**

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA  
CATARINA  
CAMPUS FLORIANÓPOLIS  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
CURSO SUPERIOR DE TECNOLOGIA EM ELETRÔNICA INDUSTRIAL**

**GUILHERME WILM GARCIA**

**ESTUDO E IMPLEMENTAÇÃO DE CANCELAMENTO ATIVO DE  
RUÍDO USANDO UM SISTEMA MICROCONTROLADO**

Trabalho de Conclusão de Curso  
submetido ao Instituto Federal de  
Educação, Ciência e Tecnologia de Santa  
Catarina como parte dos requisitos para  
obtenção do título de Tecnólogo em  
Eletrônica Industrial.

Professor Orientador: Alexandre Leizor  
Szczipak, Dr. Eng.

**FLORIANÓPOLIS, JULHO DE 2018.**

Ficha de identificação da obra elaborada pelo autor.

Garcia, Guilherme Wilm

Estudo e Implementação de Cancelamento Ativo de Ruído Usando um Sistema Microcontrolado / Guilherme Wilm Garcia ; orientação de Alexandre Leizor Szczupak. - Florianópolis, SC, 2018.

73 p.

Trabalho de Conclusão de Curso (TCC) - Instituto Federal de Santa Catarina, Câmpus Florianópolis. CST em Eletrônica Industrial. Departamento Acadêmico de Eletrônica.

Inclui Referências.

1. Filtros adaptativos. 2. Controle ativo de ruído.

3. Processamento digital de sinais. 4. ANC. 5. Microcontroladores  
I. Leizor Szczupak, Alexandre. II. Instituto Federal de Santa Catarina.

Departamento Acadêmico de Eletrônica.



**ATA DA BANCA FINAL DO TRABALHO DE CONCLUSÃO DO CURSO SUPERIOR DE  
TECNOLOGIA EM ELETRÔNICA INDUSTRIAL DO INSTITUTO FEDERAL DE SANTA CATARINA  
- Nº 125**

Aos 9 dias do mês de **julho**, de **2018**, às **09h40min**, o estudante **Guilherme Wilm Garcia** apresentou o seu Trabalho de Conclusão de Curso para julgamento à Banca Examinadora constituída pelos seguintes integrantes: Prof. **Alexandre Leizor Szczupak, Dr. Eng.** (orientador/presidente da banca/IFSC), Prof. **André Luis Dalcastagnê, Dr. Eng.** (IFSC) e Prof. **Fernando Santana Pacheco, Dr. Eng.** (IFSC). A sessão pública de defesa foi aberta pelo Presidente da Banca, que apresentou a Banca Examinadora e deu continuidade aos trabalhos, fazendo uma breve referência ao TCC que tem como título **ESTUDO E IMPLEMENTAÇÃO DE CANCELAMENTO ATIVO DE RUÍDO USANDO UM SISTEMA MICROCONTROLADO**. Na sequência, o(a) estudante teve até 30 minutos para a exposição de seu trabalho, e cada integrante da Banca Examinadora fez a arguição após a apresentação do mesmo. Finalmente, foi aberto um espaço aos presentes para eventuais perguntas ou comentários sobre o trabalho apresentado. Ouvidas as explicações do(a) estudante, a Banca Examinadora, reunida em caráter sigiloso, para proceder à avaliação final, deliberou pelo conceito **8 (oito)**. Foi dada ciência ao(à) estudante que a versão final do trabalho deverá ser entregue até o dia **15/08/2018**, com as devidas alterações sugeridas pela banca. Nada mais havendo a tratar, a sessão foi encerrada às 11h30min, dela sendo lavrada a presente ata, que, uma vez aprovada, foi assinada por todos os membros da Banca Examinadora e pelo(a) estudante.

Prof.(a) Orientador(a) / Presidente: \_\_\_\_\_

Prof.(a) Avaliador 1: \_\_\_\_\_

Prof.(a) Avaliador 2: \_\_\_\_\_

Acadêmico: Guilherme Wilm Garcia

## **AGRADECIMENTOS**

Primeiramente, gostaria de agradecer a toda a minha família, sobretudo meus pais Jefferson Garcia e Andréa Wilm, e amigos pelo suporte e incentivo durante esta minha jornada.

Gostaria de agradecer a todos os colegas e professores do Instituto Federal de Santa Catarina pelo conhecimento adquirido e pelas valiosas trocas de experiência. Também ao Fábio Cabral Pacheco e ao Pablo Ribeiro Kodama, pelo suporte com equipamentos e na confecção das placas de circuito impresso. Agradeço ao Prof. Marcos Vinicius Matsuo por transmitir conhecimentos teóricos e pelas sugestões oferecidas durante o desenvolvimento deste projeto.

Um especial agradecimento ao Prof. André Luís Dalcastagnê, pela oportunidade e orientação em um projeto de pesquisa do CNPq durante o curso.

Também à CAPES, pela bolsa oferecida no programa de intercâmbio Ciência sem Fronteiras, e à Universidade de Halmstad, na Suécia, por me receber e me proporcionar novas experiências pessoais e acadêmicas.

Meus sinceros agradecimentos ao mestre Luiz Carlos Pinagé de Lima Filho pelo estágio oferecido na Due Laser e a todos os colegas da Mensura que me auxiliaram tanto com tarefas na empresa, quanto com questões relacionadas a este trabalho.

Finalmente, uma dedicatória especial de agradecimento ao meu orientador, o Prof. Alexandre Leizor Szczupak, por toda paciência e devoção na realização deste projeto.

## RESUMO

Neste trabalho, apresentam-se o estudo e as técnicas usadas no desenvolvimento de um sistema de controle ativo de ruído acústico implementado em um sistema microcontrolado de baixo custo. Visam-se a compreensão da teoria de controle ativo de ruído e a solução das dificuldades de uma implementação prática. Faz-se uso de ferramentas de simulação e de desenvolvimento para a programação dos algoritmos LMS e FxLMS em um sistema de desenvolvimento microcontrolado. Os sinais sonoros foram captados através de microfones com a tecnologia MEMS. Suas respostas em frequência foram ajustadas através de circuitos de pré-processamento formados por amplificadores e filtros analógicos, produzidos em placas de circuito impresso. Foi construído um gabinete de proteção para o equipamento por meio de uma impressora 3D e uma máquina de corte a laser. Foi construído também um sistema acústico com tubos, alto-falantes e microfones, tipicamente utilizado em testes de cancelamento ativo de ruído. Foram realizados testes que visavam à redução da amplitude de sinais senoidais propagados dentro deste sistema acústico. Os resultados obtidos em cenários simulados e práticos apresentaram diferenças significativas e embasam caminhos futuros para projetos com o mesmo foco.

**Palavras-chave:** Filtros adaptativos. Controle ativo de ruído. Processamento digital de sinais. ANC. Microcontroladores.

## ABSTRACT

*This paper presents the study and the techniques applied on the development of an active noise control system implemented with a low cost microcontroller system. The intention is to comprehend the theory of active noise control and to solve the difficulties of a practical implementation. It employs simulation and development tools to implement the LMS and FxLMS algorithms on a microcontroller development system. The acoustic signals were captured with MEMS microphones. Their frequency responses were adjusted by pre-processing circuits composed by analog amplifiers and filters, built on printed circuit boards. A protection box was built for the hardware using a 3D printer and a laser cutting machine. It was also built an acoustic structure made of tubes, speakers and microphones, typically used in tests of active noise control systems. Tests were made aiming the reduction of the amplitude of sine waves propagating through this acoustic system. The results obtained, in simulated and practical scenarios, presented meaningful differences and serve as building blocks for future developments with the same focus.*

**Key-words:** Adaptive filters. Active noise control. Digital signal processing. ANC. Microcontrollers

## SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	<b>8</b>
1.1. Definição do problema.....	8
1.2. Justificativa.....	9
1.3. Objetivo geral.....	10
1.4. Objetivos específicos.....	10
1.5. Estrutura do trabalho.....	10
<b>2. FUNDAMENTAÇÃO TEÓRICA</b> .....	<b>12</b>
2.1. Audição.....	12
2.2. Fones de ouvido.....	14
2.3. Controle Ativo de Ruído (ANC).....	15
2.3.1. LMS.....	18
2.3.2. MSE.....	20
2.3.3. NLMS.....	20
2.3.4. Estimativa do caminho secundário.....	21
2.3.5. FxLMS.....	23
2.3.6. FxLMS com adaptação lenta.....	24
2.3.7. FxNLMS.....	25
2.4. Microfones.....	25
2.4.1. Impedância de saída.....	25
2.4.2. Sensibilidade.....	25
2.4.3. Resposta em frequência.....	26
2.4.4. Ruído intrínseco.....	26
2.4.5. Diretividade.....	26
2.4.6. Microfone MEMS.....	27
2.5. Microcontroladores.....	29
2.5.1. Microprocessador.....	30
2.5.2. Cortex-M4F.....	30
2.5.3. Teensy.....	31
2.5.4. Audio Adaptor Board.....	32
<b>3. DESENVOLVIMENTO DO SISTEMA</b> .....	<b>34</b>
3.1. Simulação no MATLAB.....	34
3.1.1. Simulação da estimativa do caminho de propagação secundário (NLMS).....	35
3.1.2. Simulação do algoritmo de cancelamento ativo de ruído (FxNLMS).....	39
3.2. Implementação do sistema.....	43
3.2.1. Implementação do código no Teensy sem elementos externos.....	43
3.2.2. Circuito de amplificação e equalização para os microfones.....	53
3.2.3. Gabinete de proteção para o microcontrolador.....	60
3.2.4. Implementação do sistema com tubos de PVC.....	60
<b>4. RESULTADOS</b> .....	<b>67</b>
<b>5. CONCLUSÕES</b> .....	<b>69</b>
<b>REFERÊNCIAS</b> .....	<b>70</b>
<b>APÊNDICE A – CÓDIGO DESENVOLVIDO PARA O PROJETO</b> .....	<b>73</b>



## 1. INTRODUÇÃO

Diariamente, pessoas são expostas a ruídos sonoros indesejados vindos das mais diversas fontes, sejam elas máquinas, automóveis, construções civis e outras. Não bastasse o incômodo causado por estas fontes, a exposição ao ruído pode provocar danos psicológicos e físicos. Agravando-se a tudo isso, o uso de fones de ouvidos em ambientes ruidosos pode levar as pessoas a aumentar excessivamente o volume de seu equipamento pessoal de áudio para compensar os sons indesejados.

Existem dois tipos de abordagens para a atenuação de ruído: o passivo e o ativo. O passivo é o método mais utilizado e mais simples de ser produzido, pois consiste somente de uma barreira física entre o canal auditivo do ouvinte e o meio externo. Devido às limitações de espaço em fones de ouvido, este método só se torna eficiente para altas frequências. O ativo baseia-se em equipamentos eletrônicos capazes de gerar um sinal que se apropria do fenômeno da interferência destrutiva de ondas, o qual significa que ondas sobrepostas de igual magnitude e defasadas em 180° se anulam.

Este trabalho tem por objetivo estudar este fenômeno com uma aplicação prática. Pretende-se a produção de um sistema de controle (ou cancelamento) ativo de ruído (ANC, em inglês *Active Noise Control*). As etapas realizadas foram de pesquisa, de simulação, de aplicação em um microcontrolador e de validação em testes de bancada.

### 1.1. Definição do problema

O aprimoramento tecnológico nos proporciona ouvir música em qualquer lugar por meio de dispositivos eletrônicos portáteis e de fones de ouvido. Como a maioria das pessoas vive em centros urbanos, ambientes altamente ruidosos, elas tendem a aumentar demasiadamente o volume de suas músicas e outros áudios. Esta exposição demasiada pode levar a danos permanentes no sistema auditivo (HERRERA *et al.*, 2016).

De acordo com a Organização Mundial da Saúde (OMS), aproximadamente 1,1 bilhão de jovens ao redor do mundo correm o risco de perda auditiva causada pelo uso desregrado de aparelhos de áudio pessoais para uso recreacional e pela exposição a níveis prejudiciais em eventos de entretenimento, como casas noturnas, bares e estádios (WHO, 2015). Além do impacto funcional, as pesquisas da OMS constataram que a exposição duradoura a níveis de som elevados também pode afetar de outras formas a saúde do indivíduo provocando, por exemplo: distúrbios do sono, estresse, fadiga, irritabilidade, cansaço, isolamento, solidão e dor de cabeça (WHO, 2017).

## 1.2. Justificativa

Existe uma ampla variedade de fones de ouvido com cancelamento de ruído no mercado. Porém, os modelos disponíveis geralmente apresentam um custo tipicamente mais elevado do que os tradicionais.

Como mencionado anteriormente, existem problemas de saúde associados à exposição prolongada ao ruído, seja no trabalho, no ambiente urbano ou em eventos de entretenimento.

É possível criar um equipamento portátil de baixo custo que possa ser usado em conjunto com equipamentos de áudio para amenizar esses problemas. Existem dispositivos microcontroladores, com baixo custo e capacidade de processamento elevada, que podem ser usados para essa tarefa (CHANG; LI, 2011) (VU *et al.*, 2015).

Os métodos de cancelamento de ruído podem ser divididos em passivos e ativos. Os métodos passivos se utilizam de propriedades físicas do aparelho, como barreiras, invólucros e silenciadores. Estes métodos são mais tradicionais e absorvem ruídos de amplas gamas de frequência, particularmente em frequências elevadas. Por outro lado, apresentam desvantagens como custo elevado, pouca efetividade no cancelamento de ruído de baixas frequências e ainda o aumento do volume e do peso do sistema. Já os métodos ativos, foco deste trabalho, são realizados por meio de circuitos eletrônicos, sensores e atuadores. Estes dispositivos são utilizados para criar um sinal acústico com o objetivo de eliminar o

sinal de ruído através de superposição (KURO; MORGAN, 1999). Entretanto, os métodos ativos não são efetivos em altas frequências (acima de 1 kHz), além de aumentarem a complexidade e o custo do sistema.

Este projeto foi imaginado de modo a ajudar a prevenção de danos à audição e fornecer maior conforto acústico. Além disso, através de um equipamento portátil de baixo custo, realizar as etapas de processamento de sinais necessárias para atingir esses objetivos. É ainda uma tentativa de acrescentar funcionalidades a produtos já disponíveis no mercado, procurando manter um custo baixo.

### **1.3. Objetivo geral**

Construir um equipamento com base em Processamento Digital de Sinais (DSP, do inglês, *Digital Signal Processing*) voltado para o controle ativo de ruído em sistemas acústicos.

### **1.4. Objetivos específicos**

Foram selecionadas as seguintes etapas para a realização do projeto:

- a) analisar a viabilidade de implementação do sistema;
- b) definir um microcontrolador com capacidade de DSP;
- c) determinar o algoritmo ANC a ser implementado;
- d) simular o algoritmo ANC;
- e) implementar o algoritmo ANC no microcontrolador;
- f) construir uma estrutura para testar o algoritmo;
- g) coletar dados do algoritmo;
- h) ajustar algoritmo de acordo com os dados coletados;
- i) adaptar um fone de ouvido para a implementação do sistema proposto;
- j) apresentar os resultados.

### **1.5. Estrutura do trabalho**

O desenvolvimento do projeto está organizado do seguinte modo:

No Capítulo 2, apresenta-se o estudo da literatura para se obter uma base teórica do problema, dos principais componentes eletrônicos do sistema e do algoritmo a ser realizado. No Capítulo 3 está descrito o desenvolvimento do sistema

em etapas, desde a simulação até a sua construção e implementação do algoritmo. No Capítulo 4, apresentam-se os resultados obtidos. Finalmente, no Capítulo 5 são relatadas as conclusões, dificuldades e melhorias futuras para o projeto.

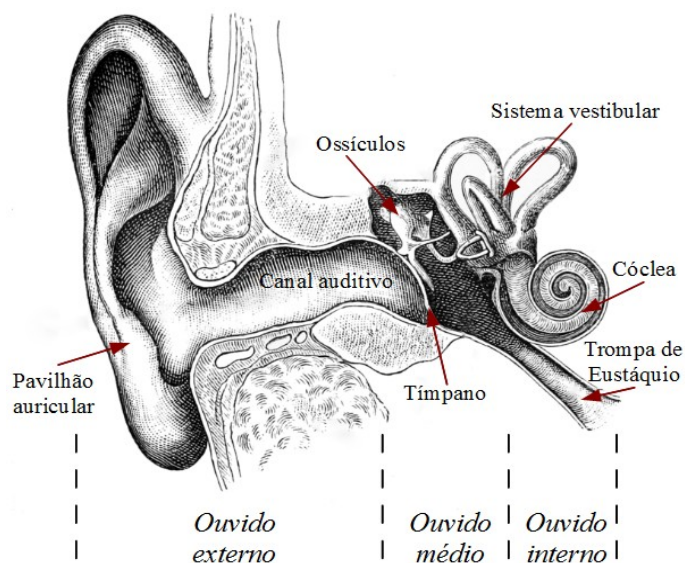
## 2. FUNDAMENTAÇÃO TEÓRICA

Esta seção tem como finalidade apresentar os conceitos que sustentam o desenvolvimento da pesquisa com base em obras de outros autores.

### 2.1. Audição

O aparelho auditivo (Figura 1) é o órgão responsável pela audição e pelo equilíbrio nos seres humanos. Ele é dividido anatomicamente em três partes: ouvido externo, ouvido médio e ouvido interno. O ouvido externo é composto pelo pavilhão auricular (orelha) e o canal auditivo, com suas funções principais sendo a captação e a filtragem do som. O ouvido médio é composto basicamente por uma cavidade, separada do canal auditivo pela membrana timpânica, e por três ossículos com variadas funções, como captar a vibração no tímpano, reduzir a pressão sonora e introduzir ondas de pressão no ouvido interno. Por último, a cóclea e o sistema vestibular formam o ouvido interno, que possibilitam a transferência de informações de equilíbrio e audição ao cérebro (DRAKE; VOGL; MITCHELL, 2014).

Figura 1 - Sistema auditivo humano.



Fonte: Imagem adaptada de (WIKIMEDIA COMMONS, 2006).

Resumidamente, o som é percebido pelo indivíduo quando ondas sonoras são captadas pelo pavilhão auricular, passam pelo canal auditivo e fazem o tímpano vibrar. Esta vibração é traduzida pelo ouvido médio em ondas de pressão no fluido do ouvido interno. A agitação no fluido movimenta milhares de células ciliadas de dentro da cóclea, que transforma as ondas de compressão em impulsos nervosos enviados ao cérebro para serem interpretados (DRAKE; VOGL; MITCHELL, 2014).

A percepção de intensidade do som, comumente medida em  $\text{dB}_{\text{SPL}}$ , está relacionada com a sua amplitude. A sigla SPL significa “nível de pressão sonora” (do inglês, *Sound Pressure Level*) que é a medida numa escala logarítmica da pressão relativa a um valor de referência, dado pela Equação 1:

$$SPL = 20 \log \left( \frac{P_E}{P_{REF}} \right). \quad (1)$$

Onde:

$P_E$  = medida em pascal (Pa) da pressão efetiva da onda sonora

$P_{REF}$  = medida de referência em pascal (Pa) da pressão efetiva da onda sonora

$SPL$  = nível de pressão sonora em decibéis referenciados a  $P_{REF}$  (comumente utilizado somente como  $\text{dB}_{\text{SPL}}$ )

O Pa (pascal) é a unidade de medida de pressão do SI (Sistema Internacional de Unidades, do francês *Le Système International d'Unités*). O valor de referência utilizado quando o meio é o ar é o limiar da audição humana: 20  $\mu\text{Pa}$  ou 0  $\text{dB}_{\text{SPL}}$ . O sistema auditivo apresenta alta sensibilidade. Sons abruptos extremamente altos e sons altos por períodos prolongados podem danificar os ouvidos temporariamente ou até permanentemente e podem causar outros danos auditivos (KINSLER *et al.*, 2000) (PURVES *et al.*, 2008).

Ondas acústicas com frequências no intervalo de aproximadamente 20 Hz até 20 kHz são perceptíveis ao ouvido humano. A filtragem e a amplificação feitas pelo sistema auditivo priorizam a faixa de frequências por volta de 2 kHz a 5 kHz, com um pico em 3 kHz. Entretanto, isto deixa os ouvidos mais suscetíveis a danos e perdas auditivas nesta faixa de frequências (PURVES *et al.*, 2008).

## 2.2. Fones de ouvido

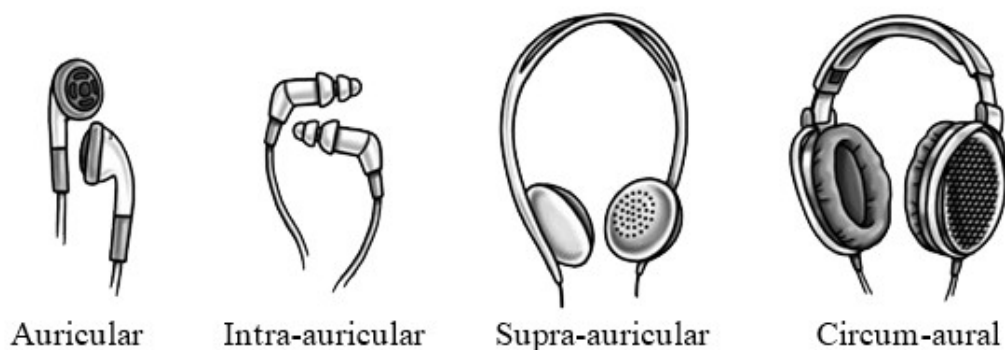
Os fones de ouvido são um par de transdutores eletroacústicos posicionados nos ouvidos do usuário, que o permitem reproduzir dois sinais independentes de áudio (SHURE, 2016).

Aplicações de fones de ouvido abrangem desde o uso para o lazer até o uso profissional, como, por exemplo, o uso por músicos. Atualmente, para o uso individual, os meios pelos quais os fones de ouvido recebem o sinal de áudio podem ser: o leitor de MP3, o computador pessoal, o *home theater*, entre outros.

Os fones são comumente conectados à fonte de áudio por meio de fios com conectores. Mais recentemente, começaram a surgir fones com capacidade de se conectarem à fonte de sinal sem o uso de fios (*wireless*), mas através de uma conexão Bluetooth, o que ocasionou a propagação ainda maior dos fones de ouvido em locais públicos.

Os fones de ouvido se dividem em quatro categorias gerais: os auriculares, os intra-auriculares, os supra-auriculares e os circum-aurais (SHURE, 2016) (BORWICK, 2001).

Figura 2 - Categorias gerais de fones de ouvido.



Fonte: Imagem adaptada de (GETDRAWINGS, 2018).

Ambos os fones de ouvido supra-auriculares e circum-aurais podem ser divididos em três tipos: abertos, fechados e semiabertos. Isso é determinado verificando se o fone possui ou não a parte de trás do alto-falante fechada por uma cápsula, podendo também ser parcialmente fechado no caso do semiaberto. Essas

diferenças refletem o quanto o som ambiente interfere no uso dos fones e o quanto os fones liberam o som produzido ao ambiente, o que pode acarretar em um aumento excessivo do volume pelo usuário (BEYERDYNAMIC, 2016).

A relação entre o volume de ar e a distância da fonte de som para o tímpano resulta no nível relativo da exposição ao ruído. Sendo assim, quanto mais próximos estiverem os alto-falantes dos ouvidos internos, maiores são os riscos de perdas auditivas.

Fones de ouvido são comumente usados em locais públicos, porém isto produz dois efeitos como consequência, um no público e um no ouvinte:

- a) efeito no público: particularmente para fones de ouvido supra-auriculares, o aumento excessivo no volume causa vazamentos de áudio que podem ser perturbadores para as pessoas ao redor;
- b) efeito no ouvinte: o aumento gradativo do volume do fone de ouvido para compensar o ruído ambiente e também pelo desejo do usuário de satisfazer seus ouvidos com intensidades maiores de som, podem gerar danos irreversíveis à audição com o passar dos anos (BORWICK, 2001).

Os limites diários de som para o ouvinte estão expostos na Tabela 1 abaixo:

Tabela 1 - Limites diários de som para o ouvinte.

<b>Limite diário total (min)</b>	<b>Nível de intensidade sonora (dB(A)<sub>SPL</sub>)</b>
120	99
60	102
30	105
15	108

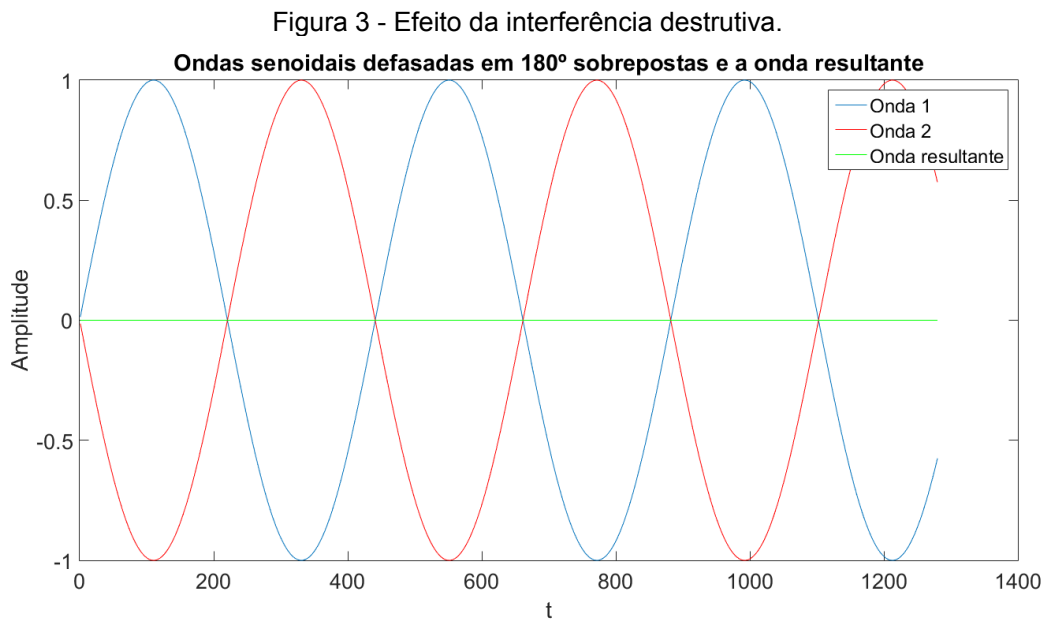
Fonte: BORWICK (2001).

### **2.3. Controle Ativo de Ruído (ANC)**

Silenciadores passivos são usualmente grandes, dispendiosos e ineficientes em baixas frequências (KUO; MORGAN, 1999). Alternativamente, pode-se usar o controle ativo de ruído, um método para a redução do ruído baseado no



princípio da superposição. O ANC é obtido ao se introduzir o mesmo ruído, com amplitude igual, porém com fase invertida em relação ao sinal original, através de fontes secundárias, obtendo-se o cancelamento de ambos os sinais (KURO; MORGAN, 1999). A Figura 3 representa um exemplo deste fenômeno com duas ondas senoidais sobrepostas em  $180^\circ$  e sua onda resultante:



Fonte: Elaboração própria (2018).

Por outro lado, um sistema ANC pode eficientemente atenuar os ruídos de baixas frequências (abaixo de 1 kHz) e possui um potencial benéfico em tamanho, peso, volume e custo (KURO; MORGAN, 1999).

Como as características da fonte de ruído acústico e do ambiente são variantes no tempo, o conteúdo frequencial, a amplitude e a fase do ruído indesejado são não-estacionários. Um sistema ANC deve, portanto, ser adaptativo de modo a acompanhar estas variações (KURO; MORGAN, 1999).

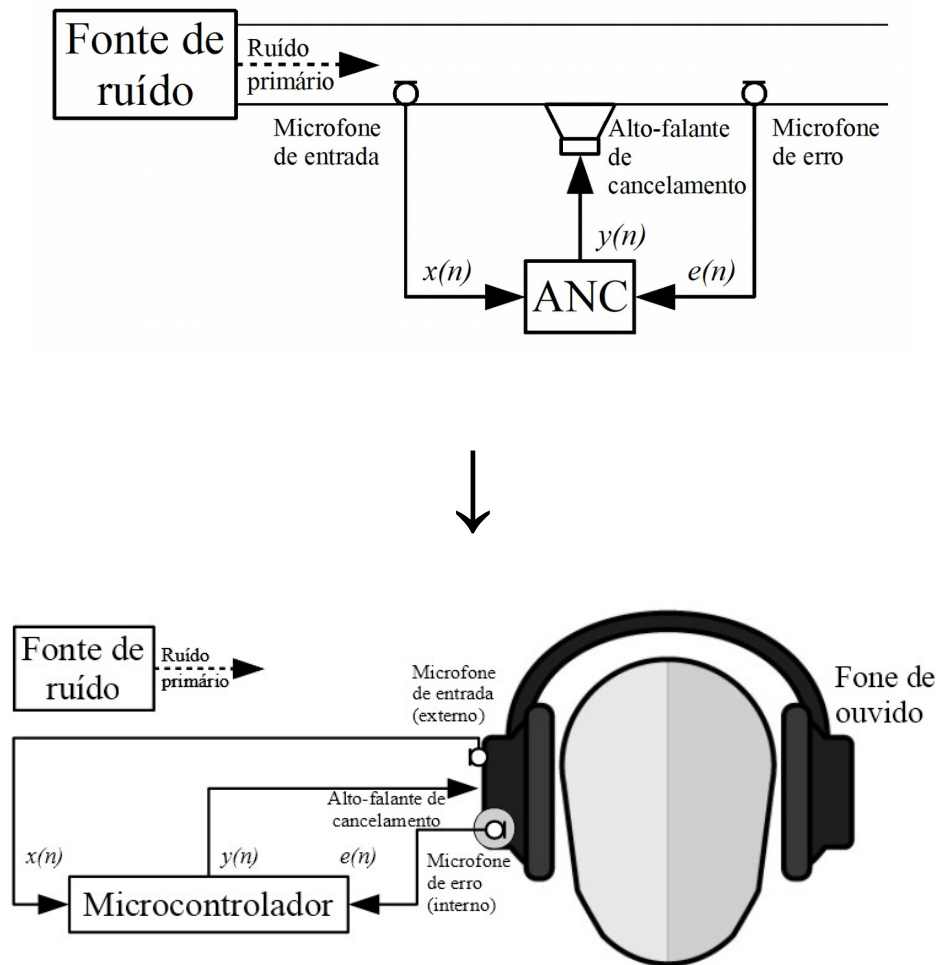
Um filtro adaptativo criado em torno do algoritmo LMS (*Least Mean Squares*) será utilizado neste projeto, porém, em sua forma básica ele não consegue operar adequadamente em um sistema ANC devido a atrasos pelo sistema que devem ser compensados. Isto leva à busca de estruturas alternativas derivadas do LMS (LIU *et al.*, 2008).

Existem diferentes tipos de estruturas derivadas do algoritmo LMS para o controle ativo de ruído, como a estrutura *feedforward* (FxLMS), a *feedback* e a híbrida. Este projeto estará focado em implementar o algoritmo FxLMS, por ser a solução mais efetiva e ser bastante tolerante a erros feitos na estimativa do caminho secundário, que serão abordados na etapa de desenvolvimento do sistema na Seção 3 (LIU *et al.*, 2008).

Para o projeto da estrutura *feedforward*, será necessária a utilização de dois microfones, um externo para captar o ruído ambiente e o outro interno para realimentar o sistema com o ruído residual após o cancelamento. Além disso, também será preciso um circuito eletrônico programável, um microcontrolador no caso deste projeto, para gerar a onda sonora “antirruído” com uma polaridade oposta ao som que chega no microfone.

Um sistema acústico formado por tubos, alto-falantes e microfones é tipicamente utilizado em testes de cancelamento ativo de ruído. A Figura 4 faz a comparação entre o modelo de uma estrutura com tubos construída para os testes deste trabalho e o modelo da estrutura ANC para fones de ouvido equivalente:

Figura 4 - Comparação entre os modelos equivalentes com tubos e em fones de ouvido.



Fonte: Elaboração própria (2018).

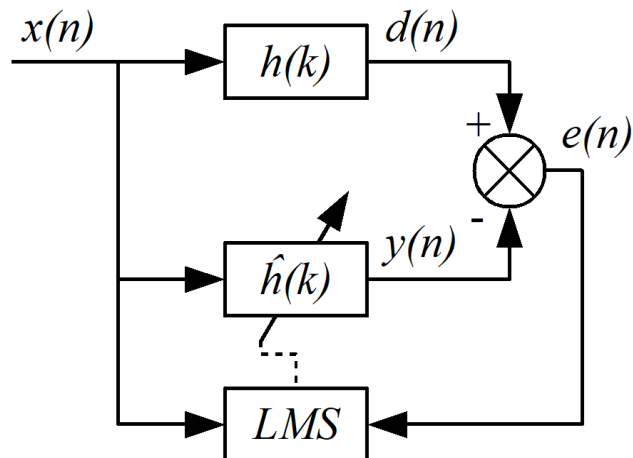
A seguir será feita a explicação dos algoritmos escolhidos para realizar o processo de cancelamento de ruído ativo, com base na literatura. Não serão descritos os passos para se chegar nas equações. Estas serão apenas apresentadas para o leitor com uma explicação de seu uso para os algoritmos.

### 2.3.1. LMS

O algoritmo LMS tem por objetivo minimizar o erro quadrático médio (MSE, do inglês *mean squared error*) do sistema por meio do ajuste dos coeficientes de um filtro adaptativo (DINIZ, 2002) (INGLE; PROAKIS, 2000).

A Figura 5 esquematiza o diagrama de blocos do filtro LMS genérico.

Figura 5 - Diagrama de blocos do algoritmo LMS.



Fonte: Elaboração própria (2018).

Sendo  $n$  um índice amostral e  $k$  um índice de coeficientes, segue que  $x(n)$  representa o sinal de entrada,  $h(k)$  a resposta ao impulso do sistema desconhecido,  $\hat{h}(k)$  os coeficientes do filtro adaptativo a ser projetado,  $d(n)$  o sinal desejado,  $y(n)$  o sinal de saída do filtro e  $e(n)$  o sinal de erro do sistema (DINIZ, 2002) (INGLE; PROAKIS, 2000) (KUO; MORGAN, 1999). Idealmente, após a adaptação, os coeficientes de  $\hat{h}(k)$  serão iguais aos coeficientes de  $h(k)$ .

Podem-se inicializar os coeficientes do filtro adaptativo como zeros.

$$\hat{h}(k)=0, \quad 0 \leq k < N. \quad (2)$$

O número de amostras de  $x(n)$  a serem processadas a cada iteração tem o tamanho do filtro  $N$ . Considera-se  $T$  o número total de amostras a serem processadas. A cada iteração  $n$  do algoritmo, uma nova amostra de  $x(n)$  é acrescentada ao processamento, enquanto a mais antiga é descartada.

$$\mathbf{x}(n)=[x(n) \quad x(n-1) \quad \dots \quad x(n-N+1)], \quad 0 \leq n < T. \quad (3)$$

O sinal de saída do filtro  $y(n)$  é

$$y(n)=\sum_{k=0}^{N-1} \hat{h}_{n-1}(k)x(n-k). \quad (4)$$

E o erro  $e(n)$  é

$$e(n)=d(n)-y(n). \quad (5)$$

A equação de atualização dos coeficientes  $\hat{h}(n)$  do filtro é dada por

$$\hat{h}_n(k) = \hat{h}_{n-1}(k) + 2\mu e(n)x(n-k), \quad 0 \leq k < N. \quad (6)$$

Sendo  $\mu$  o fator de convergência (também referido como o tamanho do passo) do algoritmo. Este fator pondera a velocidade de adaptação do filtro (DINIZ, 2002). O multiplicador 2 em  $2\mu e(n)x(n-k)$  é, comumente, incorporado à constante  $\mu$ .

Por sua baixa complexidade computacional, garantia de convergência em um ambiente estacionário e comportamento estável quando implementado com aritmética de precisão finita, o LMS é um dos algoritmos adaptativos mais utilizados em diversas aplicações (DINIZ, 2002).

### 2.3.2. MSE

A Equação 6 deriva da equação

$$\hat{h}_n(k) = \hat{h}_{n-1}(k) - \frac{\mu}{2} \nabla \zeta(n), \quad 0 \leq k < N. \quad (7)$$

Sendo  $\nabla \zeta(n) = -2e(n)x(n-k)$  (KUO; MORGAN, 1999).

Conforme comentado anteriormente, o MSE é a figura de mérito que o algoritmo LMS deverá minimizar ao longo do tempo, sendo definido por

$$MSE = E(|e(n)|^2). \quad (8)$$

Sendo  $E$  o operador valor esperado.

Com isso, espera-se que o algoritmo produza um conjunto de coeficientes próximo ao conjunto de coeficientes do filtro a ser estimado (INGLE; PROAKIS, 2000).

### 2.3.3. NLMS

Uma das soluções para melhorar a velocidade de convergência de MSE do sistema é o uso do algoritmo LMS normalizado, o NLMS (do inglês, *normalized least mean square*). Pode-se mostrar que esta abordagem leva também à minimização do MSE quando se aplica um algoritmo LMS. Como o termo de ajuste

do algoritmo LMS é proporcional às amostras recentes do sinal de entrada, se a amplitude do sinal de entrada aumentar também ocorrerá aumento do erro da estimativa de ajuste. Uma solução é realizar a normalização do termo de ajuste pela energia do sinal de entrada. (HAYKIN, 2014)

A equação de atualização dos coeficientes do NLMS é dada por

$$\hat{h}_n(k) = \hat{h}_{n-1}(k) + \frac{\mu_n}{\lambda + \|\mathbf{x}(n)\|^2} e(n) x(n-k), \quad 0 \leq k < N, \quad 0 \leq n < T \quad (9)$$

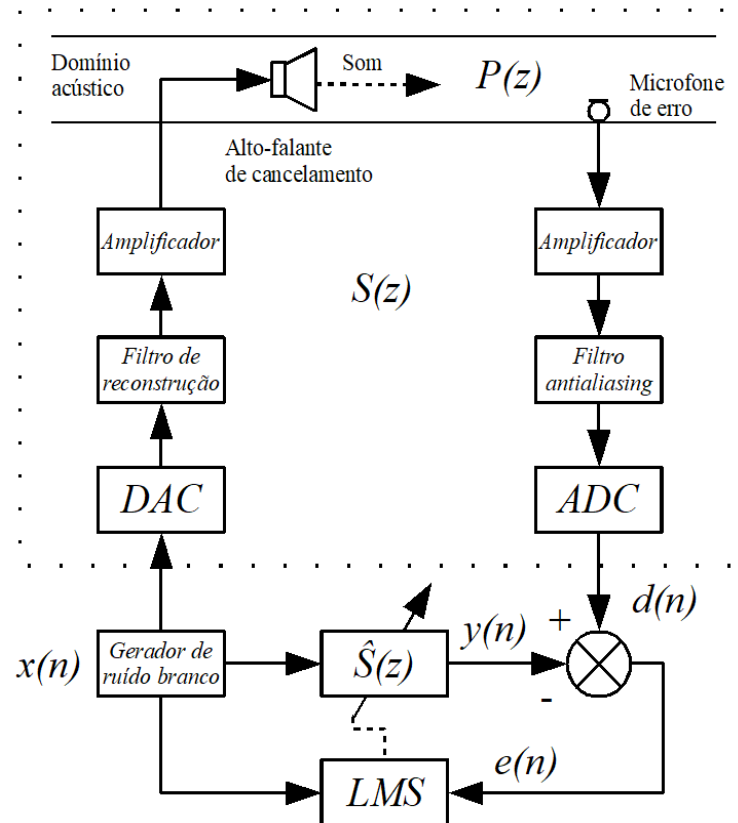
O coeficiente de convergência  $\mu_n$  deve ser escolhido com um valor entre 0 e 2, para o controle do desajuste. É adicionada uma constante  $\lambda$  de valor baixo, para evitar passos muito largos se a energia do sinal se tornar muito pequena. (DINIZ, 2002).

#### 2.3.4. Estimativa do caminho secundário

Atrasos desde a captura do ruído indesejado até a geração do sinal de saída do filtro causarão alterações significativas justamente neste sinal de saída. Os causadores destes atrasos englobam os conversores A/D e D/A, o filtro *antialiasing*, o filtro de reconstrução, os amplificadores, o alto-falante, o microfone de erro e o caminho acústico entre o microfone de entrada e o microfone de erro. O algoritmo para a geração do sinal antirruído utiliza uma estimativa da resposta desse caminho, chamado de caminho secundário  $S(z)$ , para compensar seus efeitos (MU; KO; RHEEM, 2013) (KUO; MORGAN, 1999).

A Figura 6 ilustra o diagrama de blocos para a estimativa do caminho secundário.

Figura 6 - Diagrama de blocos da estimativa do caminho secundário.



Fonte: Elaboração própria (2018).

Como o caminho secundário é invariável para este projeto, sua estimativa pode ser feita antes do cancelamento de ruído começar (chamada na literatura de estimativa *offline* de  $S(z)$ ), aplicando-se no sistema o ruído branco como um sinal de treinamento. (MU; KO; RHEEM, 2013).

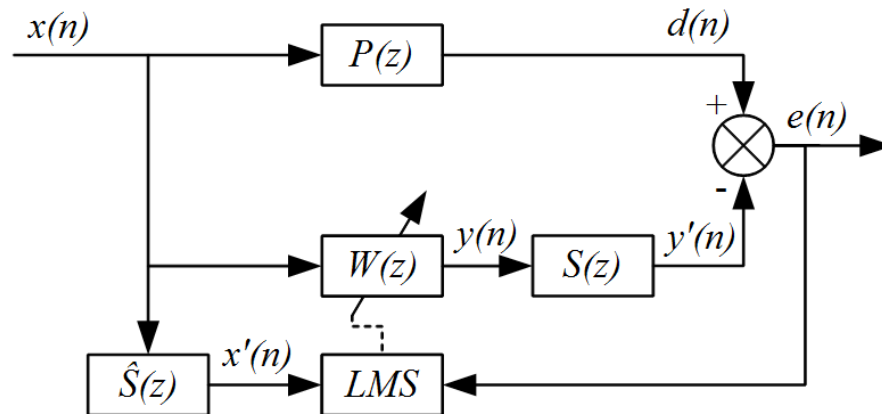
Logo, o filtro  $\hat{S}(z)$  será adaptado utilizando o algoritmo LMS para produzir coeficientes que sejam os mais semelhantes possíveis aos do caminho secundário  $S(z)$ . As equações utilizadas serão as mesmas da seção sobre o LMS, somente mudando a nomenclatura dos blocos do filtro de  $h(n)$  para  $S(z)$  e  $\hat{h}(n)$  para  $\hat{S}(z)$ . Após a obtenção destes coeficientes, o filtro  $\hat{S}(z)$  será utilizado no algoritmo FxLMS para compensar o atraso gerado por  $S(z)$ .

### 2.3.5. FxLMS

Após o processo de estimativa do caminho secundário *offline*, incorporam-se os coeficientes desse filtro na elaboração da estrutura *feedforward*, usada para gerar o sinal antirruído. Deste modo é possível realizar a compensação dos atrasos no sistema e alinhar corretamente no tempo os sinais dos microfones de erro e de entrada (KUU; MORGAN, 1999).

A estrutura FxLMS está exposta na Figura 7.

Figura 7 - Diagrama de blocos do algoritmo FxLMS.



Fonte: Elaboração própria (2018).

O caminho primário é representado por  $P(z)$ , o caminho secundário por  $S(z)$  e sua compensação por  $\hat{S}(z)$ . O filtro adaptativo  $W(z)$  deve ser um modelo de  $P(z)$  (KUU; MORGAN, 1999).

As variáveis  $x(n)$ ,  $y(n)$  e  $d(n)$  seguem sendo as mesmas da estrutura LMS: o ruído indesejado, a saída do filtro adaptativo e o sinal de saída desejado, respectivamente. Porém, agora o sinal de erro  $e(n)$  será a diferença entre  $d(n)$  e o sinal de saída do filtro adaptativo filtrado pelo  $S(z)$ , ou seja,  $y'(n)$ . Portanto, o sinal de referência para o erro no algoritmo LMS não será mais  $x(n)$ , e sim  $x'(n)$ , o sinal resultante da filtragem por  $\hat{S}(z)$  (KUU; MORGAN, 1999).

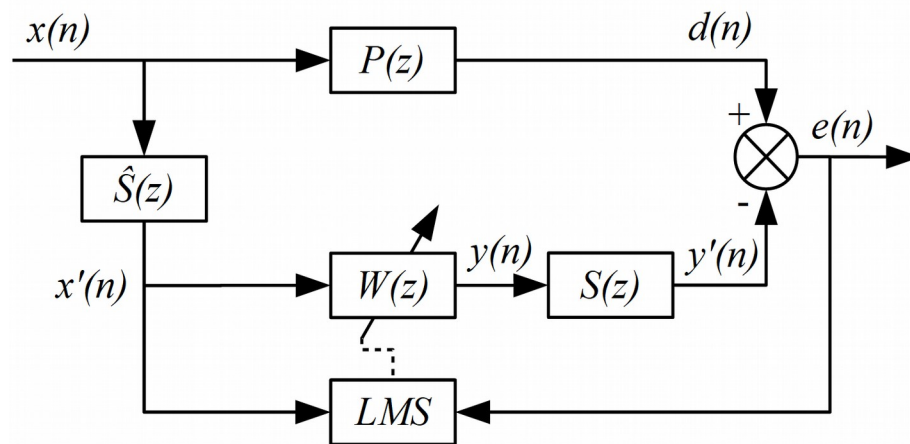


### 2.3.6. FxLMS com adaptação lenta

Uma alternativa possível ao algoritmo FxLMS é utilizar esta estrutura em sua forma chamada de FxLMS com adaptação lenta, caso o filtro adaptativo  $W(z)$  se modifique lentamente. (KUO; MORGAN, 1999)

Levando-se em conta que o algoritmo FxLMS fará o MSE convergir mesmo com um erro de fase de quase  $90^\circ$  entre  $\hat{S}(z)$  e  $S(z)$  dentro dos limites da adaptação lenta, pode-se dizer que a estrutura é bastante tolerante a erros na estimativa de  $\hat{S}(z)$  (KUO; MORGAN, 1999). A estrutura do FxLMS pode ser simplificada considerando  $\hat{S}(z)=S(z)$  para realizar a estrutura FxLMS com adaptação lenta, apresentada na Figura 8 (KUO; MORGAN, 1999).

Figura 8 - Diagrama de blocos do algoritmo FxLMS com adaptação lenta.



Fonte: Elaboração própria (2018).

É esperado que o sinal gerado pelo algoritmo seja capaz de agir destrutivamente no ruído indesejado captado pelo microfone de entrada. Este método traz resultados precisos se o fator de convergência  $\mu$  for escolhido com um valor/passos pequeno, entretanto isto significa uma convergência mais lenta (KUO; MORGAN, 1999).

### 2.3.7. FxNLMS

Assim como para o algoritmo LMS, existe uma forma normalizada do algoritmo FxLMS, que chama-se FxNLMS e que segue o mesmo princípio do NLMS. Seu objetivo também é melhorar a velocidade de minimização do MSE.

## 2.4. Microfones

Um microfone é um transdutor de pressão, composto por um diafragma e um transdutor de deslocamento que converte a energia das ondas acústicas em sinais elétricos (FRADEN, 2016).

Os microfones se diferenciam uns dos outros por algumas características importantes, tais como: a impedância de saída, a sensibilidade, a resposta em frequência, o ruído intrínseco e a diretividade (FRADEN, 2016).

### 2.4.1. Impedância de saída

A impedância de saída dos microfones pode ser classificada em três faixas: a baixa (50-1.000  $\Omega$ ), a média (5.000-15.000  $\Omega$ ) e a alta (>20.000  $\Omega$ ). Em razão disso, o meio que conecta o microfone ao amplificador, usualmente um cabo, não deve ultrapassar o comprimento de aproximadamente 6 metros para microfones com impedâncias de saída médias e altas, para não causar perdas em altas frequências e no nível do sinal. Para um microfone e um cabo de baixa impedância, pode-se utilizar praticamente qualquer comprimento que não acarretarão em perdas consideráveis (FRADEN, 2016).

### 2.4.2. Sensibilidade

A sensibilidade é a característica que revela o quão sensível é o microfone aos sinais acústicos com baixas amplitudes. Tipicamente, os fabricantes apresentam a medida da sensibilidade do microfone em dBV/Pa. Por exemplo, um microfone com sensibilidade de -55 dBV/Pa é melhor do que um de -60 dBV/Pa por entregar um sinal com maior amplitude ao amplificador (FRADEN, 2016).

### 2.4.3. Resposta em frequência

A resposta do microfone para diferentes frequências pode ser verificada em seu gráfico de resposta em frequência. Pode-se visualizar a faixa de frequências em que o microfone opera, assim como o efeito de variação frequencial sobre o ganho (FRADEN, 2016).

### 2.4.4. Ruído intrínseco

Para que um microfone opere da maneira desejada, deve-se obter uma boa razão sinal/ruído (SNR, do inglês *Signal-to-Noise Ratio*), ou seja, uma maior potência do sinal desejado em relação à potência do ruído de fundo. Logicamente, quanto menos ruído produzido intrinsecamente o microfone apresentar, melhor será a qualidade do sinal de saída do microfone (FRADEN, 2016).

O ruído intrínseco é medido em  $\text{dB(A)}_{\text{SPL}}$ , sendo A referente a uma curva de ponderação relativa à percepção humana. Um microfone considerado silencioso está na faixa dos  $20 \text{ dB(A)}_{\text{SPL}}$ , porém existem microfones super-silenciosos com ruído de  $0 \text{ dB(A)}_{\text{SPL}}$  (FRADEN, 2016).

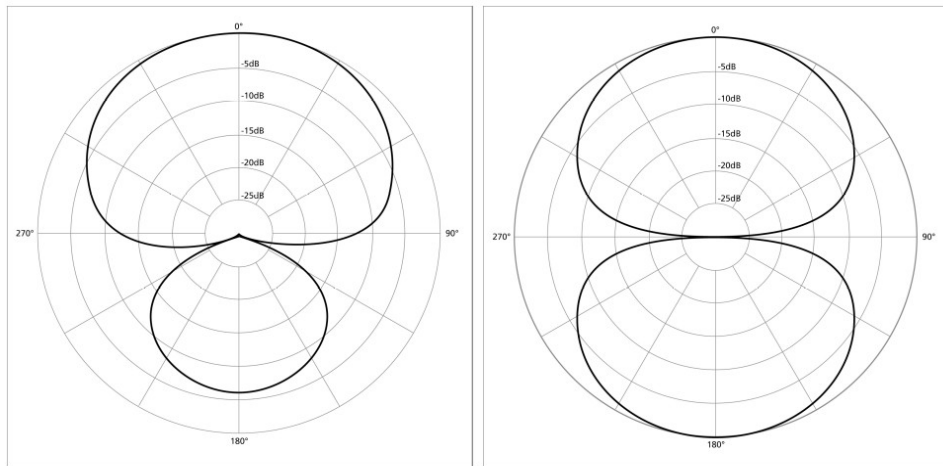
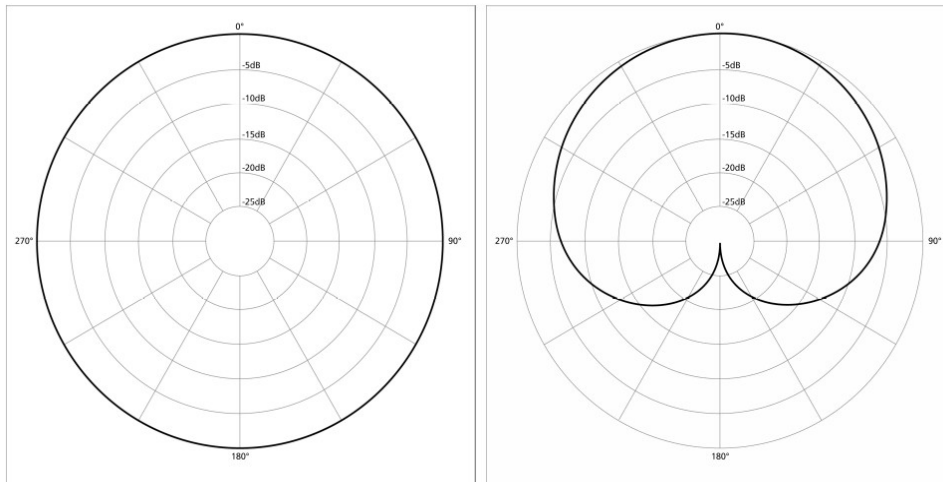
### 2.4.5. Diretividade

Os microfones são divididos em diversos tipos dependendo de sua diretividade, sendo os mais básicos: omnidirecional, cardióide, hípercardióide e bidirecional. Cada um é representado, geralmente, por um diagrama polar bidimensional onde o nível de sensibilidade para cada ângulo de incidência da onda sonora de entrada é relatado (FRADEN, 2016). A Figura 9 apresenta os exemplos básicos listados anteriormente.

Figura 9 - Diretividade para diferentes tipos de microfones.

*Omnidirecional*

*Cardióide*



*Hipercardióide*

*Bidirecional*

Fonte: Imagem adaptada de (GETDRAWINGS, 2018).

A razão dos microfones serem produzidos com diferentes diretividades é justamente para excluir sons vindos de fontes indesejadas em favor de sons de fontes da direção desejada (FRADEN, 2016).

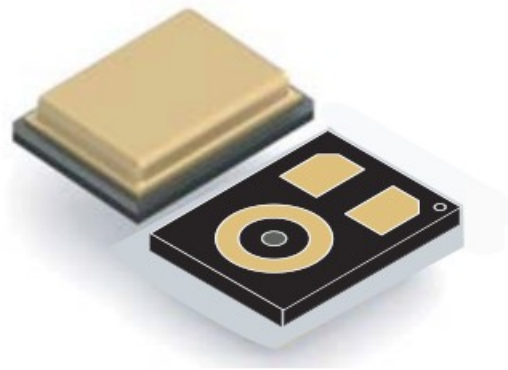
#### 2.4.6. Microfone MEMS

O transdutor escolhido para este projeto é produzido com a tecnologia MEMS (*Micro-Electrical-Mechanical Systems*), o qual é basicamente um capacitor de silício (ST, 2017).

O microfone MEMS funciona a partir das variações de capacitância entre uma placa fixa e uma móvel causadas por ondas sonoras (ST, 2017). O uso de microfones MEMS é indicado para todas as aplicações onde um tamanho pequeno, a qualidade elevada de som, a confiabilidade e um custo acessível são requisitos chave (ST, 2017), sendo precisamente o caso deste projeto.

O modelo escolhido para o projeto é o MP23AB02B, mostrado na Figura 10, da fabricante STMicroelectronics. As principais características do componente são: razão sinal-ruído de  $64 \text{ dB(A)}_{\text{SPL}}$ , tensão de alimentação entre  $1,6 \text{ V}$  e  $3,6 \text{ V}$ , sensibilidade entre  $-41 \text{ dBV/Pa}$  e  $-35 \text{ dBV/Pa}$ . Suas dimensões são de  $3,35 \text{ mm} \times 2,5 \text{ mm} \times 0,98 \text{ mm}$  (ST, 2014).

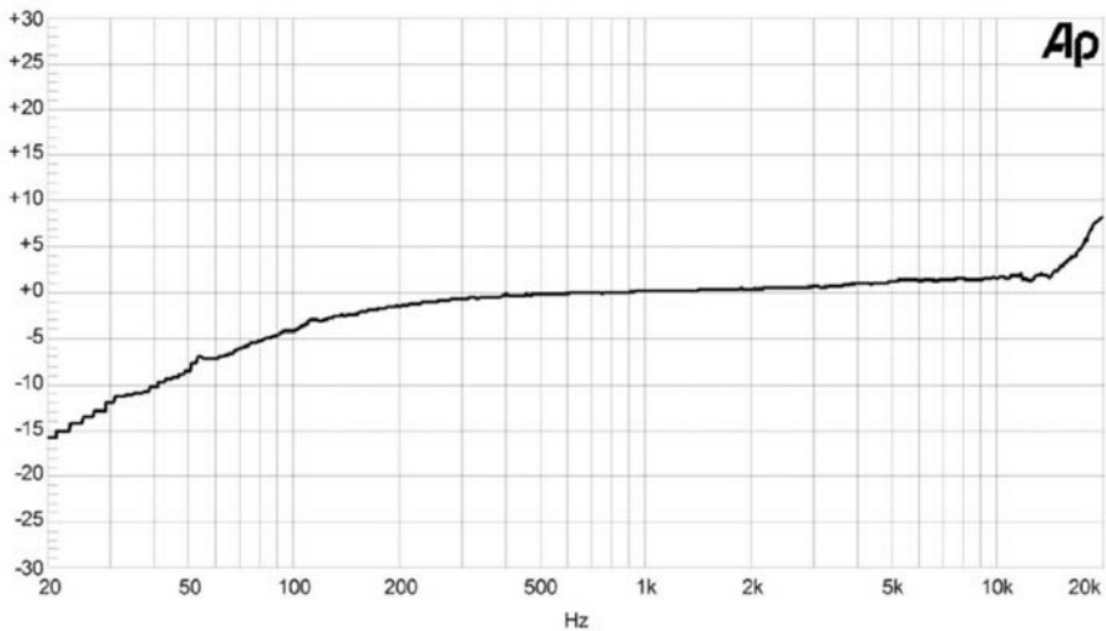
Figura 10 - Microfone MEMS MP23AB02B.



Fonte: (ST, 2014).

A resposta em frequência do microfone MP23AB02B é mostrada na Figura 11.

Figura 11 - Resposta em frequência do microfone MP23AB02B (ganho em dB x frequência em Hz).



Fonte: (ST, 2014).

Pode-se notar pelo gráfico da resposta em frequência que há uma grande atenuação do sinal do microfone em baixas frequências, até 150 Hz. Para garantir o bom funcionamento do projeto, é interessante que essa faixa de frequência seja amplificada.

## 2.5. Microcontroladores

Um microcontrolador é um sistema microprocessado composto por diversos periféricos, sendo assim ideal para realizar projetos de sistemas eletrônicos com relativa rapidez e facilidade (LIMA; VILLAÇA, 2012).

Destacam-se como exemplos as seguintes funcionalidades que podem ser encontradas no mercado de microcontroladores:

- a) terminais de propósito geral;
- b) conversores analógicos-digitais (A/D) e digitais-analógicos (D/A);
- c) interfaces de comunicação: USART, I2C, USB, Ethernet, SPI, dentre outras;
- d) saídas PWM;
- e) gerador interno de *clock*;

- f) memória SRAM, EEPROM e *flash*;
- g) temporizadores/contadores (LIMA; VILLAÇA, 2012).

### 2.5.1. Microprocessador

Um microprocessador é um circuito integrado (CI) programável, capaz de realizar diversas tarefas. Suas operações são realizadas pela Unidade Central de Processamento (CPU, do inglês *Central Processing Unit*), sincronizadas por um sinal de *clock*. Possui duas memórias, sendo uma a memória de programa, onde é armazenado o código a ser executado, e a outra a memória de dados, para fornecer informações de execução à CPU (LIMA; VILLAÇA, 2012).

Existem dois tipos de arquiteturas predominantes com relação ao barramento para comunicação de dados, a Von-Neumann e a Harvard. Já em relação ao conjunto de instruções, os processadores também são divididos em duas arquiteturas: Computadores com Conjunto Complexo de Instruções (CISC, do inglês *Complex Instructions Set Computers*) e Computadores com Conjunto Reduzido de Instruções (RISC, do inglês *Reduced Instructions Set Computers*). Ambas possuem semelhanças por utilizarem um conjunto simples de instruções, porém diferem no tamanho, complexidade e extensão. (LIMA; VILLAÇA, 2012).

### 2.5.2. Cortex-M4F

O Cortex-M4F é um microprocessador ARM de 32 bits com arquitetura RISC, de alto desempenho e baixo custo. Foi especificamente projetado para o mercado de sistemas embarcados com foco no processamento digital de sinais, portanto possui a vantagem de possibilitar o uso das instruções de DSP e operações em ponto flutuante, partes fundamentais para a escolha deste processador no projeto. Esta arquitetura suporta operações em ponto flutuante por *hardware* pela sua Unidade de Ponto Flutuante (FPU, do inglês *Floating Point Unit*) como, por exemplo, operações de multiplicação e de divisão, processadas consideravelmente mais rápido do que seriam realizadas se fossem baseadas em *software* (ARM, 2017) (PJRC, 2017c).

### 2.5.3. Teensy

Observa-se na Figura 12 o Teensy 3.6, um sistema de desenvolvimento microcontrolado de dimensões pequenas, que possui alta performance, larga memória e vários recursos, sendo os principais listados na Tabela 2. É uma plataforma de desenvolvimento adequada para uma ampla diversidade de projetos eletrônicos, programável por meio de sua porta *Universal Serial Bus* (USB). Vale destacar que seu processador é o ARM Cortex-M4F e seus pinos de I/O (do inglês *input/output*) só suportam tensões de até 3,3 V e correntes de até 10 mA (PJRC, 2017d).

Figura 12 - Teensy 3.6.



Fonte: (PJRC, 2017d).

O sistema também oferece como suporte uma interface gráfica *online*, diversos tutoriais, bibliotecas de programação e um *shield* especificamente projetado para projetos com áudio. A interface gráfica e o *shield*, que serão discutidos nas próximas seções, chamam-se *Audio System Design Tool* e *Audio Adaptor Board*. Para esclarecimento, um *shield* é uma placa de circuito impresso (PCI) que adiciona funcionalidades ao sistema de desenvolvimento, neste caso ao Teensy (PJRC, 2017a) (PJRC, 2017b).



Tabela 2 - Especificações do Teensy 3.6.

<b>Especificação</b>	<b>Teensy 3.6</b>
Processador	MK66FX1M0VMD18 Cortex-M4F 180 MHz 240 MHz ( <i>overclock</i> )
Memória Flash	1024 kB
Memória RAM	256 kB
EEPROM	4096 B
I/O	58 pinos 3,3 V 10 mA
Entradas Analógicas	25 pinos
PWM	22 pinos
UART	6
I2C	4
SPI	3
USB	2
Preço	US\$29,25

Fonte: PJRC (2017c).

#### 2.5.4. Audio Adaptor Board

O *Audio Adaptor Board* (Figura 13) é um *shield* para o Teensy que o permite gravar, modificar e reproduzir áudios com qualidade de 16 bits e taxa de amostragem de 44,1 kHz (qualidade de CD) (PJRC, 2017a).

Figura 13 - Audio Adaptor Board.



Fonte: (PJRC, 2017a).

As bibliotecas de áudio do Teensy, em conjunto com o processador Cortex-M4F, permitem ao projetista criar aplicações de áudio complexas, como por exemplo: aplicar efeitos, sintetizar formas de onda, reproduzir múltiplos arquivos de áudio e reproduzir áudio de alta qualidade para os fones de ouvido (PJRC, 2017a).

O *codec* de áudio SGTL5000 do *shield* é capaz de obter alta performance e funcionalidade no processamento de áudio estéreo, mantendo um baixo consumo de potência. Possui um amplificador integrado para fones de ouvido. Disponibiliza entradas e saídas de sinais de linha e digital com protocolo I<sup>2</sup>S, além de entrada de microfone e saída para fones de ouvido. (FREESCALE SEMICONDUCTOR, 2013)

### 3. DESENVOLVIMENTO DO SISTEMA

Com base na fundamentação teórica adquirida na pesquisa para o desenvolvimento deste trabalho, foi realizada a simulação e a implementação do algoritmo no Teensy. A implementação foi dividida em três etapas: primeiramente com o algoritmo sendo executado no microcontrolador sem elementos externos, então com o algoritmo sendo executado para o cancelamento de ruído dentro de tubos de PVC, uma estrutura tipicamente usada em testes de sistemas ANC (KUO *et al.*, 1996). A realização de cada etapa possibilita o melhor entendimento do algoritmo e aumenta as chances da próxima etapa ser realizada com sucesso.

Para o desenvolvimento da simulação e do código implementado em *hardware*, foram utilizados os programas MATLAB e Sloeber. A etapa de simulação, assim como a geração de gráficos, foi realizada no MATLAB, um programa que possibilita, entre outras coisas, a programação de rotinas algébricas de filtragem adaptativa. A implementação foi realizada no Teensy. Já o Sloeber é uma IDE (do inglês *Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado), originalmente desenvolvida como um *plugin* para a IDE Eclipse para ser uma alternativa mais versátil à IDE do Arduino. Como o Teensy utiliza da IDE do Arduino para sua programação, também é possível programá-lo via Sloeber, o que facilita no desenvolvimento e gerenciamento de projetos complexos

Os códigos modificados e criados para o projeto estão disponibilizados no Apêndice A.

A ferramenta *Audio System Design Tool* foi utilizada para realizar as conexões de entradas e saídas dos blocos de áudio no Teensy.

Finalmente, o *software* Ocenaudio e um decibelímetro foram utilizados para monitorar a saída de áudio do sistema e avaliar seu funcionamento.

#### 3.1. Simulação no MATLAB

Com base nas equações e nos diagramas de blocos obtidos da literatura e apresentadas na seção de Fundamentação Teórica, foi realizada a simulação do

caminho de propagação secundário  $S(z)$  por meio do algoritmo NLMS com o auxílio do *software* MATLAB e posteriormente o cancelamento ativo de ruído pelo algoritmo FxNLMS. O código executado foi obtido e modificado de OEY (2010).

Foram desenvolvidas seis situações de testes, sendo elas divididas em dois grupos, uma com coeficientes conhecidos, de tamanho menor, e outra com coeficientes aleatórios, de tamanho maior:

Grupo 1) Simulações com coeficientes conhecidos de um filtro passa-baixas para  $S(z)$  e  $P(z)=2S(z)$  ( $N=56$ ):

- a) NLMS
- b) FxNLMS com senoide (400 Hz)
- c) FxNLMS com ruído branco

Grupo 2) Simulações com coeficientes de filtros gerados aleatoriamente tanto para  $S(z)$  quanto para  $P(z)$  ( $N=128$ ):

- a) NLMS
- b) FxNLMS com senoide (400 Hz)
- c) FxNLMS com ruído branco

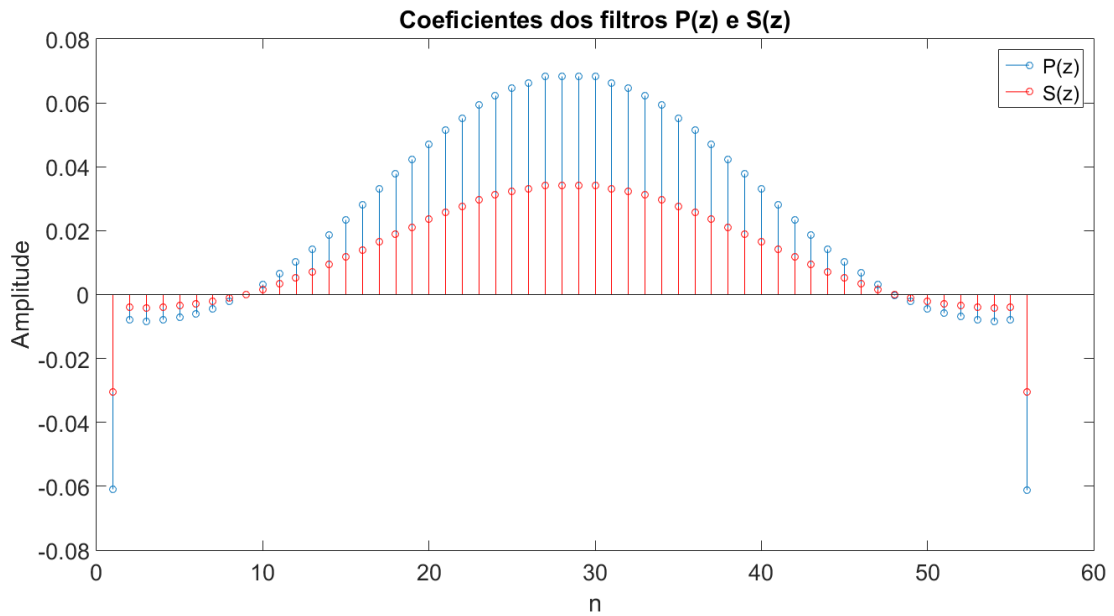
A ideia destas situações de testes é poder compará-las com os resultados obtidos na prática e avaliar se a abordagem está correta. As seis situações representam diferentes graus de complexidade.

### 3.1.1. Simulação da estimativa do caminho de propagação secundário (NLMS)

Primeiramente, como não se sabe qual é a função de transferência do caminho primário e nem do caminho secundário, na realidade, definem-se valores arbitrários. Como especificado anteriormente, teremos duas categorias de filtros a serem testadas, uma com um filtro  $S(z)$  passa-baixas e outra com  $P(z)=2S(z)$ , sendo  $N=56$ . Esta categoria foi desenvolvida com testes de baixa complexidade como uma forma de avaliar o funcionamento básico da implementação do algoritmo e como base para testes posteriores no microcontrolador. A outra categoria de testes foi desenvolvida com coeficientes de filtros gerados aleatoriamente para  $S(z)$  e para  $P(z)$  de tamanho  $N=128$ . Esta categoria foi desenvolvida com testes de maior complexidade como uma forma de avaliação mais rigorosa da implementação do algoritmo e, também, como simulação de cenários práticos mais complexos que

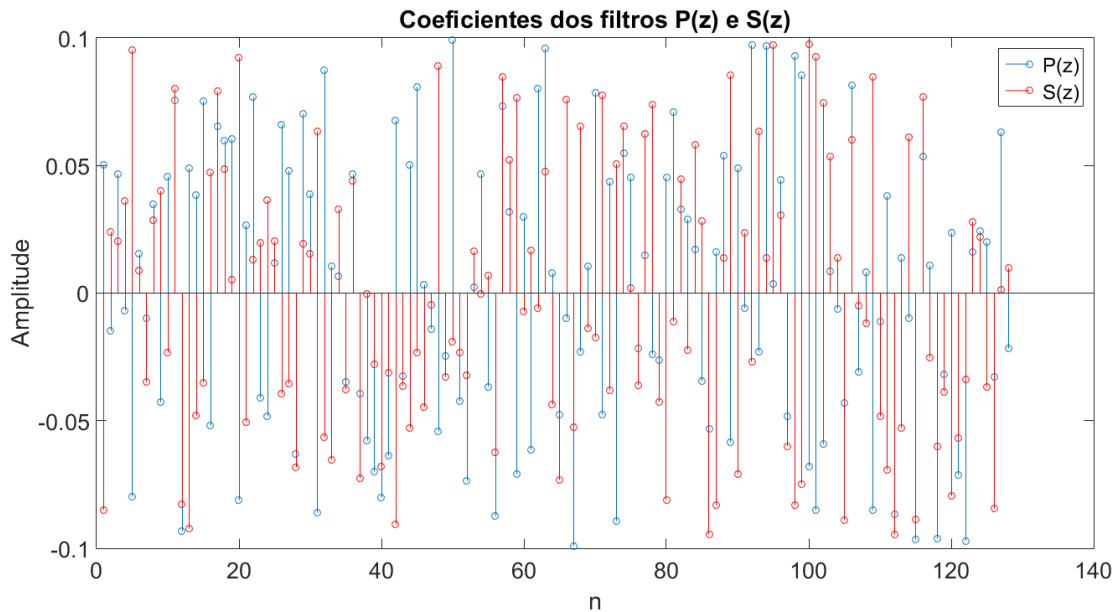
podem surgir na implementação com o microcontrolador. Assim, nas Figuras 14 e 15 estão representados os coeficientes de  $S(z)$  e  $P(z)$  que deverão ser estimados para as categorias de teste 1 e 2, respectivamente.

Figura 14 - Coeficientes de  $S(z)$  e  $P(z)$  utilizados no primeiro grupo de testes.



Fonte: Elaboração própria (2018).

Figura 15 - Coeficientes de filtros gerados aleatoriamente para  $S(z)$  e para  $P(z)$  utilizados no segundo grupo de testes.



Fonte: Elaboração própria (2018).

O sinal de ruído branco  $x(n)$  é gerado por meio de uma função para gerar valores aleatórios (com distribuição normal), com espectro de potência constante para todas as frequências até a frequência de Nyquist, de 22,05 kHz, contendo  $T=1280$  amostras. Este sinal é filtrado por  $S(z)$ , para a obtenção de  $d(n)$ . O número de amostras  $T$  foi definido a partir do número de amostras por pacote de áudio utilizado no microcontrolador (128), multiplicado por dez, para que se possa observar a adaptação do algoritmo.

Considera-se um valor para o coeficiente de convergência  $\mu$ , para ambos os casos, que não seja nem muito grande (para não desestabilizar o sistema), nem muito pequeno (para que seja possível observar um bom resultado na simulação).

Inicia-se um *loop* contendo as equações do algoritmo NLMS previamente descritas para percorrer todas as amostras do sinal de entrada  $x(n)$ . A cada iteração, o estado do filtro  $\hat{S}(z)$  é atualizado com a nova amostra de  $x(n)$  e sua saída  $y(n)$  é calculada. Subtraindo o novo valor da saída do filtro do valor desejado, resulta-se no erro do sistema  $e(n)$ . Com isso, é possível a utilização da equação de atualização dos coeficientes do filtro  $\hat{S}(z)$ . Por último, o MSE é

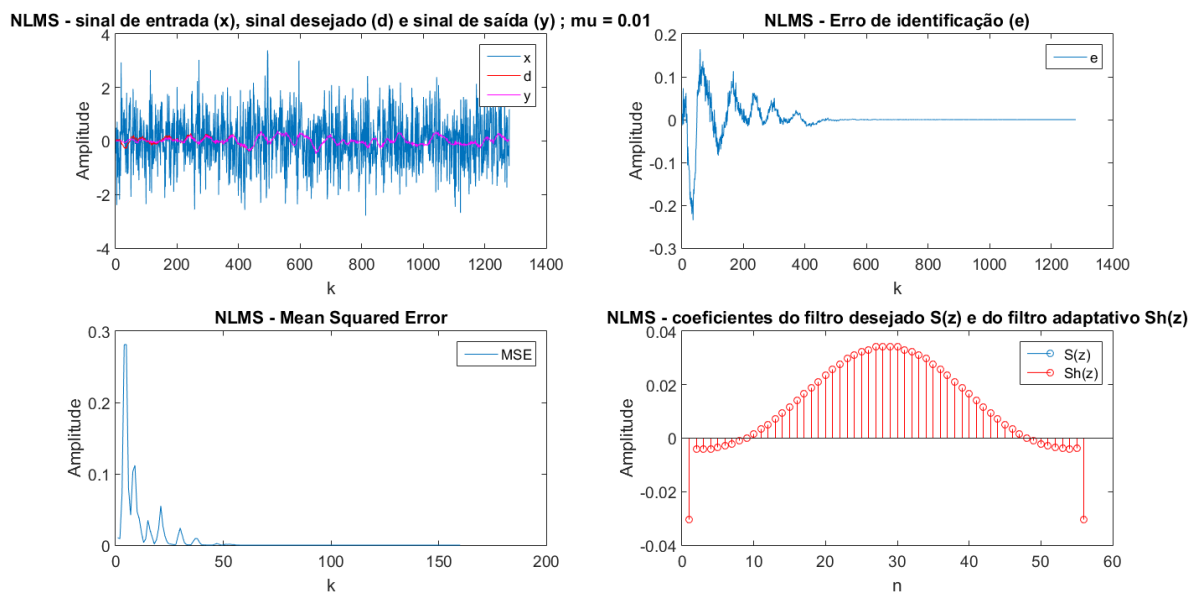
calculado a cada 128 amostras e registrado para que seja possível validar e observar a adaptação do algoritmo.

No final da simulação, gera-se para análise uma série de gráficos referentes a cada grupo de teste. O primeiro gráfico apresenta o sinal de entrada  $x(n)$ , o sinal desejado  $d(n)$  e o sinal de saída  $y(n)$ . O segundo gráfico apresenta a evolução do sinal de erro  $e(n)$  ao longo do tempo. O terceiro gráfico mostra a evolução do MSE. O quarto gráfico traz um comparativo com os coeficientes do filtro  $S(z)$  desejados e os coeficientes estimados de  $\hat{S}(z)$ . Para fins de simulação, o MSE é estimado a cada 8 amostras

$$MSE = \frac{\sum_{n=0}^7 e^2(n)}{8}. \quad (10)$$

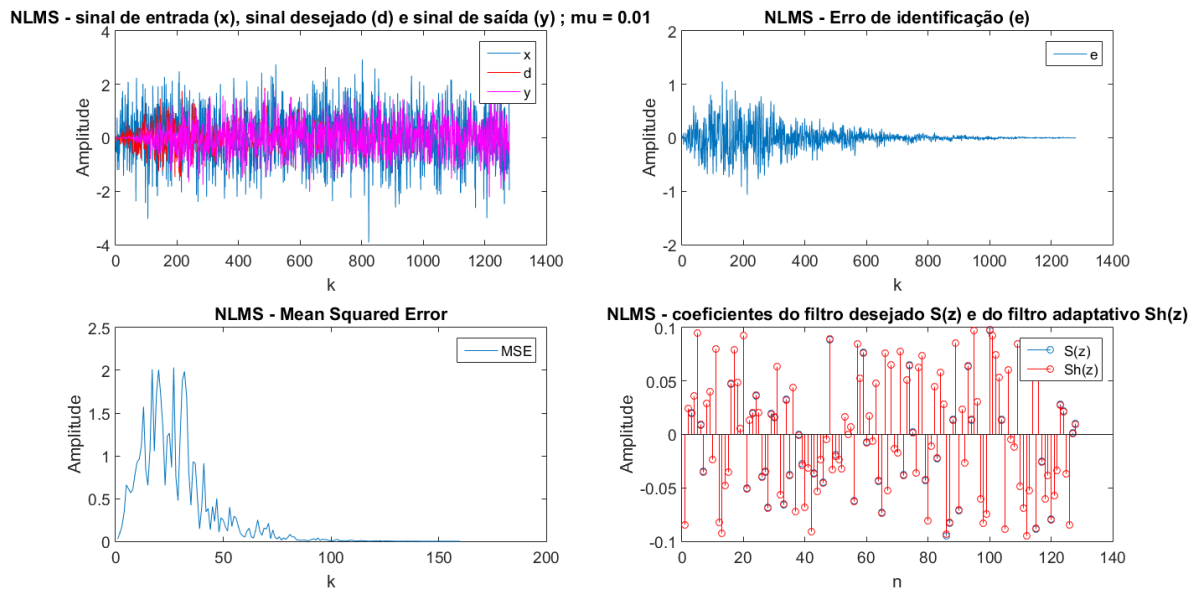
Os gráficos gerados para os testes 1.a e 2.a são mostrados nas Figuras 16 e 17, a seguir

Figura 16 - Simulação do algoritmo NLMS no MATLAB para o teste 1.a.



Fonte: Elaboração própria (2018).

Figura 17 - Simulação do algoritmo NLMS no MATLAB para o teste 2.a.



Fonte: Elaboração própria (2018).

Pode-se perceber por meio da análise dos gráficos que os coeficientes do filtro foram corretamente adaptados para minimizar o sinal de erro. O valor de  $\mu$ , o tamanho dos filtros e a correlação entre os coeficientes  $S(z)$  e  $\hat{S}(z)$ , para cada caso de teste, podem ser conferido na Tabela 3.

Tabela 3 - Tabela comparativa de valores para cada caso de teste do NLMS.

Teste do NLMS	Coefficiente de convergência ( $\mu$ )	Tamanho dos filtros $P(z)$ e $S(z)$	Correlação entre $S(z)$ e $\hat{S}(z)$
Teste 1.a	0,01	56	100%
Teste 2.a	0,01	128	99%

Fonte: Elaboração própria (2018).



### 3.1.2. Simulação do algoritmo de cancelamento ativo de ruído (FxNLMS)

A próxima etapa é a simulação do algoritmo FxNLMS. Como é uma continuação dos casos de testes especificados anteriormente, utilizam-se os mesmos  $P(z)$  e  $S(z)$  gerados no passo anterior, assim como o número de amostras  $T=1280$ . Para acompanhamento do processo, conferir a Figura 7 da Seção 2.3.5.

Buscou-se empiricamente o melhor fator de convergência  $\mu$ , para ambos os casos, assim como na etapa anterior.

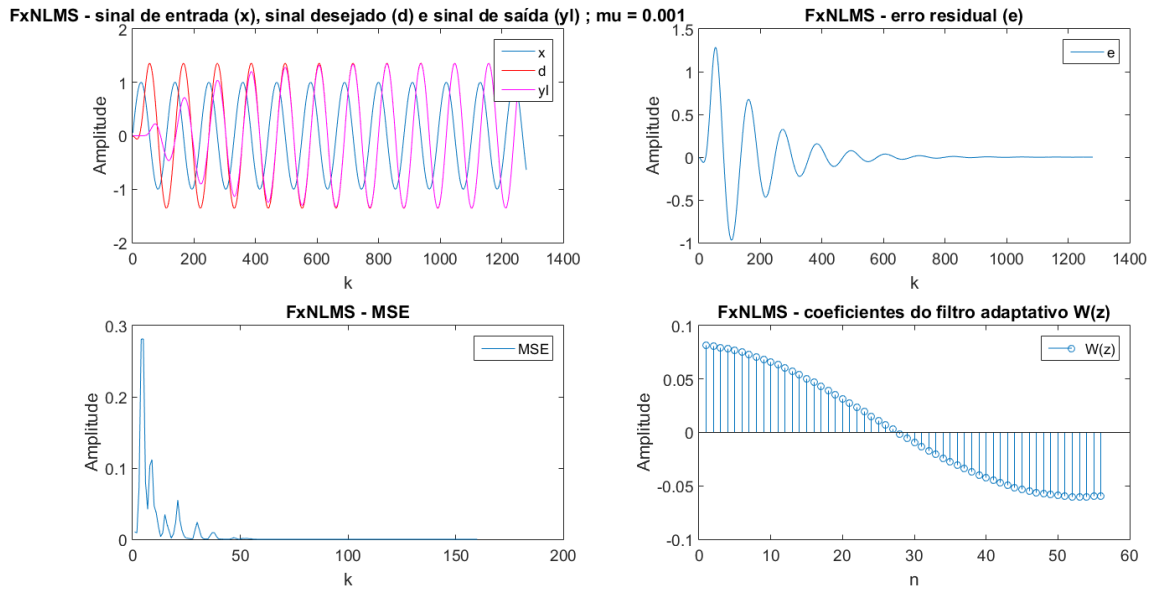
Serão feitos dois testes nesta etapa para cada caso, um com o sinal de entrada  $x(n)$  sendo uma onda senoidal de 400 Hz e o outro com  $x(n)$  sendo um ruído branco. Na prática estes sinais virão de uma fonte externa e serão recebidos pelo microfone de referência.

Deve-se filtrar  $x(n)$  pelo caminho primário  $P(z)$  para se obter  $d(n)$ , dado que o atraso causado pelo caminho secundário  $S(z)$  será compensado neste algoritmo por  $\hat{S}(z)$ . O sinal de entrada será filtrado por  $\hat{S}(z)$  e entrará no algoritmo como  $x'(n)$ .

O sinal de saída do controlador,  $y(n)$ , será propagado pelo caminho secundário e passará a ser chamado de  $y'(n)$ . Será feita então a subtração de  $d(n)$  e  $y'(n)$  para se obter o sinal de erro residual  $e(n)$  (na prática capturado pelo microfone de erro). O *loop* percorrerá todas as amostras do sinal de entrada e adaptará os coeficientes do filtro com as equações do algoritmo FxNLMS.

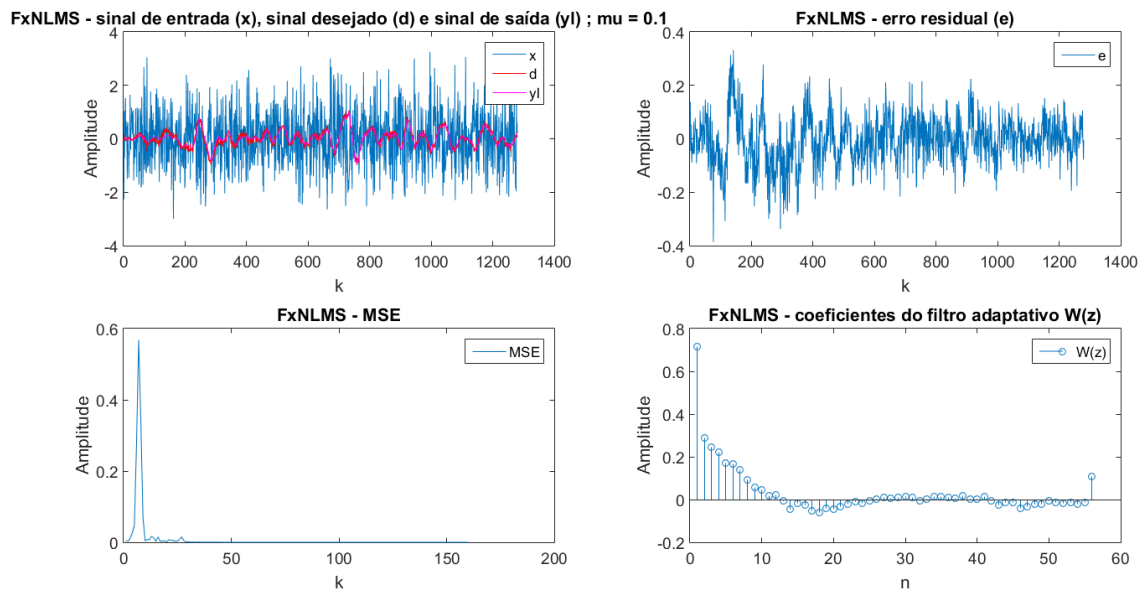
Novamente, geram-se para análise uma série de gráficos, referentes a cada grupo de testes. O primeiro gráfico apresenta o sinal de entrada  $x(n)$ , o sinal desejado  $d(n)$  e o sinal de saída  $y'(n)$ . O segundo gráfico apresenta a evolução do sinal de erro  $e(n)$  ao longo do tempo. O terceiro gráfico mostra a evolução do MSE. O quarto gráfico traz um comparativo com os coeficientes do filtro  $S(z)$  desejados e com os coeficientes estimados de  $\hat{S}(z)$ . A seguir, as Figuras 18, 19, 20 e 21 se referem aos testes do FxNLMS de 1.b, 1.c, 2.b e 2.c, respectivamente:

Figura 18 - Simulação do algoritmo FxNLMS no MATLAB para o teste 1.b.



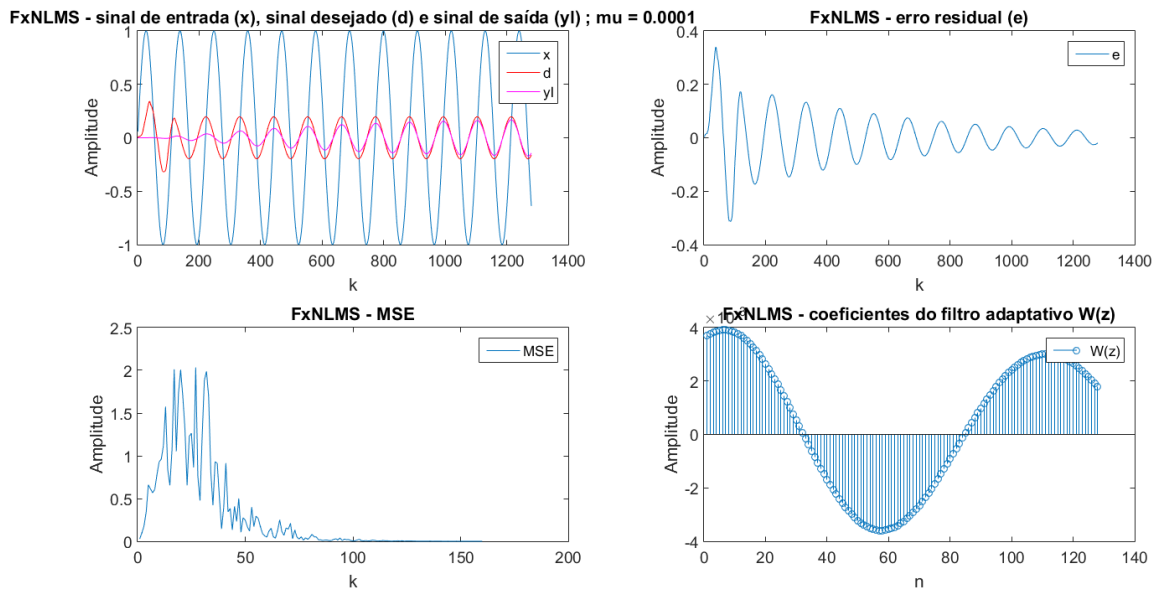
Fonte: Elaboração própria (2018).

Figura 19 - Simulação do algoritmo FxNLMS no MATLAB para o teste 1.c.



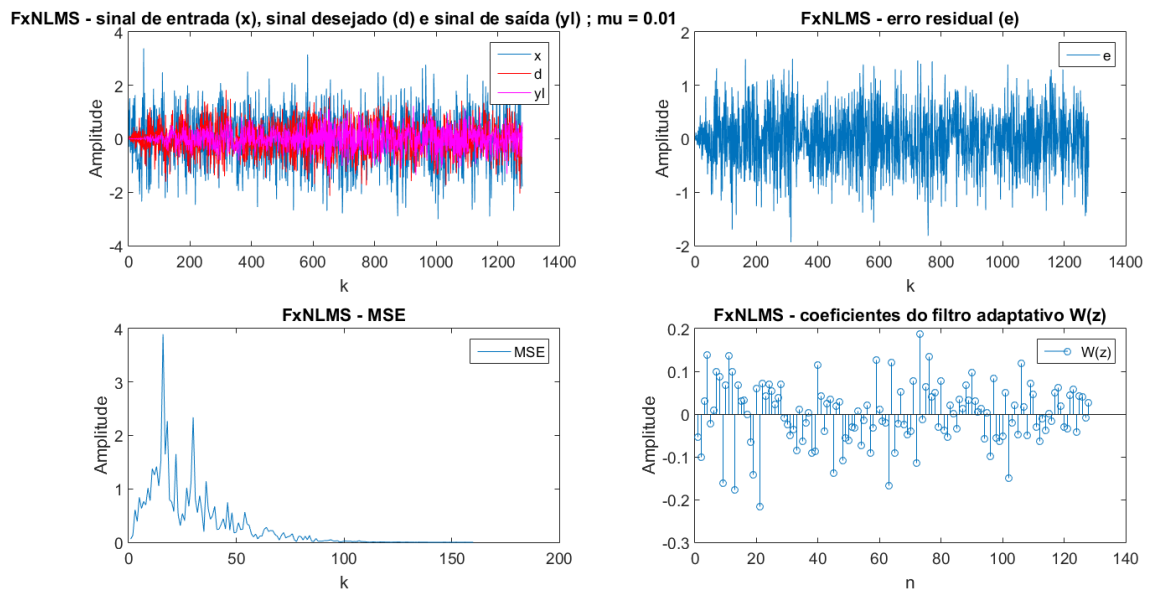
Fonte: Elaboração própria (2018).

Figura 20 - Simulação do algoritmo FxNLMS no MATLAB para o teste 2.b.



Fonte: Elaboração própria (2018).

Figura 21 - Simulação do algoritmo FxNLMS no MATLAB para o teste 2.c.



Fonte: Elaboração própria (2018).

Comparando os testes, pode ser observado que o algoritmo apresenta maior dificuldade para reduzir a potência do sinal de erro se o sinal de entrada for um ruído branco, mas, mesmo assim, ele atinge seu objetivo de minimizar o MSE.

É importante ressaltar que o valor de  $\mu$  foi ajustado individualmente para que, em cada simulação, seja possível observar o algoritmo funcionando na pequena janela de amostras.

Pode-se perceber por meio da análise dos gráficos que os coeficientes do filtro foram corretamente adaptados para minimizar o sinal de erro. O valor de  $\mu$  e o tamanho dos filtros para cada caso de teste pode ser conferido na Tabela 4.

Tabela 4 - Tabela comparativa de valores para cada caso de teste do FxNLMS.

Teste do FxNLMS	Coefficiente de convergência ( $\mu$ )	Tamanho dos filtros $P(z)$ e $S(z)$
Teste 1.b	0,001	56
Teste 1.c	0,1	56
Teste 2.b	0,0001	128
Teste 2.c	0,01	128

Fonte: Elaboração própria (2018).

## 3.2. Implementação do sistema

### 3.2.1. Implementação do código no Teensy sem elementos externos

Como a simulação no MATLAB obteve os resultados esperados, pôde-se passar para a implementação em *hardware* no Teensy por meio da IDE do Sloeber. A programação do Teensy foi feita toda em C e C++. Entretanto, antes de implementar o sistema completo, foram realizados testes para avaliar se o código no microcontrolador estava funcionando corretamente sem os elementos externos (microfones, alto-falantes, etc...). Isto foi realizado para avaliar a possibilidade de implementação do sistema ANC no Teensy sem que fontes externas de erro pudessem dificultar esta avaliação.

Assim como na etapa de simulação, foram realizados dois grupos de testes para diferentes tamanhos de filtro, cada um com suas etapas divididas como mostrado abaixo.

Grupo 3) Implementação do código no Teensy sem elementos externos com coeficientes de filtros passa-baixas para  $S(z)$  e  $P(z)=2S(z)$  ( $N=56$ ):

- a) NLMS
- b) FxNLMS com senoide (400 Hz)
- c) FxNLMS com ruído branco filtrado por um filtro passa-baixas (1,5 kHz)
- d) FxNLMS com ruído branco

Grupo 4) Implementação do código no Teensy sem elementos externos com coeficientes de um filtro gerado aleatoriamente para  $S(z)$  e para  $P(z)$  ( $N=128$ ):

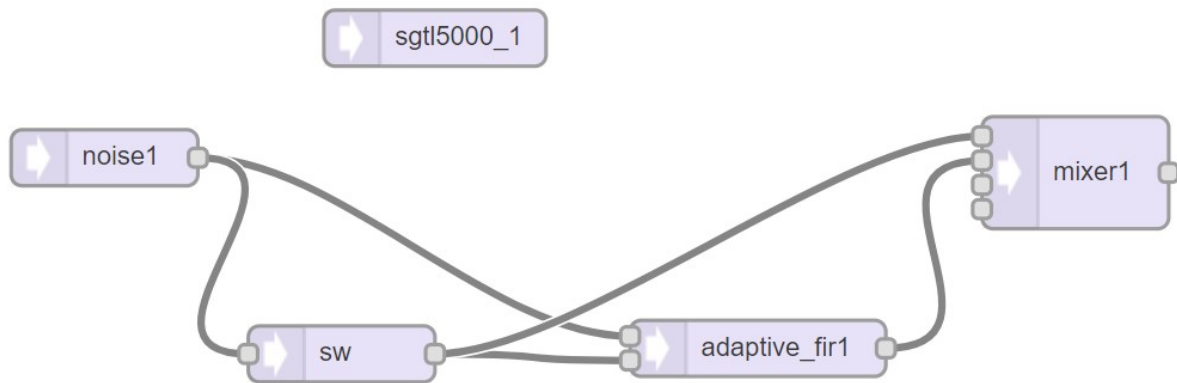
- a) NLMS
- b) FxNLMS com senoide (400 Hz)
- c) FxNLMS com ruído branco filtrado por um filtro passa-baixas (1,5 kHz)
- d) FxNLMS com ruído branco

Destaca-se o surgimento de mais um teste em relação à simulação feita anteriormente, com o ruído branco de entrada filtrado por um passa-baixas. Isso se deve por conta das limitações que a implementação em um sistema real possui e que não são vistas na simulação do MATLAB, como perdas de sinal acústico para o ambiente, limitações de geração de áudio pelos alto-falantes, limitações de captura de sinais pelos microfones.

A primeira tarefa é a construção do diagrama de blocos na ferramenta *Audio System Design Tool* para as conexões de áudio do projeto. Teremos dois diagramas de blocos distintos, um para o NLMS e outro para o FxNLMS.

O diagrama de blocos do NLMS é mostrado na Figura 22.

Figura 22 - Diagrama de blocos do NLMS para os grupos testes 3 e 4.

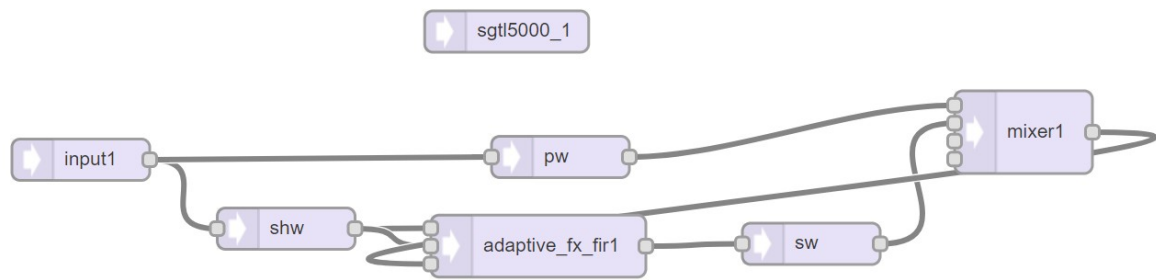


Fonte: Elaboração própria (2018).

O bloco *noise1* representa o ruído branco de entrada do sistema, enquanto que o bloco *sw* representa o filtro  $S(z)$ . O *adaptive\_fir1* foi um bloco de filtro criado pelo autor usando HTML e C++ para representar  $\hat{S}(z)$ , manipular os sinais e realizar a chamada da função *arm\_lms\_norm\_f32()* da biblioteca CMSIS-DSP. O *mixer1* representa o espaço acústico onde os sinais  $d(n)$  e  $y(n)$  se encontrarão, portanto esse bloco deve subtrair estes sinais. Por último, o bloco *sgtl5000\_1* controla o *codec* de áudio SGTL5000 do *Audio Adaptor Board*, que possui diversas funções importantes para o projeto, como por exemplo: controle de volume, configurações de entradas e saídas de sinais de áudio, controle da sensibilidade de captação, etc.

O diagrama de blocos do FxNLMS é mostrado na Figura 23.

Figura 23 - Diagrama de blocos do FxNLMS para os grupos testes 3 e 4.



Fonte: Elaboração própria (2018).

O bloco *input1* representa o ruído de entrada do sistema. Os blocos *pw*, *shw* e *sw* são os filtros  $P(z)$ ,  $\hat{S}(z)$  e  $S(z)$ . O *adaptive\_fx\_fir1* é um bloco de filtro criado pelo autor para representar  $W(z)$ , manipular os sinais e realizar a chamada da função *arm\_fxlms\_norm\_f32()*, que fará os cálculos do algoritmo FxNLMS. Esta função foi criada tendo como base a função *arm\_lms\_norm\_f32()* da biblioteca CMSIS-DSP. O *mixer1* representa o espaço acústico onde os sinais  $d(n)$  e  $y'(n)$  se encontram, portanto esse bloco deve subtrair estes sinais. Por último, o bloco *sgtl5000\_1* controla o chip SGTL5000 do *Audio Adaptor Board*, como explicado anteriormente.

A biblioteca CMSIS-DSP foi utilizada principalmente porque suas funções são otimizadas para o processador utilizado.

Para ambos os diagramas, os tipos de dados utilizados para os blocos dos filtros são diferentes dos blocos dos filtros adaptativos. Os primeiros blocos utilizam o formato *q\_15*, enquanto os segundos *float32\_t* para os coeficientes do filtro e para os sinais. O processamento dos dados no formato *float32\_t* foi realizado para não causar possíveis problemas de *scaling* e *overflow* encontrados quando utiliza-se o formato *q\_15*. Portanto, foi necessário realizar a conversão de *q\_15* para *float32\_t*, utilizando-se funções otimizadas da biblioteca CMSIS-DSP.

Foram criadas também classes em C++ para salvar o último pacote de áudio para os sinais relevantes do NLMS ( $x(n)$ ,  $d(n)$ ,  $y(n)$  e  $e(n)$ ) e do FxNLMS ( $x(n)$ ,  $x'(n)$ ,  $d(n)$ ,  $y(n)$  e  $e(n)$ ). Como só foi possível salvar o último

pacote, os gráficos gerados não serão os ideais para se fazer uma boa análise do algoritmo, mesmo assim, é possível saber se o algoritmo está funcionando.

Após a construção de cada diagrama de blocos, eles foram convertidos em linhas de código e importados nos seus respectivos projetos do Sloeber, onde foram configurados.

Como este algoritmo exige muito da memória e do processador, o chip foi configurado para salvar temporariamente até 200 pacotes de 128 amostras de áudio, o que significa aproximadamente 580 ms de áudio. O uso do processador será monitorado para os diferentes tamanhos de filtros.

Levando-se em consideração a frequência de amostragem de 44,1 kHz do microcontrolador, o ruído branco gerado pertence à faixa frequencial de 0 Hz até 22,05 kHz.

Para fins práticos, o MSE é estimado a cada pacote de áudio

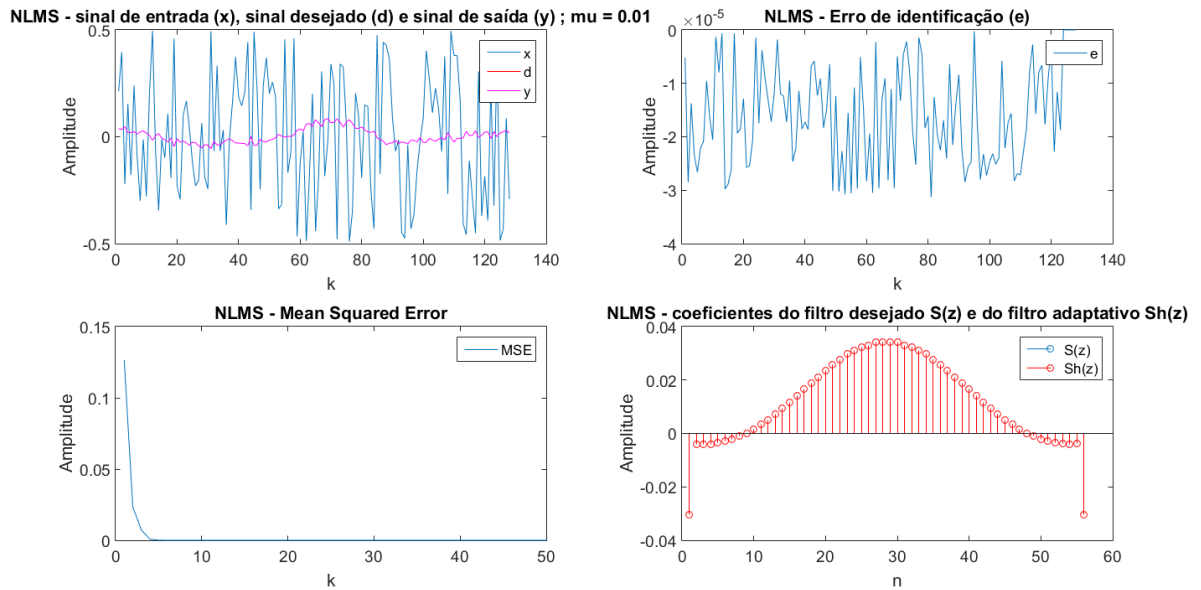
$$MSE = \frac{\sum_{n=0}^{127} e^2(n)}{128}. \quad (11)$$

A compilação e execução do código é enfim realizada no microcontrolador. Depois de alguns segundos os resultados são recebidos e exportados para o MATLAB, que os representará graficamente.

Para o NLMS, os resultados obtidos dos testes 3.a e 4.a são mostradas nas Figuras 24 e 25.

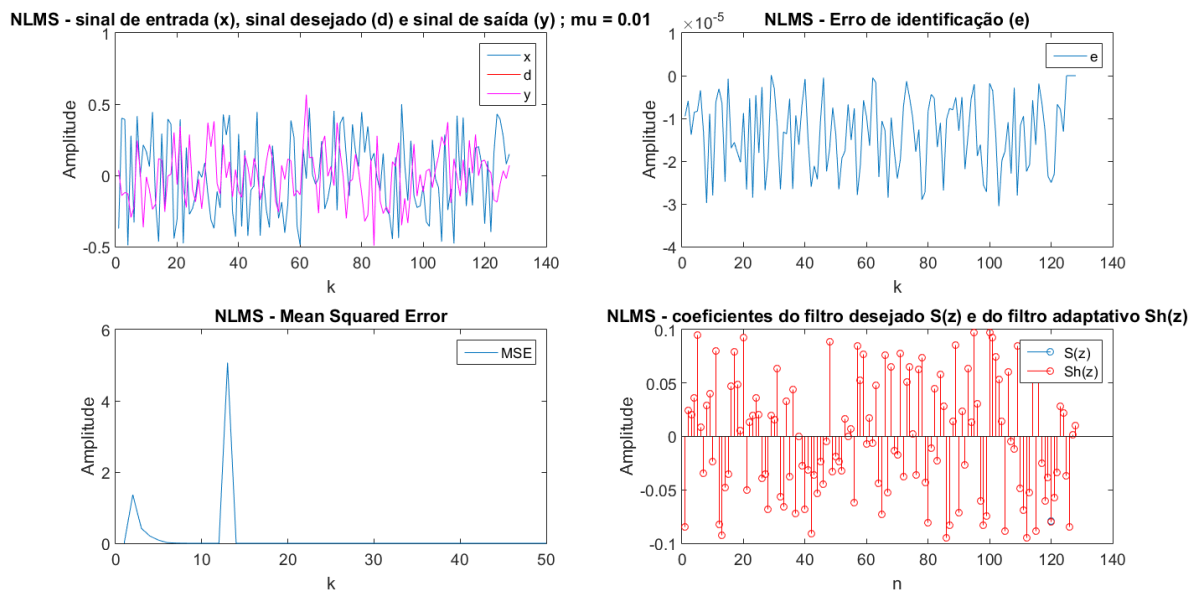


Figura 24 - Resultado da execução do algoritmo NLMS no Teensy sem elementos externos para o teste 3.a.



Fonte: Elaboração própria (2018).

Figura 25 - Resultado da execução do algoritmo NLMS no Teensy sem elementos externos para o teste 4.a.



Fonte: Elaboração própria (2018).

As Tabelas 5 e 6 fazem a comparação entre os valores obtidos nas execuções do código no Teensy:

Tabela 5 - Tabela comparativa de valores para cada caso de teste do NLMS.

<b>Teste do NLMS</b>	<b>Coefficiente de convergência (<math>\mu</math>)</b>	<b>Tamanho dos filtros <math>P(z)</math> e <math>S(z)</math></b>	<b>Correlação entre <math>S(z)</math> e <math>\hat{S}(z)</math></b>
Teste 3.a	0,01	56	100%
Teste 4.a	0,01	128	100%

Fonte: Elaboração própria (2018).

Tabela 6 - Tabela comparativa do uso de recursos do Teensy para cada caso de teste do NLMS.

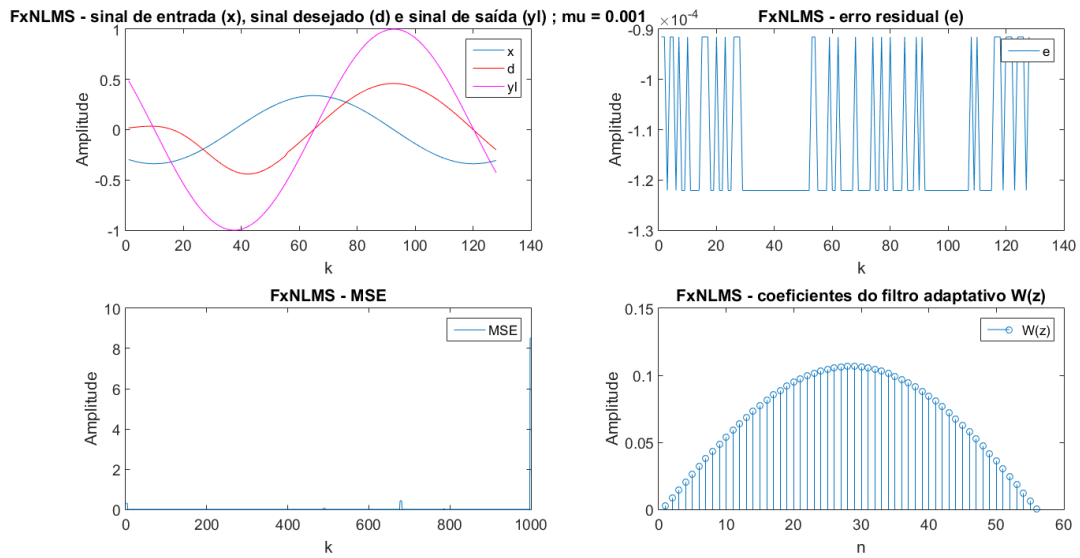
<b>Teste do NLMS</b>	<b>Número máximo de pacotes de áudio utilizados</b>	<b>Porcentagem máxima de processamento</b>
Teste 3.a	4	40,9%
Teste 4.a	6	79,0%

Fonte: Elaboração própria (2018).

Analisando os resultados, conclui-se que é possível realizar o algoritmo NLMS no Teensy com número de coeficientes igual a 128. Seria possível ainda aumentar o número de coeficientes, porém, também seria necessário manter uma certa margem no uso máximo de processamento para não ocorrerem problemas.

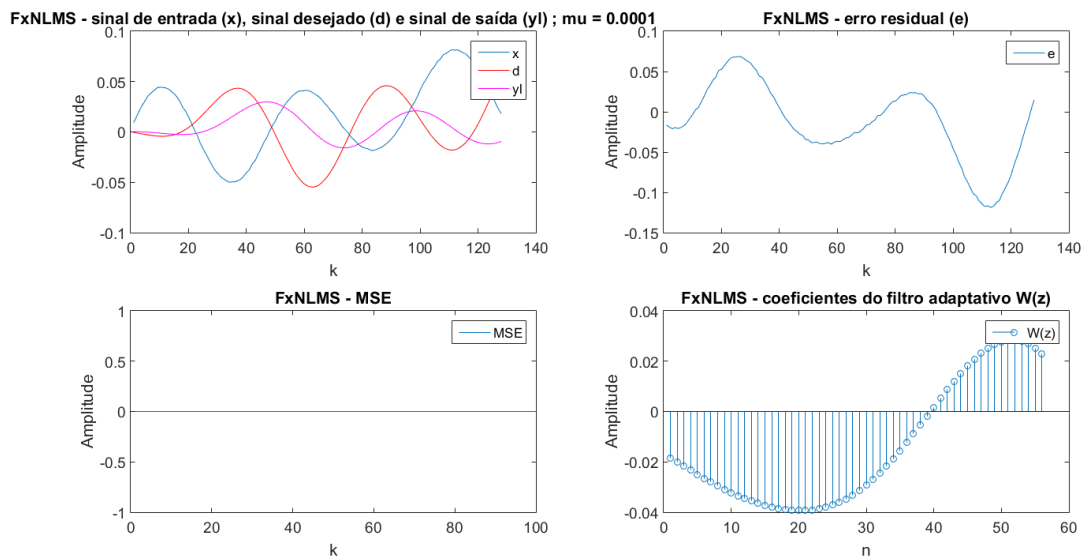
Os testes 3.b, 3.c e 3.d apresentaram os resultados das Figuras 26, 27 e 28, respectivamente:

Figura 26 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 3.b.



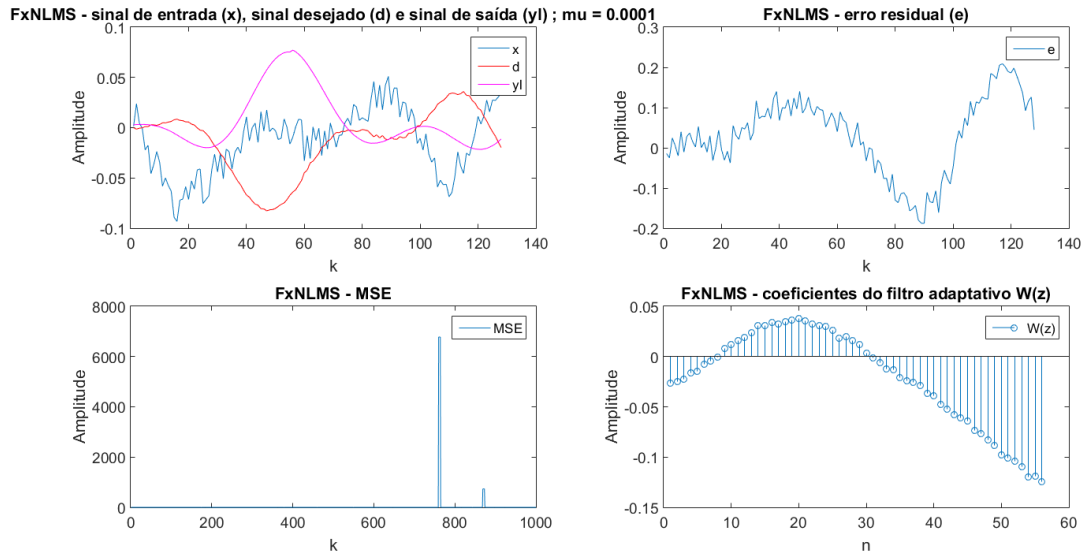
Fonte: Elaboração própria (2018).

Figura 27 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 3.c.



Fonte: Elaboração própria (2018).

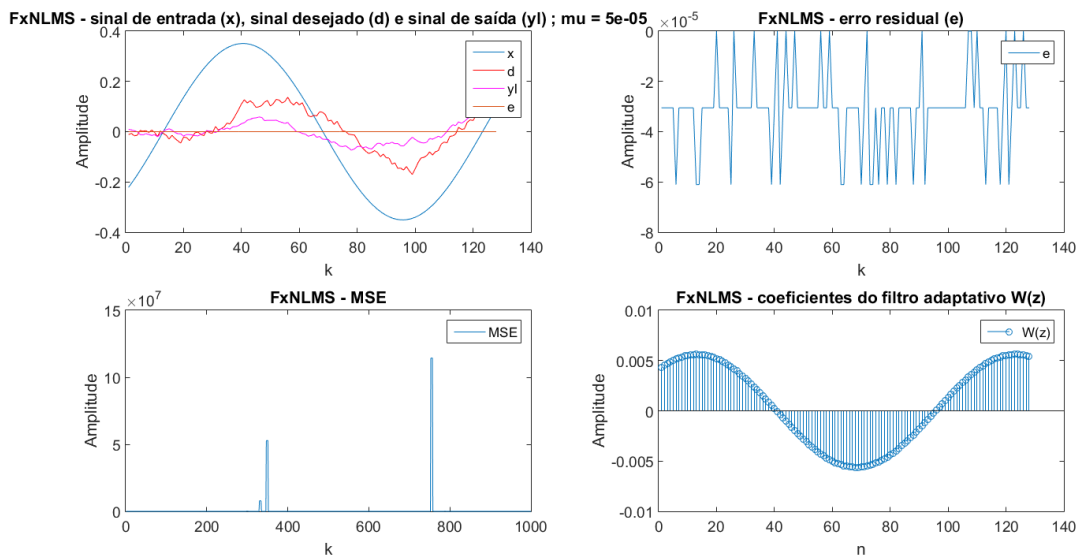
Figura 28 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 3.d.



Fonte: Elaboração própria (2018).

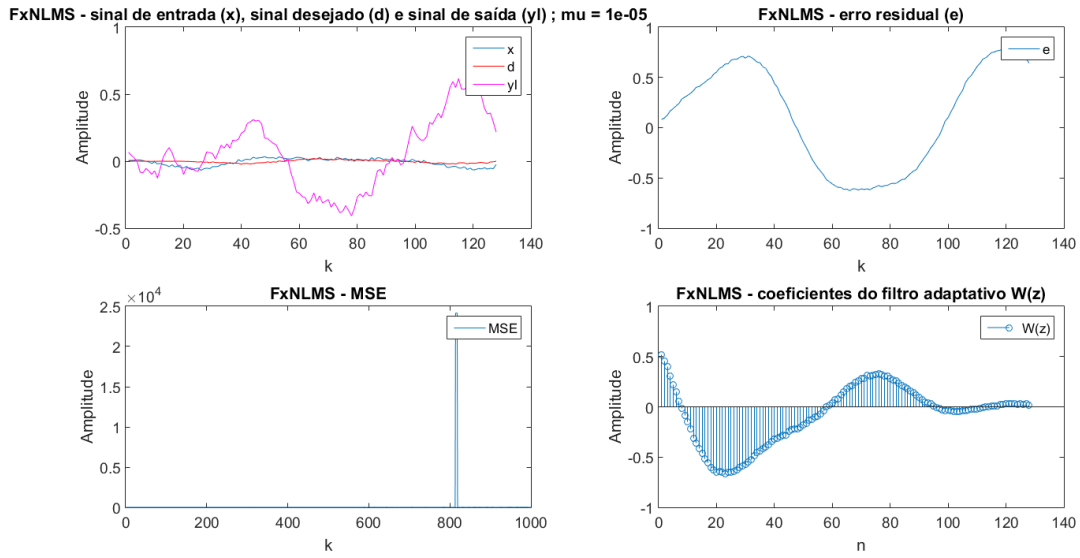
Já os testes 4.b, 4.c e 4.d resultaram nos gráficos das Figuras 29, 30 e 31, respectivamente:

Figura 29 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 4.b.



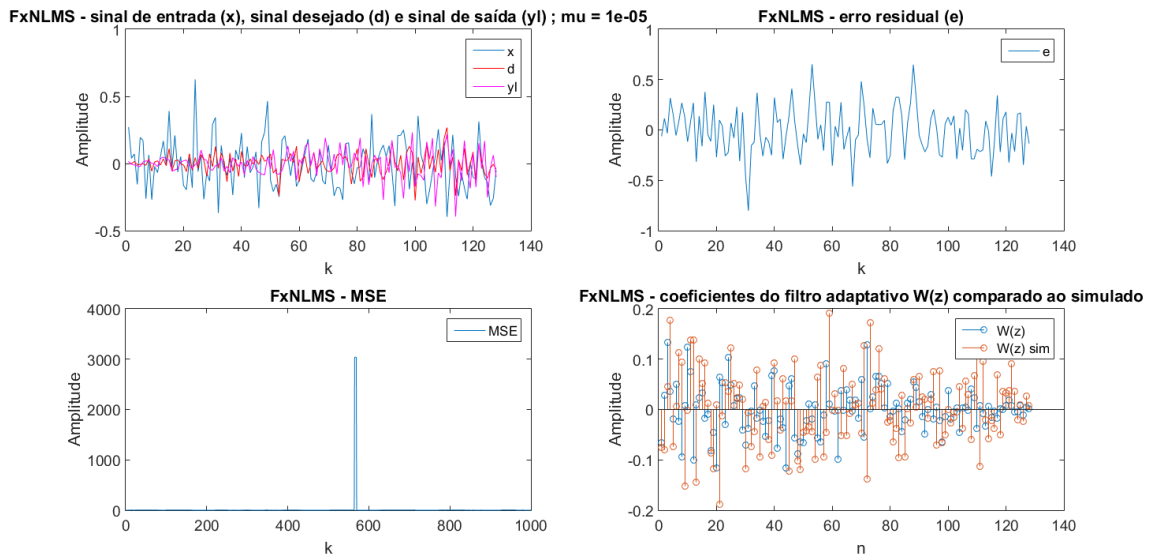
Fonte: Elaboração própria (2018).

Figura 30 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 4.c.



Fonte: Elaboração própria (2018).

Figura 31 - Resultado da execução do algoritmo FxNLMS no Teensy sem elementos externos para o teste 4.d.



Fonte: Elaboração própria (2018).

Tabela 7 - Tabela comparativa de valores para cada caso de teste do FxNLMS.

Teste do FxNLMS	Coefficiente de convergência ( $\mu$ )	Tamanho dos filtros $P(z)$ e $S(z)$	Correlação entre $W(z)$ simulado e $W(z)$ prático
Teste 3.b	0,001	56	9%
Teste 3.c	0,0001	56	não simulado
Teste 3.d	0,0001	56	71%
Teste 4.b	0,00005	128	84,8%
Teste 4.c	0,00005	128	não simulado
Teste 4.d	0,00005	128	2%

Fonte: Elaboração própria (2018).

Tabela 8 - Tabela comparativa do uso de recursos do Teensy para cada caso de teste do FxNLMS.

Teste do FxNLMS	Número máximo de pacotes de áudio utilizados	Porcentagem máxima de processamento
Teste 3.b	6	44,3%
Teste 3.c	7	50,3%
Teste 3.d	6	47,4%
Teste 4.b	7	88,1%
Teste 4.c	7	92,3%
Teste 4.d	7	89,7%

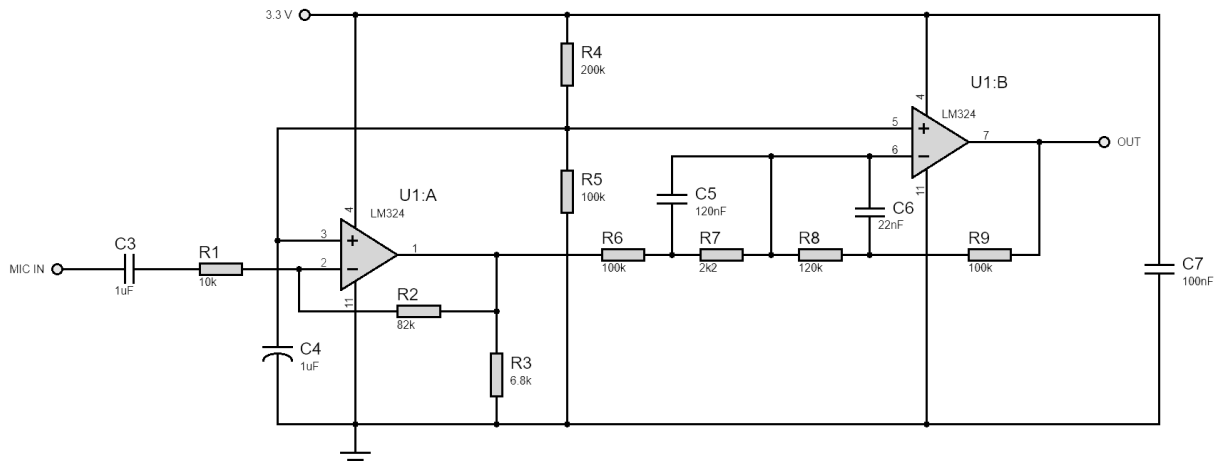
Fonte: Elaboração própria (2018).

Os resultados mostraram que apenas para os testes 3.b e 4.b, os quais possuem uma onda senoidal como sinal de entrada, e o teste 3.d, com o ruído branco como sinal de entrada, o erro foi devidamente minimizado. Para os testes 3.c, 4.c e 4.d, o algoritmo não conseguiu convergir.

### 3.2.2. Circuito de amplificação e equalização para os microfones

A fim de melhorar a razão sinal/ruído e equalizar a resposta em frequência do microfone, foi proposto um circuito de amplificação e equalização. O circuito, mostrado na Figura 32, utiliza amplificadores operacionais (ampops) para alcançar esses objetivos.

Figura 32 - Esquemático do circuito de amplificação e equalização do microfone MEMS.



Fonte: Elaboração própria (2018).

O circuito está dividido em dois estágios. A saída do primeiro estágio (terminal 1 do ampop U1:A) deve amplificar o sinal recebido do microfone na entrada inversora do primeiro ampop (terminal 2 de U1:A). O segundo estágio é composto por um circuito Baxandall de ganho para baixas frequências (BAXANDALL, 1952). Ele fará a equalização da resposta em frequência do microfone, já que o microfone utilizado possui grande atenuação nas baixas frequências, e é justamente nesta banda de frequência que o algoritmo é mais efetivo. O sinal sai do primeiro estágio e entra no segundo por sua entrada inversora, o terminal 6 de U1:B. A saída do segundo estágio (terminal 7 de U1:B) é a saída final do circuito que será conectada no Teensy por meio de um cabo P2.

O ampop LM324 foi o escolhido para o circuito, principalmente por ele operar bem com alimentação única de 3,3 V e por ser de fácil obtenção (FREESCALE SEMICONDUCTOR, 2013).

A equação de ganho para a estrutura de amplificador inversor é dada a seguir (PERTENCE JR, 2003):

$$A_v = \frac{V_o}{V_i} = -\frac{Z_f}{Z_i}. \quad (12)$$

Assim, tem-se para o primeiro estágio  $Z_f = R_2$  e  $Z_i = R_1$ .

$$A_v = -\frac{R_2}{R_1} = -\frac{82 \text{ k}\Omega}{10 \text{ k}\Omega} = -8,2. \quad (13)$$

Para o segundo estágio,  $Z_f = R_9 + R_8 \parallel X_{C6}$  e  $Z_i = R_6 + R_7 \parallel X_{C5}$ .

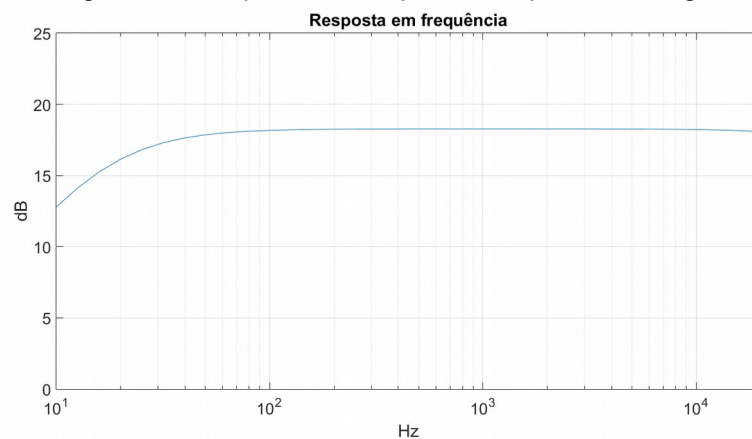
$$A_v = -\frac{R_9 + R_8 \parallel X_{C6}}{R_6 + R_7 \parallel X_{C5}}. \quad (14)$$

Onde:

$X_c$  = reatância capacitiva, definida por  $\frac{1}{2\pi f C}$ .

O circuito foi simulado através do software Proteus. A resposta em frequência do primeiro e do segundo estágio, e o do circuito como um todo foram obtidas na simulação e estão exibidas nas Figuras 33, 34 e 35, a seguir.

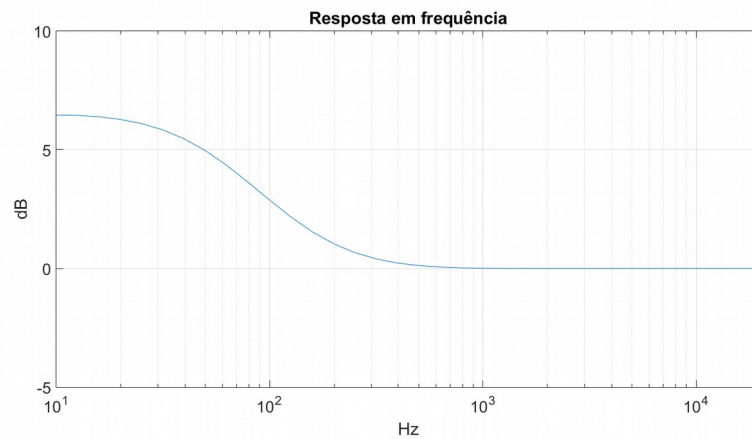
Figura 33 - Resposta em frequência do primeiro estágio.



Fonte: Elaboração própria (2018).

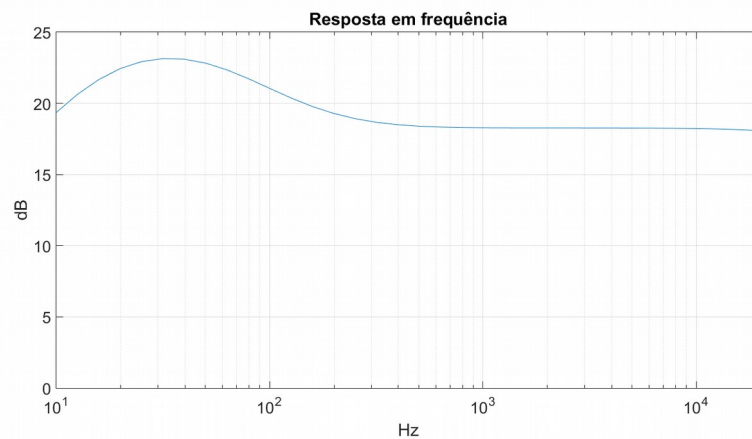


Figura 34 - Resposta em frequência do segundo estágio.



Fonte: Elaboração própria (2018).

Figura 35 - Resposta em frequência do circuito completo.

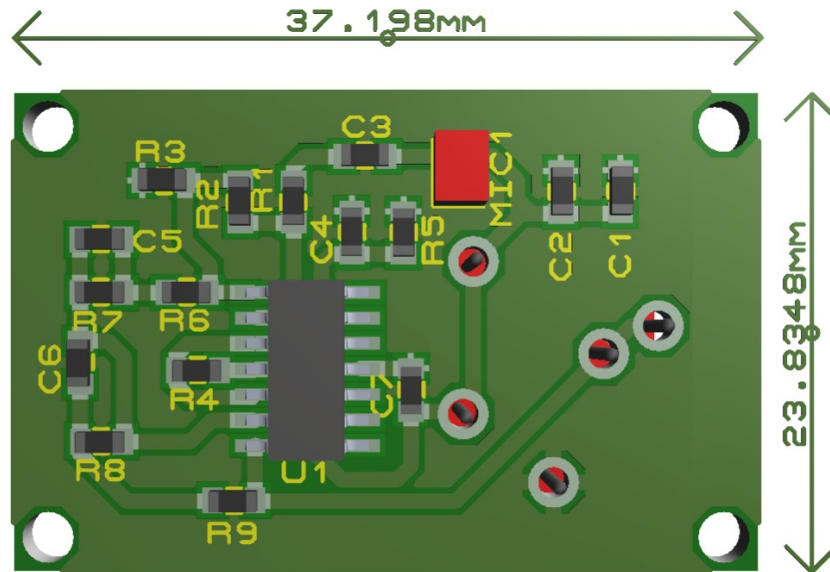


Fonte: Elaboração própria (2018).

O ganho em baixas frequências atingiu 24 dB e em altas frequências 18 dB.

O *layout* da PCI (placa de circuito impresso) foi feito com o auxílio do *software* Proteus. A placa é composta por componentes SMD (do inglês *surface-mount device*) e PTH (do inglês *pin through hole*), em duas camadas. Seu modelo 3D pode ser conferido na Figura 36.

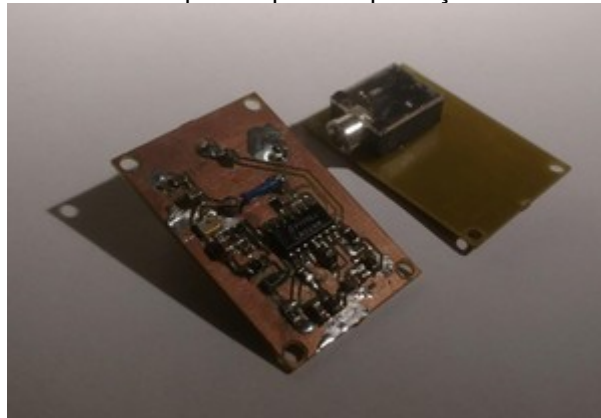
Figura 36 - Modelo 3D da PCI para a amplificação e equalização de cada microfone.



Fonte: Elaboração própria (2018).

Foram fabricadas duas placas idênticas para este projeto, uma para cada microfone. Os arquivos Gerber foram enviados para uma máquina fresadora produzir as trilhas e ilhas em uma chapa de fenolite. A soldagem dos componentes SMD e PTH foi feita de modo manual, usando ferramentas e utensílios: ferro de solda, estanho, estanho em pasta, estação de retrabalho, fluxo, etc. O resultado confere-se na Figura 37.

Figura 37 - Placas de circuito impresso para amplificação dos microfones produzidas.

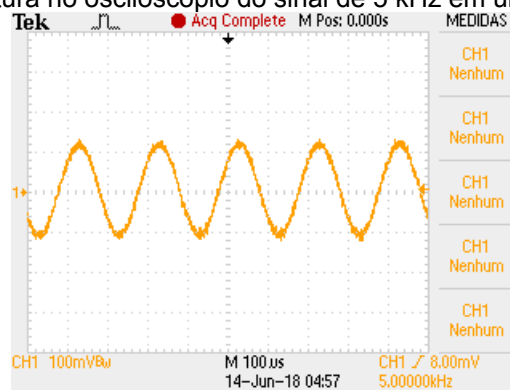


Fonte: Elaboração própria (2018).

Algumas ressalvas devem ser feitas com relação ao uso do conector fêmea de áudio no circuito. Uma conexão entre dois pinos do conector ficou em aberto, pois pensava-se que eles eram curto-circuitados internamente, mas isso só ocorre quando o conector macho não está conectado. Este descuido foi corrigido soldando um fio de cobre entre os terminais. A segunda ressalva se dá por conta do próprio uso de um conector de áudio para este circuito, porque percebeu-se que o modo como o conector é encaixado ocasionará um curto-circuito transitório na alimentação, se o microcontrolador estiver ligado ao se conectar o plugue. Pretende-se alterar o tipo de conector nas próximas versões das placas para corrigir este problema.

Com o objetivo de confirmar se o funcionamento na prática condiz com os resultados obtidos na simulação, foram realizadas medições em bancada. Alto-falantes foram posicionados em sentido oposto à PCI de cada microfone para gerar sinais senoidais com diferentes frequências por medição. Junto ao microfone, foi disposto um decibelímetro. As posições do microfone e do decibelímetro foram ajustadas para se obter  $80 \text{ dB(A)}_{\text{SPL}}$  em cada frequência desejada. A medida da tensão pico a pico para cada frequência foi feita por um osciloscópio digital, como exemplificado para a frequência de 5 kHz na Figura 38.

Figura 38 - Captura no osciloscópio do sinal de 5 kHz em um dos microfones.



Fonte: Elaboração própria (2018).

Deve-se levar em conta que, como as medidas foram feitas usando um decibelímetro na escala  $\text{dB(A)}_{\text{SPL}}$ , os resultados obtidos em Vp-p nas capturas dos sinais recebidos pelos microfones devem ser ponderados, o que gerou a Tabela 9:

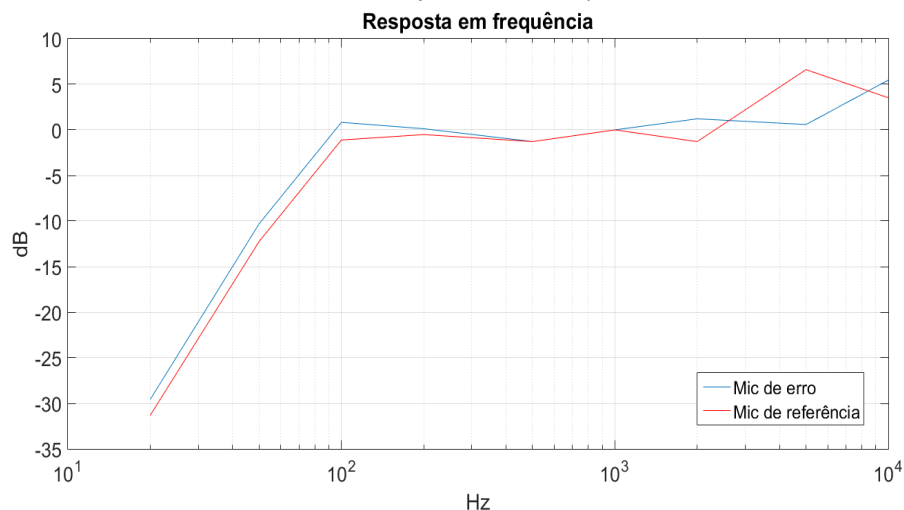
Tabela 9 - Resposta em frequência dos microfones de erro e de referência.

Frequência (Hz)	Microfone de erro (dB)	Microfone de referência (dB)
20	-31,3	-29,5
50	-12,2	-10,3
100	-1,11	0,82
200	-0,51	0,12
500	-1,28	-1,28
1k	0	0
2k	-1,28	1,21
5k	6,60	0,58
10k	3,52	5,46

Fonte: Elaboração própria (2018).

Os valores obtidos na Tabela 9 podem então ser analisados graficamente, onde o sinal vermelho representa o microfone de erro e o azul representa o microfone de referência, na Figura 39.

Figura 39 - Resposta em frequência dos microfones implementados nas PCs fabricadas (ganho em dB x frequência em Hz).



Fonte: Elaboração própria (2018).

A frequência de corte inferior passou de aproximadamente 150 Hz, da resposta em frequência do microfone, para 70 Hz com o circuito proposto.

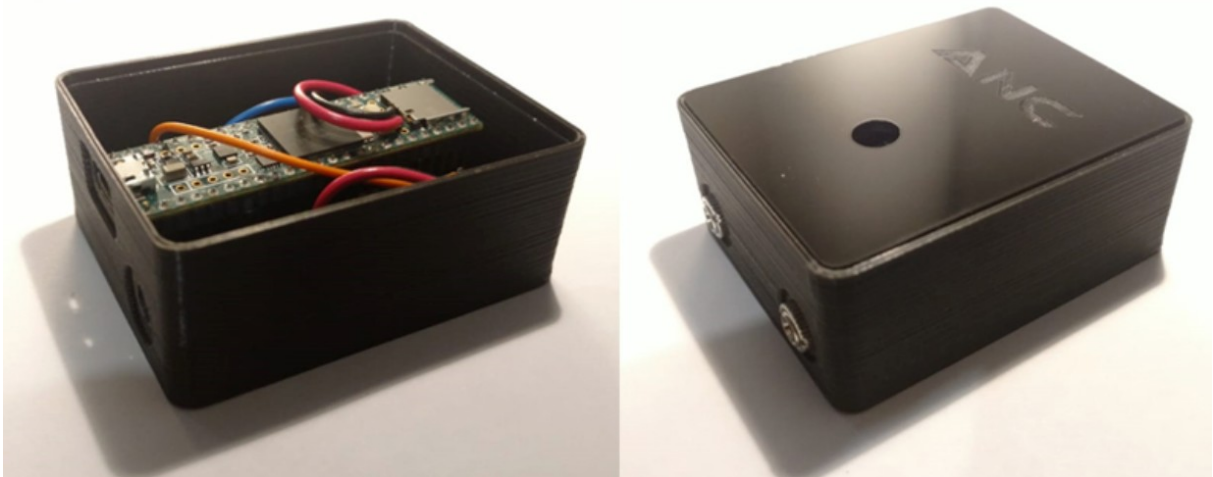
Ocorreram picos de ganho em altas frequências, que poderão ser equalizados em fabricações futuras.

### 3.2.3. Gabinete de proteção para o microcontrolador

Para proteger o microcontrolador contra choques mecânicos ou curto-circuitos acidentais, foi produzida um gabinete por meio de uma impressora 3D com o material ABS. Sua tampa foi feita por uma máquina laser, cortando uma lâmina de acrílico e a marcando com as inscrições ANC para identificação do protótipo. O projeto do gabinete foi realizado através do *software* SolidWorks e o da tampa pelo *software* Inkscape. Suas dimensões são de 7,0 x 5,5 x 2,5 cm.

A Figura 40 apresenta o gabinete produzido sem e com a tampa, respectivamente, com entradas para cabo P2 para os dois microfones e outra de saída para o alto-falante, uma entrada para o micro USB de alimentação e gravação do Teensy e um furo na tampa para acessar o botão de *reset* do microcontrolador.

Figura 40 - Gabinete produzido na impressora 3D e na máquina laser, sem e com tampa.



Fonte: Elaboração própria (2018).

### 3.2.4. Implementação do sistema com tubos de PVC

Este sistema visa representar o diagrama de blocos da Figura 4 e servir como uma etapa construtiva para o desenvolvimento futuro de um sistema voltado para fones de ouvido. O sistema montado em bancada está exibido na Figura 41.

Figura 41 - Sistema de testes controlado usando tubos de PVC.



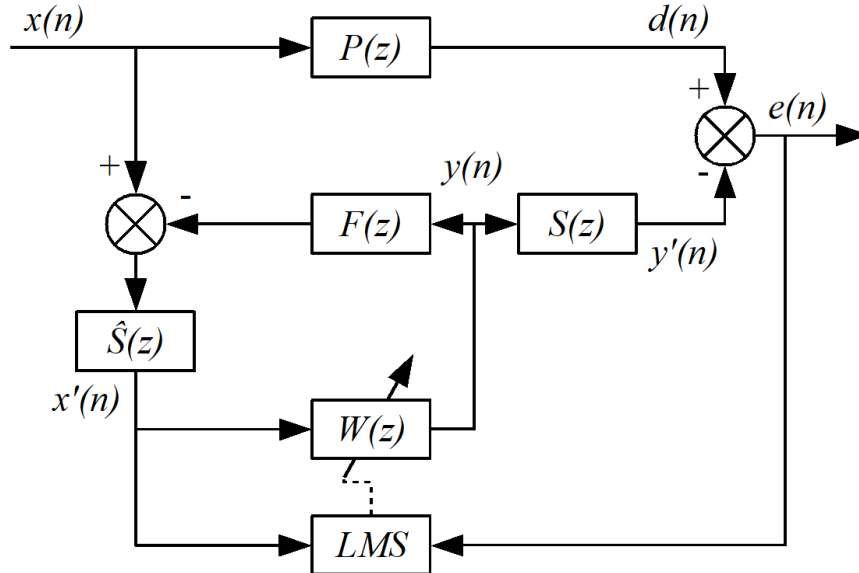
Fonte: Elaboração própria (2018).

No sistema, temos o Teensy conectado a uma bateria portátil via USB, às duas placas dos microfones e a um alto-falante. Dentro dos tubos vão os alto-falantes, um recebendo o sinal de um gerador de funções do *smartphone* para funcionar como o ruído a ser cancelado e outro recebendo o sinal antirruído gerado pelo microcontrolador, e as duas placas dos microfones. O tubo possui aproximadamente 40 cm entre o alto-falante de ruído e o microfone de referência, e 55 cm entre o alto-falante de ruído e o microfone de erro.

Foram observados problemas na implementação do sistema. Como ambos os microfones dividem o mesmo espaço sonoro, ocorre o chamado *feedback* acústico, onde o sinal de cancelamento gerado pelo sistema retorna ao microfone de referência. Para corrigir este problema, foi adicionado um novo filtro ( $F(z)$ ) para representar o caminho do alto-falante de cancelamento para o microfone de

referência, o que, segundo KUO e MORGAN (1999), gera a neutralização do efeito de *feedback*. A Figura 42 apresenta o novo diagrama de blocos para o sistema ANC:

Figura 42 - Diagrama de blocos do FxNLMS com adaptação lenta e neutralização de *feedback*.



Fonte: Elaboração própria (2018).

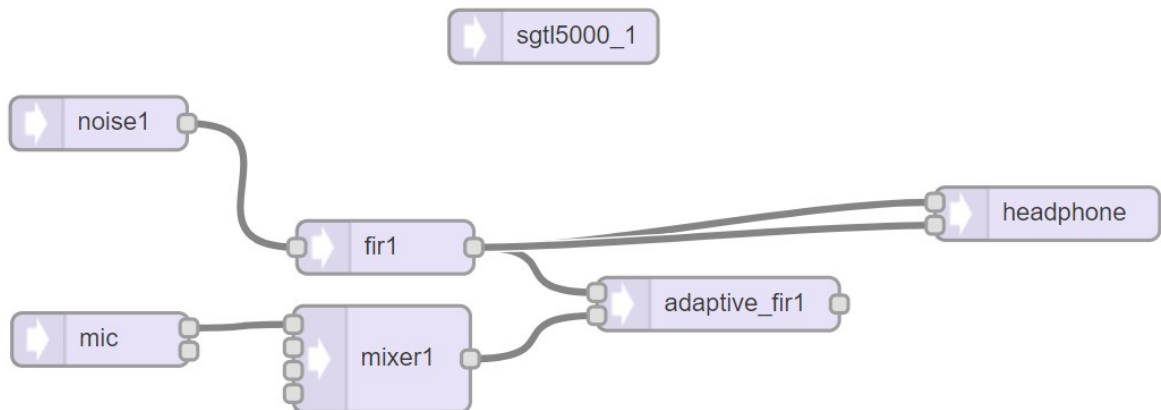
Como observado anteriormente na implementação do código no Teensy, o processamento do filtro adaptativo com 128 coeficientes foi bastante intenso, mas ainda dentro do que ele é capaz. Por isso, este será o número de coeficientes com os quais serão feitos os testes do grupo 5, a seguir.

Grupo 5) Implementação do sistema ANC com tubos de PVC ( $N=128$ ):

- NLMS para a estimativa de  $\hat{S}(z)$ ;
- NLMS para a estimativa de  $F(z)$ ;
- FxNLMS com senoide (400 Hz).

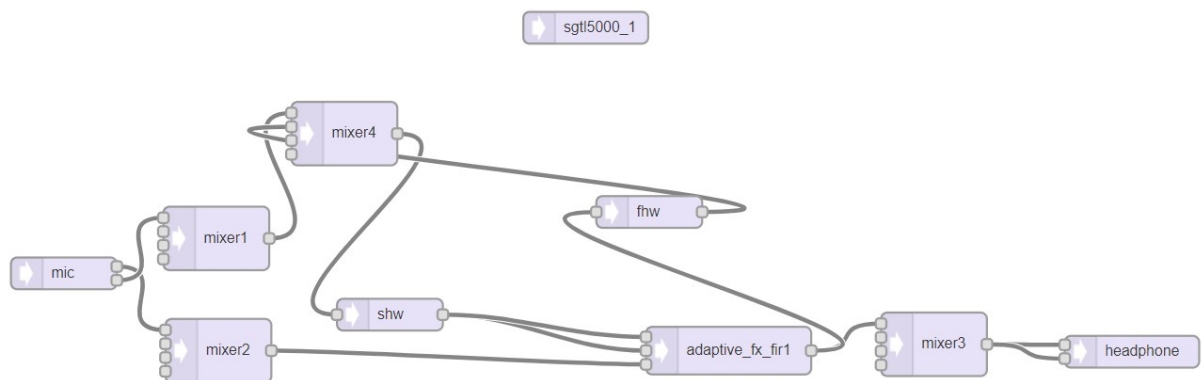
O diagrama de blocos com as conexões de áudio foi construído na ferramenta *Audio System Design Tool*, como nos passos anteriores, para o NLMS (Figura 43) e FxNLMS (Figura 44):

Figura 43 - Diagrama de blocos do NLMS para os testes 5.a e 5.b.



Fonte: Elaboração própria (2018).

Figura 44 - Diagrama de blocos do FxNLMS para o teste 5.c.

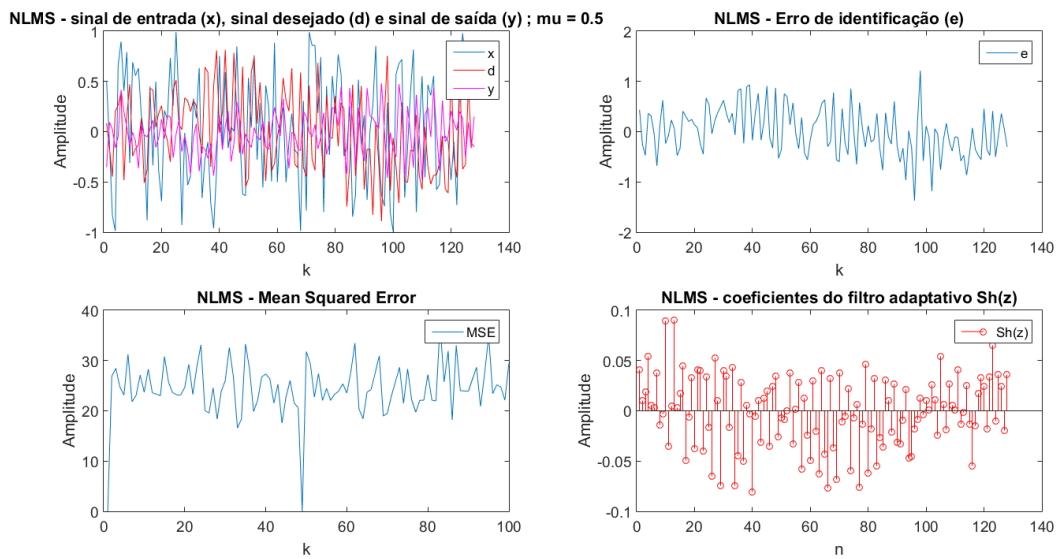


Fonte: Elaboração própria (2018).

Os resultados para o NLMS do teste 5.a foram os seguintes (Figuras 45 e 46):

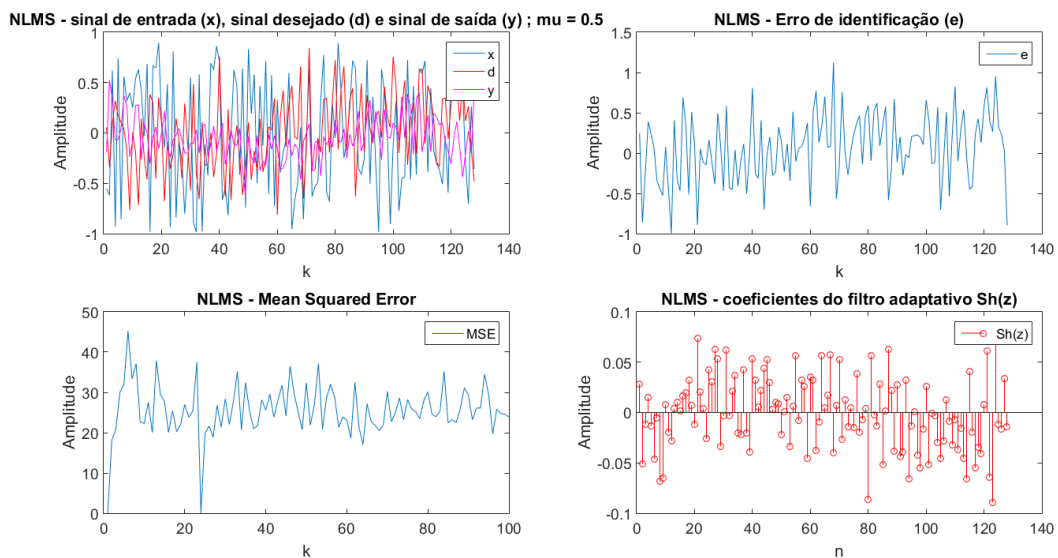


Figura 45 - Resultado da execução do algoritmo NLMS para a estimativa de  $\hat{S}(z)$  na estrutura com tubos para o teste 5.a.



Fonte: Elaboração própria (2018).

Figura 46 - Resultado da execução do algoritmo NLMS para a estimativa de  $F(z)$  na estrutura com tubos para o teste 5.b.



Fonte: Elaboração própria (2018).

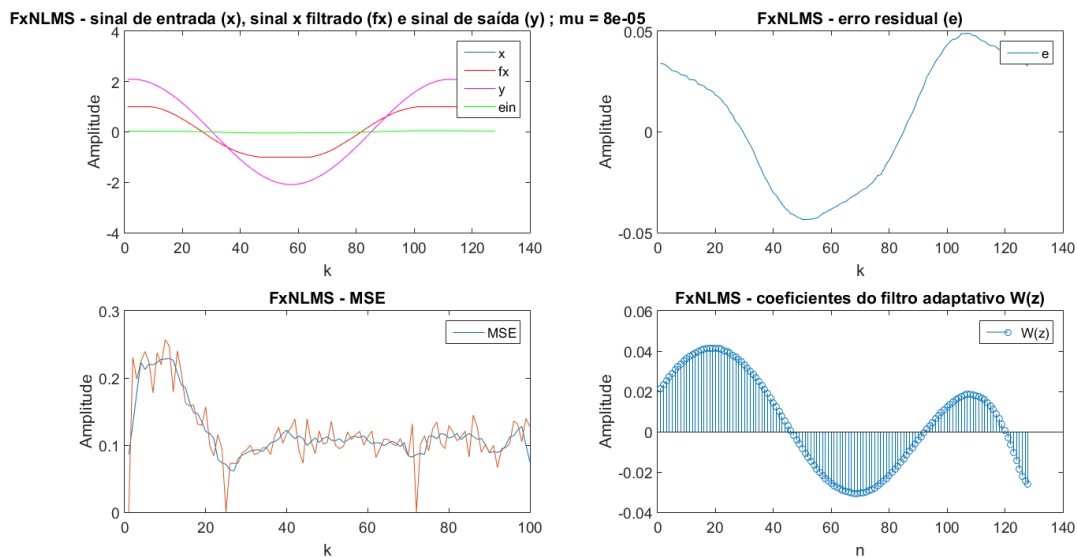
Como a estrutura montada não possui elementos com características ideais, existem perdas na captura dos sinais. Isso, acoplado ao próprio ruído

ambiente, pode dificultar uma estimativa precisa do caminho secundário. Neste caso foi necessário utilizar um fator de convergência maior, visto que valores similares aos utilizados em testes anteriores resultavam em pouca variação do sinal de saída.

O ruído branco foi gerado acusticamente na faixa de frequências de 180 Hz até 20 kHz, devido à resposta em frequência dos alto-falantes utilizados.

Os resultados do FxNLMS para o teste 5.c podem ser analisados na Figura 47.

Figura 47 - Resultado da execução do algoritmo FxNLMS para o cancelamento de uma senoide de 400 Hz para o teste 5.c.



Fonte: Elaboração própria (2018).

A execução do algoritmo resultou em uma queda na magnitude do MSE, como pode ser visto na imagem acima, de aproximadamente 0,23 para 0,1. Na prática, isso gerou uma queda na intensidade do som ao redor do microfone de erro de 79,7 dB(A)<sub>SPL</sub> para 76,6 dB(A)<sub>SPL</sub>, ou seja, aproximadamente 3 dB em aproximadamente 2,5 segundos.

O sistema chegou a obter cancelamento de até 10 dB(A)<sub>SPL</sub> na intensidade sonora, mas devido ao problema da instabilidade do sistema gerada pelo *feedback* acústico mencionada anteriormente, não foi possível mantê-lo por muito tempo. O sistema divergia, com sinal de saída do filtro adaptativo ultrapassando os limites possíveis do alto-falante de cancelamento.

Com a neutralização do *feedback* proposta, o sistema apresentou uma resposta mais estável, porém ainda ocorrem instabilidades no sistema dependendo do sinal de ruído indesejado. Observa-se na Figura 48 que mesmo com a diminuição do MSE, o sinal de entrada está saturando, provavelmente devido ao *feedback*, o que causa um sinal de saída do sistema excessivamente alto. Deve-se corrigir este problema para que se possam testar outros sinais de entrada.

A Tabela 10 apresenta os resultados das execuções do algoritmo NLMS nos tubos:

Tabela 10 - Tabela de valores na execução do algoritmo NLMS no sistema de tubos.

<b>Teste do NLMS</b>	<b>Coefficiente de convergência (<math>\mu</math>)</b>	<b>Tamanho dos filtros <math>P(z)</math> e <math>S(z)</math></b>	<b>Porcentagem máxima de processamento</b>
Teste 5.a	0,5	128	74,2%
Teste 5.b	0,5	128	73,9%

Fonte: Elaboração própria (2018).

A Tabela 11 apresenta os valores obtidos na execução do algoritmo FxNLMS no sistema de tubos:

Tabela 11 - Tabela de valores na execução do algoritmo FxNLMS no sistema de tubos.

<b>Teste do FxNLMS</b>	<b>Coefficiente de convergência (<math>\mu</math>)</b>	<b>Tamanho dos filtros <math>P(z)</math> e <math>S(z)</math></b>	<b>Porcentagem máxima de processamento</b>
Teste 5.c	0,00008	128	79,0%

Fonte: Elaboração própria (2018).

## 4. RESULTADOS

A Tabela 12 apresenta a comparação dos resultados da execução do algoritmo NLMS na simulação, na implementação no Teensy sem elementos externos e na implementação da estrutura com tubos:

Tabela 12 - Tabela de comparação dos resultados para o algoritmo NLMS na simulação, no Teensy sem elementos externos e na implementação da estrutura com tubos.

Teste do NLMS	Coeficiente de convergência ( $\mu$ )	Tamanho dos filtros $P(z)$ e $S(z)$	Convergência (redução do MSE)	Porcentagem máxima de processamento	Correlação entre $S(z)$ e $\hat{S}(z)$
Simulação	0,01	56	Convergiu	-	100%
Teensy sem elementos externos	0,01	56	Convergiu	40,9%	100%
Simulação	0,01	128	Convergiu	-	100%
Teensy sem elementos externos	0,01	128	Convergiu	79,0%	100%
Estrutura de testes com tubos	0,5	128	Não convergiu	74,2%	Desconhecido

Fonte: Elaboração própria (2018).

A Tabela 13 apresenta a comparação dos resultados da execução do algoritmo FxNLMS na simulação, na implementação no Teensy sem elementos externos e na implementação na estrutura com tubos para uma entrada senoidal de 400 Hz como ruído indesejado:

Tabela 13 - Tabela de comparação dos resultados para o algoritmo FxNLMS na simulação, no Teensy sem elementos externos e na implementação da estrutura com tubos para uma onda senoidal de entrada.

Teste do FxNLMS	Coefficiente de convergência ( $\mu$ )	Tamanho dos filtros $P(z)$ e $S(z)$	Convergência (redução do MSE)	Porcentagem máxima de processamento	Correlação entre $W(z)$ simulado e $W(z)$ prático
Simulação	0,001	56	Convergiu	-	-
Teensy sem elementos externos	0,001	56	Convergiu	44,3%	9%
Simulação	0,0001	128	Convergiu	-	-
Teensy sem elementos externos	0,00005	128	Convergiu	88,1%	84,8%
Estrutura de testes com tubos	0,00008	128	Convergiu	79,0%	Desconhecido

Fonte: Elaboração própria (2018).

Os resultados para outros sinais de ruído indesejado foram omitidos, visto que devem-se corrigir os problemas mencionados anteriormente na implementação do sistema prático.

A simulação e a implementação somente no microcontrolador dos algoritmos NLMS e FxNLMS apresentou os resultados esperados. A implementação do NLMS na estrutura de testes com tubos gerou resultados práticos com uma redução de 3 dB<sub>SPL</sub> até 10 dB<sub>SPL</sub> para ondas senoidais utilizando o algoritmo FxNLMS, o que indica a possibilidade de construção futura de um sistema com eficiência elevada.

## 5. CONCLUSÕES

Foram realizados neste trabalho a pesquisa, a simulação, a implementação no microcontrolador Teensy e a elaboração de uma estrutura de testes para um sistema ANC, utilizando versões derivadas dos algoritmos LMS e FxLMS. A maior parte dos objetivos definidos foi cumprida e um sistema funcional foi construído, apesar de o objetivo final ainda não ter sido alcançado. No mais, conhecimentos adicionais foram obtidos no uso de tecnologias como, por exemplo, microfones MEMS, microcontroladores e amplificadores operacionais. Também foram adquiridas novas compreensões de programação em C/C++, processamento digital de sinais, processamento de áudio, soldagem de componentes SMD e experiências práticas na realização de um protótipo.

Neste projeto, foi feita uma tentativa de aumentar ao máximo o número de coeficientes usados nos filtros adaptativos sem ultrapassar o limite de uso do processador para a execução dos algoritmos. Foram encontradas dificuldades na elaboração do código no microcontrolador, na aplicação prática no sistema de tubos. Algumas dificuldades foram superadas, outras terão de ser solucionadas em manutenções futuras. O maior dos problemas ocorreu na estabilidade do sistema, que impossibilitou, por agora, uma funcionalidade mais confiável.

Ainda há muito o que aprender sobre as diversas estruturas possíveis de cancelamento de ruído, mas este trabalho apresentou uma base para que isso seja possível no futuro. Também deve-se buscar técnicas na literatura da área para superar as dificuldades enfrentadas na implementação prática do sistema.

## REFERÊNCIAS

- ARM. **Cortex-M4 Processor**. [S.l.: s.n.], 2017. Disponível em: <<https://www.arm.com/products/processors/cortex-m/cortex-m4-processor.php>>. Acesso em: 4 jun. 2017.
- BAXANDALL, P. J. Negative-Feedback Tone Control: **Wireless World** Independent Variation of Bass and Treble Without Switches, p. 402–405, out 1952.
- BEYERDYNAMIC. *What is the difference between open and closed headphones?* Disponível em: <<https://support.beyerdynamic.com/hc/en-us/articles/200994502-What-is-the-difference-between-open-and-closed-headphones->>. Acesso em: 9 out. 2017.
- BORWICK, John. **Loudspeaker and Headphone Handbook**. 3. ed. Oxford: Focal Press, 2001. 718 p.
- CHANG, Cheng-Yuan; LI, Kuan-Hung. Active Noise Control in Headsets by Using a Low-Cost Microcontroller. **IEEE Transactions on Industrial Electronics** v. 58, p. 1936–1942, maio 2011.
- DINIZ, Paulo S. R. **Adaptive Filtering: Algorithms and Practical Implementation**. 2. ed. Norwell, Massachusetts, USA: Kluwer Academic Publishers, 2002. 568 p.
- DRAKE, Richard L.; VOGL, A. Wayne; MITCHELL, Adam W. M. **Gray's Anatomy for Students**. 3. ed. Philadelphia: Churchill Livingstone, 2014. 1191 p.
- FRADEN, Jacob. **Handbook of Modern Sensors: Physics, Designs, and Applications**. 5. ed. San Diego: Springer, 2016. 758 p.
- FREESCALE SEMICONDUCTOR. **Low Power Stereo Codec with Headphone Amp**. Technical data, nº SGTL5000. Austin, Texas, USA: Freescale Semiconductor, Inc., nov. 2013. Disponível em: <<https://www.pjrc.com/teensy/SGTL5000.pdf>>. Acesso em: 24 jul. 2018.
- GETDRAWINGS. *Earphones Drawings*. Disponível em: <<http://getdrawings.com/earphones-drawing#earphones-drawing-21.jpg>>. Acesso em: 21 jul. 2018.
- HAYKIN, Simon. **Adaptive Filter Theory**. 5. ed. Canadá: Pearson, 2014. 889 p. .
- HERRERA, Sarah *et al.* Amplified music with headphones and its implications on hearing health in teens. **International Tinnitus Journal** v. 20, n. 1, p. 42–47 , 2016.
- INGLE, Vinay K.; PROAKIS, John G. **Digital Signal Processing Using MATLAB** Physics, Designs, and Applications. Canadá: Cole Publishing Company, 2000. 418 p.
- KINSLER, Lawrence E. *et al.* **Fundamentals of Acoustics**. 4. ed. New York: John Wiley & Sons, 2000. 568 p.

KUO, S. M. et al. **Design of Active Noise Control Systems With the TMS320 Family**. Application report, n° SPRA042. Dallas, Texas, USA: Texas Instruments Incorporated, jun. 1996. Acesso em: 14 fev. 2018.

KUO, S.M.; MORGAN, D.R. Active Noise Control: A Tutorial Review. **Proceedings of the IEEE** v. 87, n. 6, p. 943–973, jun. 1999.

LIMA, Charles Borges De; VILLAÇA, Marco V. M. **AVR e Arduino**. 2. ed. Florianópolis: Ed. dos autores, 2012. 632 p.

LIU, Kuang-Hung et al. **Active Noise Cancellation Project**. Michigan: University of Michigan, abril 2008.

MU, Xiangbin; KO, JinSeok; RHEEM, JaeYeol. Modified FxLMS Algorithm for Active Noise Control and Its Real-Time Implementation. **Journal of The Institute of Electronics Engineers of Korea** v. 50, n. 9, p. 2416–2420, set 2013.

OEY, Agustinus. *Active noise control system using FxLMS algorithm*. Disponível em: <<https://www.mathworks.com/matlabcentral/fileexchange/27277-active-noise-control-system-using-fxlms-algorithm>>. Acesso em: 14 jul. 2017.

PERTENCE JR, Antonio. **Eletrônica Analógica** Amplificadores Operacionais e Filtros Ativos. 6. ed. Porto Alegre: Bookman, 2003. 299 p.

PJRC. **Audio Adaptor Board for Teensy 3.0 - 3.6**. [S.l: s.n.], 2017a. Disponível em: <[https://www.pjrc.com/store/teensy3\\_audio.html](https://www.pjrc.com/store/teensy3_audio.html)>. Acesso em: 4 jun. 2017.

PJRC. **Teensy Audio Library**. [S.l: s.n.], 2017b. Disponível em: <[https://www.pjrc.com/teensy/td\\_libs\\_Audio.html](https://www.pjrc.com/teensy/td_libs_Audio.html)>. Acesso em: 4 jun. 2017.

PJRC. **Teensy Technical Specifications**. [S.l: s.n.], 2017c. Disponível em: <<https://www.pjrc.com/teensy/techspecs.html>>. Acesso em: 4 jun. 2017.

PJRC. **Teensy USB Development Board**. [S.l: s.n.], 2017d. Disponível em: <<https://www.pjrc.com/teensy/index.html>>. Acesso em: 4 jun. 2017.

PURVES, Dale *et al.* **Neuroscience**. 4. ed. Sunderland: Sinauer Associates, 2008. 857 p.

SHURE. *Understanding Earphone / Headphone Specifications*. Disponível em: <<http://www.shure.com/americas/support/find-an-answer/understanding-earphone-headphone-specifications>>. Acesso em: 9 out. 2017.

ST. **MP23AB02B MEMS audio sensor high-performance analog bottom-port microphone**. Datasheet - production data, n° 026104. Geneva: STMicroelectronics, set 2014. Disponível em: <<https://www.mouser.com/datasheet/2/389/mp23ab02b-955093.pdf>>. Acesso em: 25 mar. 2018.

ST. **Tutorial for MEMS microphones**. Application note, n° AN4426. Geneva: STMicroelectronics, fev 2017. Disponível em: <[http://www.st.com/content/ccc/resource/technical/document/application\\_note/46/0b/](http://www.st.com/content/ccc/resource/technical/document/application_note/46/0b/)>



3e/74/cf/fb/4b/13/DM00103199.pdf/files/DM00103199.pdf/jcr:content/translations/en.DM00103199.pdf>. Acesso em: 8 out. 2017.

VU, Hong-Son *et al.* A 6.42 mW Low-Power Feed-Forward FxLMS ANC VLSI Design for In-ear Headphones. **IEEE** p. 2585–2588, 2015.

WHO. **1.1 billion people at risk of hearing loss, WHO highlights serious threat posed by exposure to recreational noise**. Comunicado à imprensa, Genebra: World Health Organization, de fevereiro de 2015. Disponível em: <<http://www.who.int/mediacentre/news/releases/2015/ear-care/en/>>. Acesso em: 6 mar. 2017.

WHO. **Deafness and hearing loss**. Lista de dados, Genebra: World Health Organization, atualização em fevereiro de 2017. Disponível em: <<http://www.who.int/mediacentre/factsheets/fs300/en/>>. Acesso em: 6 mar. 2017.

WIKIMEDIA COMMONS. *Tidens naturlære*. Disponível em: <[https://commons.wikimedia.org/wiki/Ear#/media/File:Tidens\\_naturl%C3%A6re\\_fig40.png](https://commons.wikimedia.org/wiki/Ear#/media/File:Tidens_naturl%C3%A6re_fig40.png)>. Acesso em: 8 out. 2017.

**APÊNDICE A – CÓDIGO DESENVOLVIDO PARA O PROJETO**

<https://github.com/gwgarcia21/anc>



MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA  
CATARINA  
CAMPUS FLORIANÓPOLIS

### TERMO DE AUTORIZAÇÃO DE DIVULGAÇÃO

Eu, Guilherme Wilm Garcia, de nacionalidade brasileira, CPF nº 043.221.009-02, RG nº 4.051-512, estudante do Curso Superior de Tecnologia em Eletrônica Industrial na qualidade de autor do TCC intitulado Estudo e Implementação de Cancelamento Ativo de Ruído Usando um Sistema Microcontrolado, AUTORIZO, neste ato de depósito, sua divulgação total e gratuita, para fins acadêmicos, em meio eletrônico, mediante registro nesta biblioteca, bem como em via impressa, se necessário, de acordo com determinação institucional e viabilidade técnica do Instituto Federal de Santa Catarina, Campus Florianópolis.

Ocasionará registro de patente? [ ] sim [x] não

Florianópolis, 10 de agosto de 2018.

Guilherme Wilm Garcia  
Assinatura do autor



MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA  
CATARINA  
CAMPUS FLORIANÓPOLIS

### TERMO DE COMPROMISSO DE ORIGINALIDADE

O presente termo é documento integrante de todo Trabalho de Conclusão de Curso (TCC) a ser submetido à avaliação do IFSC *Campus* Florianópolis como requisito necessário e obrigatório à obtenção do grau de Tecnólogo em Eletrônica Industrial.

Eu, Guilherme Wilm Garcia, CPF 093.229.009-07, Registro de Identidade 4051-51a, na qualidade de estudante de Graduação do Curso de Tecnologia em Eletrônica Industrial do IFSC *Campus* Florianópolis, declaro que o Trabalho de Conclusão de Curso apresentado em anexo, requisito necessário à obtenção do grau de Tecnólogo em Eletrônica Industrial, encontra-se plenamente em conformidade com os critérios técnicos, acadêmicos e científicos de originalidade.

Nesse sentido, declaro, para os devidos fins, que:

a) o referido TCC foi elaborado com minhas próprias palavras, ideias, opiniões e juízos de valor, não consistindo, portanto PLÁGIO, por não reproduzir, como se meus fossem, pensamentos, ideias e palavras de outra pessoa;

b) as citações diretas de trabalhos de outras pessoas, publicados ou não, apresentadas em meu TCC, estão sempre claramente identificadas entre aspas e com a completa referência bibliográfica de sua fonte, de acordo com as normas estabelecidas pelo IFSC *Campus* Florianópolis;

c) todas as séries de pequenas citações de diversas fontes diferentes foram identificadas como tais, bem como às longas citações de uma única fonte foram incorporadas suas respectivas referências bibliográficas, pois fui devidamente informado(a) e orientado(a) a respeito do fato de que, caso contrário, as mesmas constituiriam plágio;

d) todos os resumos e/ou sumários de ideias e julgamentos de outras pessoas estão acompanhados da indicação de suas fontes em seu texto e as mesmas constam das referências bibliográficas do TCC, pois fui devidamente informado(a) e orientado(a) a respeito do fato de que a inobservância destas regras poderia acarretar alegação de fraude.

O (a) Professor (a) responsável pela orientação de meu trabalho de conclusão de curso (TCC) apresentou-me a presente declaração, requerendo o meu compromisso de não praticar quaisquer atos que pudessem ser entendidos como plágio na elaboração de meu TCC, razão pela qual declaro ter lido e entendido todo o seu conteúdo e submeto o documento em anexo para apreciação do IFSC *Campus* Florianópolis como fruto de meu exclusivo trabalho.

Florianópolis, 10 de agosto de 2018.

Guilherme Wilm Garcia  
Assinatura do Estudante