



**Centro Federal de Educação Tecnológica de Minas Gerais
Unidade Araxá**

Gabriel Silva Ferreira

**CONSTRUÇÃO DE UMA UNIDADE REPROGRAMÁVEL DE EFEITO
PARA GUITARRA ELÉTRICA UTILIZANDO SISTEMA
MICROCONTROLADO**

Araxá/MG
2022

Gabriel Silva Ferreira

**CONSTRUÇÃO DE UMA UNIDADE REPROGRAMÁVEL DE EFEITO
PARA GUITARRA ELÉTRICA UTILIZANDO SISTEMA
MICROCONTROLADO**

Monografia apresentada ao Curso de Engenharia de Automação Industrial, do Centro Federal de Educação Tecnológica de Minas Gerais – Unidade Araxá, como requisito parcial para obtenção do grau de Bacharel em Engenharia de Automação Industrial.

Orientador: Prof. Me. Frederico Duarte Fagundes

Araxá/MG
2022

Gabriel Silva Ferreira

**CONSTRUÇÃO DE UMA UNIDADE REPROGRAMÁVEL DE EFEITO
PARA GUITARRA ELÉTRICA UTILIZANDO SISTEMA
MICROCONTROLADO**

Trabalho de Conclusão de Curso apresentado ao
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá,
como requisito parcial para obtenção do grau de
Bacharel em Engenharia de Automação Industrial

Araxá, 18 de julho de 2022.

BANCA AVALIADORA

Presidente e Orientador: Prof. Me. Frederico Duarte Fagundes
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

Membro Titular: Dr. Kléber Lopes Fontoura
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

Membro Titular: Dr. Sergio Luiz da Silva Pithan
Centro Federal de Educação Tecnológica de Minas Gerais - Unidade Araxá

DEDICO ESTE TRABALHO

*Aos meus pais, meus irmãos e meus amigos,
que sempre me apoiaram e me fizeram acreditar na realização dos meus sonhos.*

AGRADECIMENTOS

Aos meus pais, Doriedson Ferreira e Nice Aparecida Silva, por todo amor, apoio e suporte.

Aos meus irmãos que sempre estiveram presentes me ajudando nessa caminhada.

Aos meus amigos por todas as palavras de incentivo, bem como gestos que sempre me deram muita força.

Ao meu orientador por ter me acompanhado pontualmente, dando todo o auxílio necessário para a elaboração do projeto.

Aos professores que aceitaram participar dessa banca por contribuírem e por serem parte desse trabalho.

Ao CEFET-MG e a todos os seus membros.

“É muito melhor lançar-se em busca de conquistas grandiosas, mesmo expondo-se ao fracasso, do que alinhar-se com os pobres de espírito, que nem gozam muito nem sofrem muito, porque vivem numa penumbra cinzenta, onde não conhecem nem vitória, nem derrota.”

THEODORE ROOSEVELT

RESUMO

O processamento digital de sinais de uma unidade digital de efeitos para instrumentos musicais tem se aprimorado ao longo do tempo, tornando possível a obtenção de resultados sonoros com qualidade, semelhantes aos resultados obtidos com unidades analógicas de efeitos, mas com características sonoras um pouco diferentes. Em vista deste cenário, surgiu o seguinte questionamento de pesquisa: de que maneira é possível desenvolver uma unidade digital de efeito tremolo para guitarra elétrica que seja reprogramável? Ante a essa pergunta, aventa-se, como hipótese de pesquisa, que se for elaborado um sistema utilizando uma plataforma microcontroladora, suficientemente eficiente, será possível desenvolver uma unidade digital de efeito para guitarra elétrica, a qual seja reprogramável e possibilite uma alta qualidade sonora. Dito isso, o presente trabalho teve como objetivo geral desenvolver uma unidade de efeito de tremolo para guitarra elétrica, reprogramável, utilizando sistema microcontrolado. Este estudo poderá auxiliar profissionais que atuam nas áreas de eletrônica, elétrica, música e sonorização, uma vez que apresenta diversos conceitos e oferece um equipamento de qualidade. Para a execução desta proposta, algumas etapas metodológicas foram contempladas, quais sejam: (i) estudo teórico sobre: efeitos de áudio; os conceitos da conversão analógica-digital e digital-analógica de sinais elétricos; o detalhamento de processamento de sinais; e as plataformas microcontroladoras que tenham *hardwares* suficientemente eficientes; (ii) após estudo sobre as plataformas microcontroladoras, foi escolhida a ESP32 DevKit v1 e os demais componentes eletrônicos utilizados que atenderam aos objetivos desse trabalho. Após essa escolha, passou-se à realização da programação proposta, utilizando o ambiente de desenvolvimento integrado Arduino IDE e a linguagem de programação C; (iii) criação do protótipo de uma unidade digital de efeito para guitarra elétrica; (iv) testagem e verificação da qualidade do sinal, antes e após processamento digital realizado pela plataforma microcontroladora. Após realização dos testes, foi validado a capacidade do microcontrolador ESP32 de realizar processamento de áudio em tempo real e em alta frequência.

Palavras-chave: Processamento de sinais. Guitarra. Sonorização. Microcontrolador. Frequência.

ABSTRACT

The digital signal processing of an effects digital unit for musical instruments has improved over time, making it possible to attain quality sound results, similar to the results got with effects analog units, but with slightly different sound characteristics. Given this scenario, the following research question arose: how is it possible to develop a re-programmable tremolo effect digital unit for electric guitar? Faced with this question, it suggested, as a research hypothesis, that if a system is elaborated using a microcontroller platform, sufficiently efficient, it will be possible to create an effect digital unit for electric guitar, which is re-programmable and enable high sound quality. That said, the present work aimed to develop a re-programmable tremolo effect unit for electric guitar, using a micro-controlled system. This study may help professionals who work in electronics, electrical, music, and sound since it presents several concepts and offers quality equipment. For the execution of this proposal, some methodological steps were considered, namely: (i) theoretical study on: audio effects; the concepts of analog-digital and digital-analog conversion of electrical signals; the detailing of signals processing; and microcontroller platforms that have sufficiently efficient and cost-effective hardware; (ii) after studying the microcontroller platforms, ESP32 DevKit v1 was chosen and the other electronic components used that met the objectives of this work. After this choice, it carried out the proposed programming, using the Arduino IDE integrated development environment and the C programming language; (iii) creation of a prototype of an effect digital unit for electric guitar; (iv) testing and verification of signal quality, before and after digital processing performed by the microcontroller platform. After carrying out the tests, the ability of the ESP32 microcontroller to perform real-time and high-frequency audio processing was validated.

Keywords: Signal processing. Guitar. Sound reinforcement. Microcontroller. Frequency.

LISTA DE ILUSTRAÇÕES

Figura 1: (a) pedal RV-5 (b) pedaleira ME-25.	14
Figura 2: (a) sinal contínuo no tempo (b) sinal discreto no tempo.....	17
Figura 3: Formato da onda produzida por diferentes instrumentos para uma mesma nota musical.....	18
Figura 4: (a) sinal discreto no tempo (b) sinal discreto no espectro da frequência.	22
Figura 5: Processamento digital de sinais analógicos.	22
Figura 6: Primeiro modelo de captador eletromagnético.	24
Figura 7: Linha de sinais do protocolo I2S.....	29
Figura 8: Plataforma ESP32 DevKit v1.	31
Figura 9: Diagrama em blocos da unidade de efeito.	33
Figura 10: Esquemático circuito amplificador diferencial.	35
Figura 11: Esquemático circuito subtrator.....	36
Figura 12: Módulo CJMCU-1808.	37
Figura 13: Esquemático circuito CJMCU-1808.	39
Figura 14: Esquemático circuito de clock do sistema CJMCU-1808.....	40
Figura 15: Módulo CJMCU-1334.	41
Figura 16: Esquemático circuito integrado UDA-1334.....	42
Figura 17: Esquemático circuito conversor digital-analógico.	43
Figura 18: Esquemático circuito interface display LCD.	44
Figura 19: Fluxograma efeito tremolo.....	45
Figura 20: Circuito amplificador diferencial.	46
Figura 21: Saída diferencial AN7523N.....	47
Figura 22: Circuito Subtrator.....	48
Figura 23: Offset do circuito subtrator.	49
Figura 24: Circuito de Clock SCK.	49
Figura 25: Sinais de clock.	50
Figura 26: Circuito conversores de sinais.....	51
Figura 27: Forma de onda conversores de sinais.....	51
Figura 28: Circuito interface através de simulador.	52
Figura 29: Circuito interface elaborado na protoboard.	52

Figura 30: Circuito unidade de efeito.....	53
Figura 31: Sinal de áudio com efeito tremolo.....	54

LISTA DE QUADROS

Quadro 1: Arduino UNO REV3, Arduino MEGA 2560, ESP32 e ESP8266.	27
Quadro 2: Tipo de variável x Espaço ocupado em disco.	32
Quadro 3: Modo de operação CJMCU-1808.....	38

LISTA DE ABREVIATURAS E SIGLAS

ADC	<i>Analog to Digital Converter</i> (Conversor Analógico Digital)
CPU	<i>Central Processing Unit</i> (Central Única de Processamento)
DAC	<i>Digital to Analog Converter</i> (Conversor Digital Analógico)
DFT	<i>Discrete Fourier Transform</i> (Transformada de Fourier Discreta)
FFT	<i>Fast Fourier Transform</i> (Transformada Rápida de Fourier)
HPF	<i>High Pass Filter</i> (Filtro Passa-Alta)
I2C	<i>Inter-Integrated Circuit</i> (Circuito Inter Integrado)
I2S	<i>Inter-Integrated Circuit Sound</i> (Som de Circuito Inter Integrado)
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
LPF	<i>Low Pass Filter</i> (Filtro Passa-Baixa)
PLL	<i>Phase Locked Loop</i> (Loop de Fase Bloqueada)

SUMÁRIO

1	INTRODUÇÃO	13
2	FUNDAMENTAÇÃO TEÓRICA.....	16
2.1	SINAIS E SISTEMAS	16
2.2	SONORIZAÇÃO	17
2.3	PROCESSAMENTO DE SINAIS	19
2.3.1	Filtros de sinais.....	20
2.3.2	Processamento digital de sinais analógicos contínuos no tempo	21
2.4	BREVE HISTÓRICO DA GUITARRA ELÉTRICA.....	23
2.5	UNIDADES DE EFEITOS	25
2.5	SISTEMAS MICROCONTROLADOS	27
3	METODOLOGIA.....	30
3.1	ESP-32.....	31
3.1.1	IDE e linguagem de programação	32
3.2	ETAPAS DA CONSTRUÇÃO DA UNIDADE DE EFEITO	33
3.2.1	Sistema de pré-amplificador.....	33
3.2.2	Sistema de conversão analógica-digital.....	36
3.2.3	Sistema de conversão digital-analógica.....	41
3.2.4	Interface eletrônica de sinais	43
3.2.5	Configuração protocolo I2S.....	44
3.2.6	Programação efeito tremolo.....	45
4	RESULTADOS E DISCUSSÕES	46
4.1	RESULTADOS E ANÁLISES DOS CIRCUITOS ELETRÔNICOS	46
4.1.1	Pré-amplificador	46
4.1.2	Circuito de clock.....	49
4.2.3	Circuito conversores de sinais	50
4.2.4	Interface eletrônica	52
4.2	TESTES FINAIS: EFEITO TREMOLO.....	53
5	CONSIDERAÇÕES FINAIS.....	55
	REFERÊNCIAS	57
	APÊNDICE A – PROGRAMAÇÃO DO SISTEMA ESP-32.....	59

1 INTRODUÇÃO

O processamento de sinais lida com a representação de informações e a transformação e a manipulação de dados contidos nos sinais. Dessa forma, é possível, através do processamento de sinais, combiná-los por meio de operações matemáticas, separá-los em um conjunto de sinais que foram anteriormente combinados, amplificá-los, entre outras operações (OPPENHEIM; SHAFER, 2013).

As técnicas de processamento de sinais têm sido muito utilizadas nas mais diversas aplicações. Segundo Oppenheim e Shafer (2013, p. 1), “algoritmos e *hardwares* de processamento de sinais são prevalentes em uma grande variedade de sistemas, desde sistemas militares altamente especializados e aplicações industriais até sistemas eletrônicos de baixo custo e alto volume de produção voltados ao consumidor”. Ante a essa diversidade de emprego, esse processamento pode ser realizado de forma analógica ou digital. O processamento analógico é executado por meio de sistemas analógicos implementados por circuitos eletrônicos ou dispositivos mecânicos. O processamento digital é efetivado para sinais digitais por meio de algoritmos computacionais.

Quantidades e variáveis mensuradas e controladas, oriundas do mundo real, são em grande parte analógicas. Apesar disso, o processamento digital de sinais é predominante em diversos sistemas. Isso porque as evoluções dos controladores digitais, dos conversores de sinais e das técnicas de processamento de sinais digitais proporcionaram o desenvolvimento de sistemas digitais altamente eficientes e mais flexíveis se comparados aos sistemas analógicos (TOCCI; WIDMER; MOSS, 2007). Diante disso, no ramo de unidades de efeitos para guitarras elétricas, o processamento de sinais digitais tem sido amplamente utilizado para criar timbres variados e adicionar efeitos sonoros. Entre as unidades de efeitos para guitarra elétrica mais empregadas na atualidade estão os pedais de efeitos e as pedaleiras.

Os pedais de efeitos para guitarra elétrica podem ser implementados tanto por sistemas analógicos quanto por sistemas digitais e contêm, normalmente, a configuração de apenas um efeito de áudio. Devido à limitação no número de efeitos contidos nos pedais de efeitos, é necessário, diversas vezes, criar conjuntos de vários pedais para obter o som desejado. Por outro lado, as pedaleiras são unidades digitais multiefeitos para guitarra elétrica que contêm diversas opções de controle de timbres e efeitos de áudio. As pedaleiras oferecem a possibilidade de utilização simultânea de diferentes efeitos sonoros, tornando o sistema de unidade digital multiefeitos um sistema mais compacto e portátil, se comparado aos sistemas que utilizam conjuntos de pedais de efeitos analógicos. Atualmente, as pedaleiras de alto

custo possibilitam ainda a simulação de diversos modelos de amplificadores para guitarra elétrica e interface de áudio integrada. Em contrapartida, as unidades digitais multiefeitos mais básicas possuem menos recursos disponíveis e menor qualidade do sinal processado.

A Figura 1 traz a ilustração do pedal digital de *Reverb* RV-5 e da unidade digital multiefeitos, ou pedaleira, BOSS ME-25.

Figura 1: (a) pedal RV-5 (b) pedaleira ME-25.



Fonte: BOSS (2021).

De acordo com Rocha (2011), as unidades digitais multiefeitos surgiram como alternativa, destinada aos músicos amadores e de orçamento limitado, de banco de efeitos de menor custo e mais prático do que o proporcionado pelos pedais de efeitos. Porém, as unidades digitais multiefeitos ainda apresentam um custo elevado. Com o avanço da tecnologia, as unidades digitais proporcionaram processamentos de efeitos de áudio com qualidade semelhante aos realizados em unidades de efeitos analógicas e passaram a ser utilizadas pelos mais diversos guitarristas.

Atualmente, existem no mercado diversos tipos de unidades de efeitos para guitarras elétricas, como, por exemplo: os pedais de efeitos de ambiência, de ganho, de modulação e de tonalidade. O efeito de modulação tremolo, por exemplo, resulta na modulação da amplitude do sinal de áudio, ou seja, da intensidade sonora, em intervalos fixos no tempo.

Ante a essa contextualização, com este projeto, buscou-se desenvolver uma unidade digital de efeito tremolo para guitarra elétrica. A unidade digital criada é reprogramável, permitindo a programação de novos efeitos e customização do efeito já programado. Assim

procedendo, a pergunta de pesquisa que motivou este estudo é: de que maneira é possível desenvolver uma unidade digital de efeito tremolo para guitarra elétrica que seja reprogramável? Para responder essa pergunta sugere a hipótese de que, se for elaborado um sistema utilizando uma plataforma microcontroladora, suficientemente eficiente, será possível desenvolver uma unidade de efeito para guitarra elétrica que possibilite uma alta qualidade sonora e que seja reprogramável.

Esta pesquisa se justifica na medida em que a efetivação do projeto trará a oportunidade do músico acessar unidades de efeitos para guitarra elétrica, no qual o próprio músico poderá montar, reprogramar e selecionar os efeitos conforme desejado. Além disso, o presente trabalho poderá auxiliar profissionais que atuam nas áreas de automação, eletrônica, elétrica, música e sonorização, uma vez que apresenta aplicação e conceitos de áudio, microcontroladores e processamento de sinais.

Ante ao questionamento de pesquisa, o objetivo geral deste trabalho foi construir uma unidade digital de efeito tremolo, reprogramável, para guitarra elétrica utilizando sistema microcontrolado. Em face desse objetivo geral, este trabalho foi norteado pelos seguintes objetivos específicos: pesquisar teoricamente os conceitos atinentes ao estudo, bem como as plataformas microcontroladoras que tenham hardwares suficientemente eficientes; escolher uma plataforma microcontroladora e proceder à realização da programação do efeito de tremolo, de modo a obter efeito de áudio de boa qualidade; criar o protótipo de uma unidade digital reprogramável de efeito para guitarra elétrica e verificar a qualidade do sinal antes e após processamento digital realizado pela plataforma microcontroladora.

Frente ao delineamento de estudo apresentado, este projeto se organiza, além desta introdução e das referências, em quatro capítulos, quais sejam: fundamentação teórica, metodologia, resultados e discussões e, por fim, considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

A guitarra elétrica surgiu a partir da necessidade de se obter maior projeção sonora de determinados instrumentos musicais, como, por exemplo, do violão, da guitarra havaiana e da guitarra acústica archtop (ROCHA, 2011). Na sociedade moderna, guitarristas frequentemente utilizam unidades de efeitos para modificar o timbre de seus instrumentos. Essas unidades podem ser tanto analógicas quanto digitais, tendo por função modificar o som captado do instrumento através de diferentes técnicas de processamento de sinais.

2.1 SINAIS E SISTEMAS

Segundo Oppenheim e Schaffer (2013), o conceito “sinal”, geralmente, é adotado para representar algo que transmite informação. Em outras palavras, sinal pode ser definido como veículo portador e transmissor de dados. Tanto é assim que, por exemplo, em uma planta industrial, os sinais podem transmitir dados sobre o estado ou o comportamento de um sistema físico ou ainda sinais de comando para atuadores do sistema.

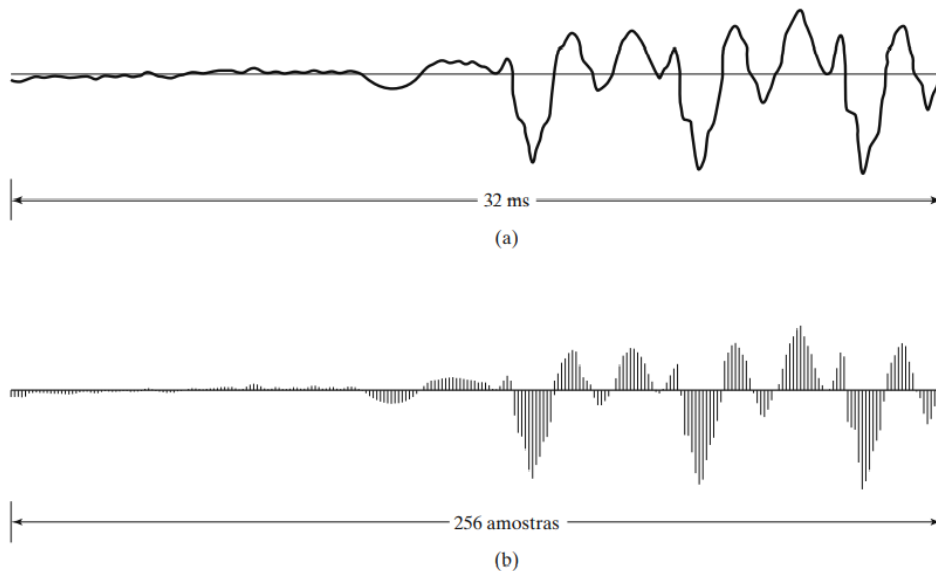
Os sistemas são responsáveis por modificar e/ou extrair informações dos sinais de entrada. Segundo Lathi (2007), um sistema pode ser construído com componentes físicos, elétricos, mecânicos, hidráulicos ou ainda por algoritmos que fazem análise ou processamento dos sinais de entrada.

Os sinais podem ser classificados quanto ao número de variáveis independentes, quanto à frequência e quanto à amplitude. Quanto ao número de variáveis independentes, são classificados em sinais: (i) unidimensionais, (ii) bidimensionais ou (iii) multidimensionais (OPPENHEIM; SCHAFFER, 2013). Os sinais unidimensionais possuem apenas uma variável independente, os quais podem ser representados pela variação da amplitude de um sinal elétrico correspondente à variável independente tempo. Os sinais bidimensionais possuem duas variáveis independentes, podendo ser representados pela variação de amplitudes do sinal em um espaço bidimensional. Por fim, os sinais multidimensionais possuem três ou mais variáveis independentes, como, por exemplo, variação de amplitudes do sinal, no tempo, em um espaço bidimensional.

De acordo com a frequência em que um sinal é especificado no tempo, os sinais são denominados sinais contínuos no tempo ou sinais discretos no tempo (conforme a Figura 2). Sinais contínuos no tempo são aqueles especificados para valores contínuos no tempo,

enquanto sinais discretos no tempo são especificados para um conjunto finito de valores discretos no tempo (LATHI, 2007).

Figura 2: (a) sinal contínuo no tempo (b) sinal discreto no tempo.



Fonte: Oppenheim e Schaffer (2013 p. 8).

Conforme as características da amplitude, os sinais são classificados em analógicos ou digitais. Sinais analógicos são aqueles em que sua amplitude pode assumir qualquer valor em uma faixa contínua, já os sinais digitais são aqueles em que a sua amplitude está limitada a um número finito de valores (LATHI, 2007). Ante a essas definições, portanto, a corrente elétrica, sinal de saída da guitarra elétrica, é classificada como sinal analógico e contínuo no tempo. Em contrapartida, o sinal que será processado pelo microcontrolador é um sinal digital e discreto no tempo, que se tornará, posteriormente, um sinal contínuo no tempo a ser reproduzido por alto-falantes.

2.2 SONORIZAÇÃO

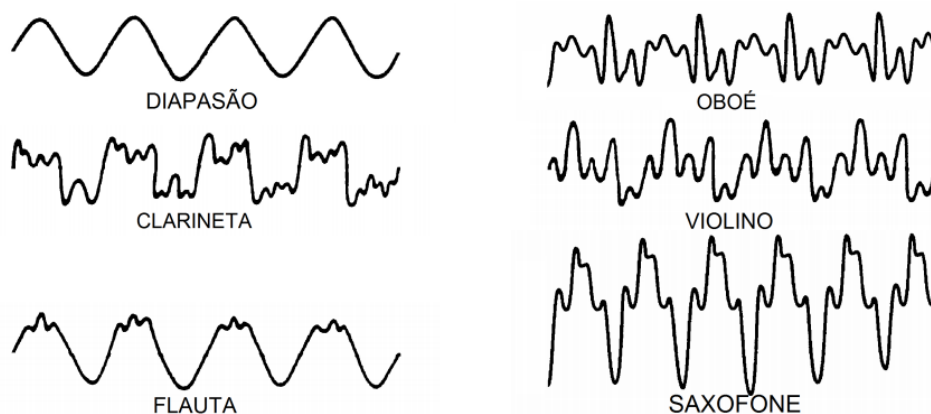
O conceito “som” pode ser definido como onda capaz de se propagar em meios físicos, como, por exemplo, pelo ar a partir das vibrações de suas moléculas. Segundo Davis e Jones (1990), sons de maior volume advém de ondas físicas de maior amplitude, de forma que o volume do som é diretamente proporcional à amplitude em que moléculas de ar são comprimidas e expandidas.

O som pode ser descrito por algumas de suas propriedades fisiológicas e a partir da fonte sonora em que é produzido, que influenciam em como ele é percebido pelo ser humano. Isso ocorre porque as ondas sonoras se diferem quanto à intensidade, à altura e/ou ao timbre, sendo possível, por meio dessa alteração, identificar sons diferentes. A intensidade do som resulta da pressão exercida pela onda sonora sobre algum sensor, e está intimamente relacionada à taxa de transmissão de energia (BORGES; RODRIGUES, 2017). Dessa forma, quanto maior a intensidade da onda, maior a amplitude e o volume do som percebido.

Segundo Borges e Rodrigues (2017), a altura do som está relacionada à frequência da onda sonora, sendo que, quanto maior a frequência, mais agudo será o som. A frequência da onda determina qual a nota musical que o sinal irá produzir, sendo que, a frequência de som audível, ou seja, aquela que o ouvido humano consegue identificar, está compreendida aproximadamente entre 20 Hz e 20 kHz. Dito isso, onda sonora de baixa frequência apresenta som grave (som baixo), enquanto onda sonora de alta frequência apresenta som agudo (som alto). Por outro lado, som que possui elevado nível de intensidade é denominado som forte, ao passo que som de baixo nível de intensidade é conhecido como som fraco.

O timbre do som depende diretamente da fonte sonora e possui relação direta com o formato da onda. Instrumentos musicais diferentes apresentam timbres diferentes, dessa forma, o som de uma nota musical produzido, por exemplo, por um piano e por uma guitarra serão percebidos como diferentes mesmo que tenham a mesma amplitude e frequência. A Figura 3 ilustra essa característica.

Figura 3: Formato da onda produzida por diferentes instrumentos para uma mesma nota musical.



O sinal de áudio em um sistema elétrico pode ser representado por meio da variação da corrente ou da diferença de potencial elétrico. A corrente elétrica ou a diferença de potencial oscila exatamente na mesma proporção da taxa de energia acústica da onda sonora correspondente, de forma que o sinal elétrico corresponda, dentro dos limites do equipamento de áudio, fielmente ao sinal sonoro (DAVIS; JONES, 1990).

As unidades de efeitos para guitarras elétricas processam o sinal elétrico, correspondente à onda sonora gerada pelo instrumento musical, com o objetivo de modificar as propriedades do sinal sonoro, como, por exemplo, a altura, a intensidade e/ou o timbre, para obter diferentes características de som e aplicá-las nos mais variados estilos musicais.

2.3 PROCESSAMENTO DE SINAIS

A maioria dos sinais provenientes de sistemas físicos são sinais contínuos no tempo, como, por exemplo, a fala, a pressão exercida sobre superfícies, a temperatura, nível de líquido em tanques, etc. Porém, com o avanço tecnológico, tem sido cada vez mais vantajoso aplicar as técnicas de processamento digital de sinais, ao invés do processamento analógico, para modificar matematicamente ou interpretar dados derivados de sistemas físicos.

O processamento digital de sinais lida com a transformação de sinais que são discretos tanto na amplitude quanto no tempo, representados matematicamente por sequências de números, usando algum tipo de hardware digital (DINIZ; SILVA; NETO, 2014). Entre as principais vantagens em se trabalhar com processamento digital de sinais estão: (i) a flexibilidade proporcionada ao sistema, (ii) a maior imunidade a ruídos e (iii) a possibilidade de se obter um sistema mais compacto.

No entanto, a obtenção dessas vantagens só é possível devido ao avanço da tecnologia que proporcionou o surgimento de microcontroladores e conversores de sinais cada vez mais eficientes e condensados. Os conversores de sinais são sistemas empregados para converter a natureza do sinal (analógico para digital, ou vice-versa) para serem, posteriormente, analisados ou processados por algum equipamento. Existem basicamente dois tipos de conversores: o conversor analógico-digital (ADC - *Analog to Digital Converter*) e o conversor digital-analógico (DAC - *Digital to Analog Converter*).

O sinal analógico, convertido por um ADC, resulta em um sinal digital representado por números binários de n bits que correspondem às amostras do sinal em relação a uma tensão de referência. A resolução do conversor é diretamente proporcional ao número de bits, sendo que, quanto maior o número de bits, menor será o erro associado ao processo de

conversão do sinal. O DAC realiza a operação inversa do ADC, ou seja, converte sinais digitais em sinais analógicos com base em uma tensão de referência (ALVES, 2018). Dessa forma, grandezas analógicas podem ser processadas digitalmente, desde que sejam utilizados conversores de sinais adequados.

A fim de evitar problemas de perda de informações importantes ao sistema, deve ser definida corretamente a frequência de amostragem do sinal. De acordo com o teorema de amostragem de Nyquist, a frequência de amostragem de um sinal contínuo no tempo deve ser, pelo menos, duas vezes maior que a maior frequência do sinal a ser amostrado. Caso a frequência de amostragem seja menor do que a estabelecida pelo teorema de Nyquist, a reconstrução do sinal contínuo poderá resultar em um sinal de menor frequência do que a frequência do sinal original, caracterizando um fenômeno de *aliasing* (DINIZ; SILVA; NETO, 2014).

A quantidade de amostras processadas por um sistema digital é proporcional à quantidade de processamento matemático realizado pelo microcontrolador. Dessa forma, é desejável minimizar, sem que ocorram perdas de informações, a taxa de amostragem dos sinais contínuos no tempo. Para isso, são incluídos nos sistemas filtros *antialiasing*, responsáveis por excluir ou atenuar as altas frequências indesejadas do sinal.

2.3.1 Filtros de sinais

Filtros lineares podem ser definidos como sistemas seletores de frequências. Tais sistemas são utilizados com o objetivo de selecionar faixas de frequências desejadas e atenuar as frequências indesejadas (ruídos) do sinal. Os filtros podem ser analógicos ou digitais. Os filtros analógicos são projetados utilizando capacitores, resistores e/ou amplificadores operacionais, ao passo que os filtros digitais são projetados por meio de algoritmos computacionais processados por um processador ou controlador do sistema.

Alguns filtros de sinais, digitais ou analógicos, são caracterizados quanto a suas repostas no tempo e na frequência em filtros passa-baixa (LPF - *Low Pass Filter*) e filtros passa-alta (HPF - *High Pass Filter*). Os filtros LPF atenuam altas frequências do sinal, enquanto os HPF atenuam as baixas frequências. O projeto de filtros corresponde à determinação de seus parâmetros e/ou componentes.

De acordo com Oppenheim e Schaffer (2013, p. 294), “o projeto de filtros de tempo discreto corresponde à determinação dos parâmetros de uma função de transferência ou de uma equação de diferenças que aproxima uma resposta ao impulso ou uma resposta em

frequência dentro de tolerâncias especificadas”. De modo geral, o projeto de filtros de tempo discretos pode ser dividido em três etapas. Na primeira etapa, são realizadas as especificações das propriedades desejadas do sistema. Na segunda etapa, é feita a aproximação das especificações usando um sistema de tempo discreto casual. Por fim, é implementado, a partir das informações obtidas nas etapas anteriores, o sistema do filtro.

2.3.2 Processamento digital de sinais analógicos contínuos no tempo

A série de Fourier é muito aplicada nas ciências envolvidas com engenharia, matemática, música, processamento de imagens, etc. Com base na série de Fourier, expressa na Equação 1, Lathi (2007, p. 530) afirma que “um sinal periódico $x(t)$ com período T_0 pode ser descrito como a soma de senóides de frequência f_0 ($f_0=1/T_0$) e todas as suas harmônicas”.

$$x(t)=a_0+\sum_{n=1}^{\infty}(a_n \cos n\omega_0 t+b_n \operatorname{sen} n\omega_0 t) \quad (1)$$

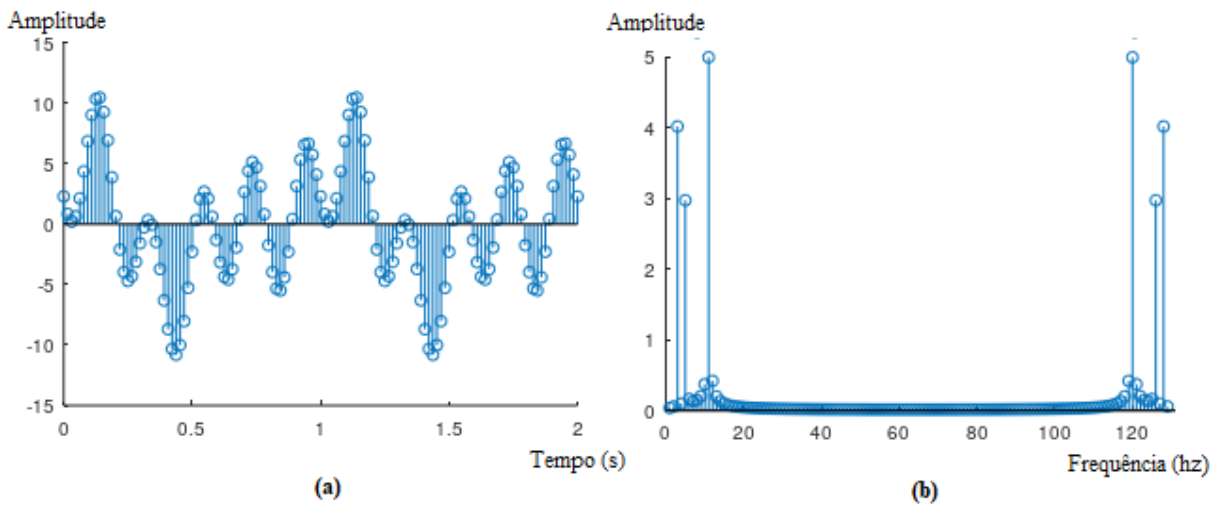
A frequência ω_0 , expressa em radianos, é denominada frequência fundamental, ao passo que a_0 corresponde ao valor médio da função $x(t)$ e os coeficientes a_n e b_n são valores de amplitudes das senoides presentes no somatório. As frequências múltiplas inteiras da frequência fundamental são chamadas de frequências harmônicas (LATHI, 2007).

A representação por série de Fourier de um sinal discreto no tempo com período N requer apenas N exponenciais harmonicamente complexas, enquanto que para sinais contínuos no tempo são necessárias infinitas senoides (OPPENHEIM; SCHAFER, 2013). Dessa forma, a representação por série de Fourier de uma sequência periódica $x[n]$ pode ser expressa pela Equação 2.

$$x[n]=\frac{1}{N}+\sum_{k=0}^{N-1}X[k]e^{j(2\pi/N)kn} \quad (2)$$

A Transformada Discreta de Fourier (DFT – *Discrete Fourier Transform*) equivale à decomposição de um sinal discreto no tempo como uma soma de senoides complexas discretas no tempo. As senoides contidas no somatório apresentam frequências, amplitudes e fases diferentes (LATHI, 2007). A Figura 4 (a) apresenta um sinal periódico e a Figura 4 (b) apresenta o espectro de frequência desse sinal calculado através da DFT.

Figura 4: (a) sinal discreto no tempo (b) sinal discreto no espectro da frequência.



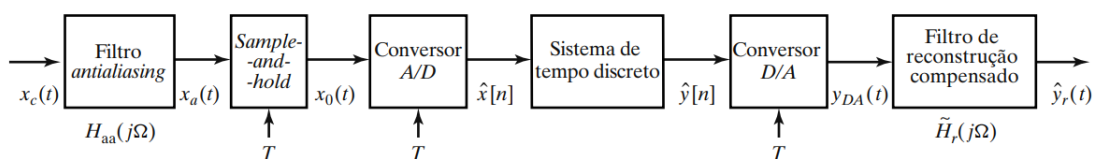
Fonte: Elaborada pelo autor.

A transformada de Fourier discreta, expressa pela Equação 3, é uma ferramenta extremamente valiosa para análise no domínio da frequência de sinais discretos no tempo. A utilização da DFT pode ser realizada por algoritmos computacionalmente eficientes, chamados transformadas rápidas de Fourier (FFT - *Fast Fourier Transform*). A utilização da FFT representa economia de multiplicações e adições complexas em relação à implementação direta da DFT (DINIZ; SILVA; NETO, 2014).

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n) \cdot e^{-j\omega n} \quad (3)$$

Sistemas analógicos contínuos no tempo podem ser processados eficientemente por sistemas digitais no domínio da frequência ou no domínio do tempo, desde que, sejam utilizados conversores de sinais, filtros equalizadores e controladores digitais adequados (conforme a Figura 5). O segundo bloco, denominado sample-and-hold, é um sistema do tipo amostragem e retenção, ou seja, responsável por manter o valor da amostra até que seja realizada a próxima (DINIZ; SILVA; NETO, 2014).

Figura 5: Processamento digital de sinais analógicos.



Fonte: Oppenheim e Schaffer (2013, p. 125).

Processamento de sinais de baixa amplitude normalmente requerem a implementação de um estágio extra: estágio de pré-amplificador. Esse estágio tem como objetivo amplificar o sinal de entrada e aumentar impedância de entrada do sistema de forma a aumentar a taxa de energia do sinal e evitar que ruídos trafeguem para os próximos estágios.

2.4 BREVE HISTÓRICO DA GUITARRA ELÉTRICA

A guitarra elétrica de corpo maciço, popularmente denominada guitarra no Brasil, constitui um instrumento musical pertencente à família de instrumentos de cordas, cujo som produzido é amplificado eletronicamente por meio de uma conexão elétrica com um amplificador de áudio. Esse instrumento musical originou-se da guitarra acústica, popularmente conhecida no Brasil como violão, com objetivo de obter maior projeção de volume do som produzido pelo instrumento sem a utilização de microfones.

A principal diferença entre a guitarra e o violão é que a guitarra possui corpo maciço e som amplificado eletronicamente, enquanto, no violão, o som é amplificado pelo corpo oco do instrumento que forma uma caixa acústica. De acordo com Souza (2018, p. 23), o violão é uma “evolução natural do alaúde árabe ou da khetara grega”.

Diversos autores se divergem quanto a qual foi e quando foi criada a primeira guitarra elétrica. Embora existam essas divergências, de acordo com Souza (2018), os créditos pela criação do instrumento são do músico George Beauchamp (1899-1941) e do engenheiro eletricista Adolph Rickenbacker (1886–1976) no desenvolvimento da Flying Pan, uma guitarra de corpo de alumínio, som amplificável e comercialmente viável, criada em 1931. Beauchamp e Rickenbacker também desenvolveram o primeiro dispositivo eletromagnético que captava as oscilações das cordas da guitarra elétrica com muita clareza. Esse dispositivo é denominado captador e seu princípio de funcionamento é baseado na lei de indução de Faraday. A Figura 6 traz a representação desse captador.

Figura 6: Primeiro modelo de captador eletromagnético.



Fonte: Rei da Les Paul (2015).

A lei de indução de Faraday, relatada em 1831, por meio de experimentos realizados Michael Faraday, evidencia que o movimento de uma espira, através de um campo magnético fixo, resulta em uma corrente elétrica na espira. Conseqüentemente, o movimento de um campo magnético em relação a uma espira fixa resulta em uma corrente elétrica na espira (QUEIRÓZ FILHO, 2017).

O captador eletromagnético é um transdutor devido à sua característica de converter a energia física em energia elétrica. O funcionamento desse captador é baseado nessas características do eletromagnetismo, sendo, até o momento, o sistema mais utilizado para a captação de vibrações das cordas do instrumento. Esse captador é constituído de um ímã permanente, responsável pela magnetização das cordas da guitarra elétrica, uma vez que elas são feitas de materiais magnetizáveis. Ao redor dos ímãs dos captadores, existem várias espiras de cobre que têm as terminações conectadas a um resistor interno à guitarra. O resistor interno, por sua vez, é conectado paralelamente ao conector do sinal de saída de áudio do instrumento. Quando a corda da guitarra entra em movimento oscilatório, o campo magnético presente oscila de acordo com a frequência de vibrações das cordas, induzindo corrente elétrica nas espiras dos captadores (ALVES, 2018).

Dessa forma, as vibrações das cordas da guitarra alteram o campo magnético presente e induzem corrente elétrica, conforme a lei de indução de Faraday, nas espiras contidas no ímã do captador. O sinal elétrico é posteriormente transmitido para uma unidade de efeitos ou diretamente para um amplificador, sendo reproduzido por alto falantes.

Com as guitarras atuais, por meio da amplificação do sinal de corrente elétrica induzida nas espiras dos captadores, é possível obter grande amplitude e sustentação das notas

musicais. No entanto, segundo Rocha (2011), apenas a partir do lançamento da guitarra Gibson ES-150, em 1936, que finalmente foi atingido o objetivo de amplificação do som da guitarra de maneira satisfatória do ponto de vista sonoro, industrial e mercadológico.

Os amplificadores para guitarra datados da década de 1940 funcionavam a base de válvulas e possuíam alto-falantes pequenos. Devido à sua baixa potência o sinal de áudio era frequentemente saturado, provocando uma distorção não desejada no som produzido pelo instrumento. No decorrer do tempo, foram desenvolvidos amplificadores para guitarra que possuem, em sua maioria, alta potência, opções de controle de timbre sonoro e outros efeitos de áudio (ROCHA, 2011).

Com o surgimento de novos estilos musicais, como, por exemplo, do blues e do rock, passaram a ser cada vez mais comum a inserção de efeitos sonoros de distorção, da modulação do ganho e da frequência do sinal, entre outros, para alterar o som dos instrumentos. Segundo Rocha (2011), antes do surgimento das unidades de efeitos, muitos guitarristas utilizavam métodos não ortodoxos para provocar efeitos em seus amplificadores, tais como chutes e cortes no alto-falante para obter efeitos de distorção.

Atualmente, várias empresas fabricam unidades de efeitos, tanto digitais quanto analógicas, para obtenção dos mais variados sons em diferentes instrumentos. As unidades de efeitos para guitarra são normalmente ligadas em série, entre a guitarra e o amplificador, de forma a modificar o sinal elétrico correspondente à vibração das cordas da guitarra e amplificá-lo posteriormente.

2.5 UNIDADES DE EFEITOS

Segundo Queiróz Filho (2017, p. 38), “o pedal [unidade] de efeito para guitarra é um equipamento eletrônico que normalmente é utilizado para alterar o som natural de uma guitarra elétrica que, em sua maioria, acompanha as suas próprias fontes elétricas e cabos de força”. Em outras palavras, os pedais de efeitos são equipamentos de processamento, analógicos e/ou digitais, de sinais elétricos que correspondem ao som natural do instrumento.

O principal objetivo do processamento do sinal em unidades de efeito para guitarra é obter características sonoras diferentes através da alteração da forma, da amplitude e/ou da frequência do sinal de áudio. A indústria musical tem aderido, cada vez mais, à utilização de efeitos de áudios, tanto para alterar as características sonoras da guitarra elétrica quanto para modificar voz e sons de outros instrumentos musicais.

Atualmente, existem no mercado diversos tipos de unidades de efeitos para guitarras elétricas, como, por exemplo: os pedais de efeitos de ambiência, de ganho, de modulação e de tonalidade. Os efeitos de ambiência têm como objetivo, normalmente, criar ecos ou a sensação de reverberação do som no ambiente. O efeito de eco é criado por unidades de efeitos de *Delay*, enquanto o efeito de reflexões do som é obtido por meio da utilização de unidades de efeito de *Reverb*. O efeito de eco é obtido a partir da repetição do sinal original do instrumento em espaços fixos no tempo, ao passo que o efeito da reverberação é atingido através da inclusão de cópias atrasadas do sinal que vão decaindo a intensidade com o tempo (ROCHA, 2011).

Os efeitos de ganho aumentam a amplitude do sinal elétrico criando sinais de *clipping*, de forma a distorcer e gerar um som saturado nos amplificadores. Os efeitos de ganho, normalmente, atuam no sinal em dois estágios. No primeiro estágio, o sinal de entrada é amplificado, enquanto no segundo estágio ocorre a saturação, ou seja, o corte na amplitude do sinal. São exemplos de unidades de efeitos de ganho, as unidades de *Overdrive*, *Distortion*, *Fuzz* e *Boost* (QUEIRÓZ FILHO, 2017).

Os efeitos de modulação criam um novo sinal por meio do sinal original, gerando o efeito a partir de repetições do sinal em unidades de tempo desigualmente espaçadas e/ou levemente desafinadas. Tais efeitos alteram a amplitude, a frequência e a fase do sinal. O efeito de *Chorus* é um exemplo de modulação, resultante da geração de um sinal com um pequeno atrasado e pequenas variações na frequência (ALVES, 2018). Por outro lado, há efeitos de modulação que modulam apenas a frequência, como, por exemplo, o vibrato, e que modulam apenas a amplitude, tal como o efeito de tremolo.

Os efeitos de tonalidade alteram a equalização do som de forma a modificar o timbre do som original do instrumento. Como exemplo de efeitos de tonalidade existe o efeito *wah-wah*. As unidades de efeitos de *wah-wah* com pedal de expressão possibilita a alteração manual da frequência do sinal por meio de pedais de expressão (ROCHA, 2011).

Os efeitos dinâmicos, por sua vez, são efeitos de controle da amplitude do sinal. Entre os pedais de efeitos dinâmicos mais utilizados atualmente estão os pedais *Compressor* e *Sustain*. Todos esses efeitos podem ser obtidos através da utilização de uma unidade digital multiefeitos. As unidades digitais multiefeitos incluem diversos efeitos para guitarra elétrica, podendo incluir, em alguns casos, simulações de amplificadores e efeitos de *loop*.

As unidades digitais de multiefeitos podem ser implementadas utilizando sistemas microcontrolados/processados eficientes para modificar e analisar os sinais. A unidade GT-1000 da BOSS, por exemplo, é um unidade digital multiefeitos para guitarra com

processamento de 32 bits, conversor digital-analógico e analógico-digital com 32 bits de resolução e taxa de amostragem de 96 kHz.

2.6 SISTEMAS MICROCONTROLADOS

Por microcontrolador compreende-se um circuito integrado composto por unidade de processamento, memórias de leitura, escrita de dados, armazenamento de *firmware*, periféricos e portas programáveis de entrada e saída. Esses componentes se diferem de acordo com a arquitetura de cada microcontrolador. Os sistemas microcontrolados estão presentes em máquinas e dispositivos industriais, sistemas de supervisão, eletrodomésticos, controle de processos e diversas outras aplicações (AGNOL, 2018).

No que concerne a esse trabalho, é preciso identificar os requisitos necessários para a realização do projeto, como, por exemplo, a frequência de operação do microcontrolador e os protocolos de comunicação presentes na plataforma. O Quadro 1 apresenta algumas informações importantes das plataformas Arduino e ESP.

Quadro 1: Arduino UNO REV3, Arduino MEGA 2560, ESP32 e ESP8266.

Descrição	Arduino UNO REV3	Arduino MEGA 2560	ESP32	ESP8266
Alimentação	5 V	5 V	2,2-3,3 V	2,2-3,3 V
Corrente de consumo	Média de 15 mA	Média de 70 mA	Média de 80 mA	Média de 80 mA
Microcontrolador	ATmega328P 8-bit	ATmega2560 8-bit	Xtensa® Dual-Core 32-bit LX6	Tensilica® L106 ultra-low power 32-bit
Frequência de Operação	Até 16 MHz	Até 16 MHz	80-240 MHz	80-160 MHz
Memória FLASH	32 kB	256 kB	4-16 MB externa	4 MB
Pinos de I/O	14 pinos (6 PWM)	54 pinos (15 PWM)	34 pinos (16 PWM)	13 pinos (9 PWM)

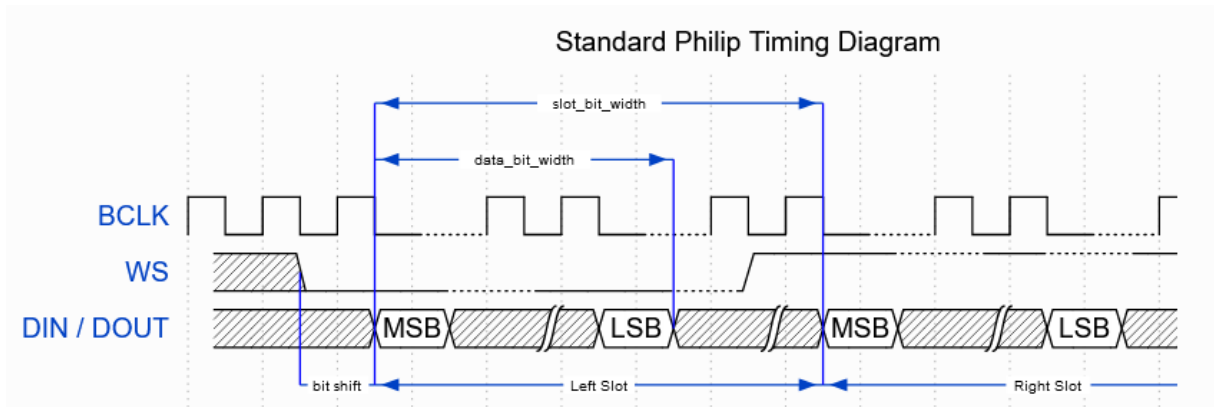
ADC	6 canais ADC de 10 bits	16 canais ADC de 10 bits	18 canais ADC de 12 bits	1 canal ADC de 10 bits
DAC	Nenhum	Nenhum	2 canais DAC de 8 bits	Nenhum
Interfaces	I2C, SPI, UART e LED PWM	I2C, SPI, 4 UART e LED PWM	SPI, SDIO, LED PWM, Motor PWM, I2S e IR	SPI, SDIO, LED PWM, I2S e I2C
WiFi integrado	Nenhum	Nenhum	2,4 GHz, 802.11 b/g/n/e/i (802.11n @ 2.4 GHz até 150 Mbps)	2,4GHz com suporte a WPA e WPA2, 802.11 b/g/n, P2P e soft-AP
Bluetooth integrado	Nenhum	Nenhum	Bluetooth Low Energy v4.2	Nenhum

Fonte: xProjetos.net (2019).

Além dos plataformas apresentadas no quadro anterior, a família de microcontroladores STM32, baseada no processador ARM Cortex-M3 de 32 bits, também oferece uma estrutura para criação de uma vasta quantidade de sistemas. O Mapple Mini, plataforma da família STM32, possui frequência de operação de 72 MHz, contém 128 kB de memória Flash, 20 kB de memória RAM, 34 GPIOs (sendo 12 PWM) e 9 canais ADC de 12 bits (FERNANDO K. TECNOLOGIA, 2018).

O protocolo de comunicação Som de Circuito Inter Integrado (I2S - *Inter-Integrated Circuit Sound*) criado pela Philips Semicondutores, presente no ESP32 e em outros microcontroladores, permite enviar dados digitais em alta resolução e frequência, sendo assim, muito utilizado em projetos de áudio digitais. Ele é formado por três linhas de sinais (conforme a Figura 7). O sinal BCLK é utilizado para a sincronização entre o transmissor e o receptor dos dados. O sinal WS indica se os dados referem-se ao canal direito (quando em nível lógico alto) ou esquerdo (quando em nível lógico baixo). E por fim, o sinal DIN/OUT contém os dados seriais que estão sendo enviados ou recebidos pelo equipamento. Alguns dispositivos requerem a utilização de mais uma linha de *clock* denominada *clock* do sistema.

Figura 7: Linha de sinais do protocolo I2S.



Fonte: Espressif Systems (2022).

3 METODOLOGIA

Neste capítulo, apresenta-se os métodos e os recursos de análise que foram empregados para realizar este trabalho. Quanto a sua natureza, esta pesquisa é classificada, metodologicamente, como aplicada, uma vez que, a partir de uma situação concreta, se buscou desenvolver um protótipo cuja aplicação poderá gerar melhorias no contexto estudado. Quanto à forma de abordagem, trata-se de uma pesquisa qualitativa, dado que os resultados obtidos por meio do protótipo foram descritos e interpretados, de modo a compreender o tema. Quanto aos objetivos, trata-se de uma pesquisa exploratória, na medida em que busca, formulando hipótese, compreender de forma aprofundada o problema. Por fim, quanto aos procedimentos técnicos, lançou-se mão da pesquisa bibliográfica, a fim de melhor delimitar o tema, e da pesquisa experimental, uma vez que, pela montagem do protótipo, foi possível lidar com as variáveis que alteram os resultados obtidos, propondo um controle que produza o efeito de tremolo. Para a execução desta proposta de pesquisa, 4 etapas foram contempladas.

A primeira etapa consistiu o estudo teórico sobre os efeitos de áudio; os conceitos de conversão analógica-digital e digital-analógica de sinais elétricos; o detalhamento de processamento de sinais; e as plataformas microcontroladoras que tenham *hardwares* suficientemente eficientes. Por meio desse estudo, foi possível melhor compreender os conceitos abordados, bem como as variáveis que atuam no processo.

A segunda etapa compreendeu, após estudo sobre as plataformas microcontroladoras, na escolha de uma que atenda ao projeto: Esp-32 DevKit v1. Após isso, foi realizada análise dos protocolos de transferência de dados disponível na plataforma escolhida objetivando definir um que atenda melhor o projeto: I2S.

A terceira etapa equivaleu à criação do protótipo de uma unidade digital de efeito para guitarra elétrica. Essa etapa do processo incluiu a definição do modelo físico da unidade digital de efeito, dos materiais a serem utilizados, da montagem do circuito eletrônico e da implementação de algoritmos responsáveis pelo processamento digital dos sinais. Foram definidos os módulos conversores de sinais CJMCU-1808 e CJMCU-1334, o amplificador de áudio diferencial AN7523N, o amplificador operacional LM224N, o circuito integrado inversor 74HC04, o display LCD com módulo I2C acoplado, *push buttons*, cristal oscilador de 12 MHz e os demais componentes: capacitores e resistores, a serem utilizados na construção da unidade.

A quarta e última etapa consistiu na testagem e verificação da qualidade do sinal nos diferentes estágios de processamento de sinais. Para essa etapa, foi empregado o osciloscópio

Hantek 2D72, o gerador de sinais Minipa MFG-4200 e o analisador lógico Analyzer de 8 canais na realização da leitura dos sinais de saída e verificação do comportamento do sistema, além de testes do som obtido através da utilização da unidade de efeito em um sistema composto por guitarra e amplificador.

3.1 ESP32

Para o desenvolvimento do protótipo foi necessário definir uma plataforma microcontroladora que tenha *hardware* suficientemente eficiente. Considerando essa necessidade partiu-se para escolha da plataforma microcontroladora ESP32 DevKit v1, comumente chamada de ESP32. A figura 8 apresenta a plataforma microcontroladora definida.

Figura 8: Plataforma ESP32 DevKit v1.



Fonte: Elaborada pelo autor.

A ESP32 trabalha com o microprocessador *dual core* XTensa 32-Bits LX6. Tal dispositivo foi selecionado por seu processador possuir maior frequência de *clock* do que os das demais plataformas analisadas e conter dois canais de interface I2S que possibilitam transferência serial de dados em alta velocidade.

O microcontrolador abordado possui dois núcleos de Central Única de Processamento (CPU - *Central Processing Unit*), de frequência de *clock* ajustável de 80 MHz a 240 MHz, que podem ser controlados individualmente. Além disso, a plataforma possui interface UART (3 canais), SPI (3 canais), I2C (2 canais) e I2S (2 canais).

Apesar do seu tamanho relativamente pequeno (27,5 x 51,0 x 7,0 mm), a plataforma dispõe de 34 GPIOs, sendo possível configurar 18 canais para realizar conversão analógica-digital com até 12 bits de resolução e 2 canais de conversão digital-analógica de 8 bits de resolução.

Após analisadas essas informações, foi realizada uma relação entre a memória disponível para uso no ESP32 e o tamanho reservado por diferentes tipos variáveis. O ESP32 analisado possui 448 kB de memória ROM, 520 kB de memória SRAM, duas memórias de 8 kB RTC e 4 MB de memória flash. O quadro 2 apresenta o espaço ocupado em disco pelas principais variáveis.

Quadro 2: Tipo de variável x Espaço ocupado em disco.

Tipo de Variável	Espaço ocupado
byte	8 bits
bool	1 byte
int	4 bytes
float	4 bytes
double	8 bytes

Fonte: Elaborado pelo autor.

Os sinais digitais de áudio em um sistema de processamento digital são normalmente armazenados em vetores de dados do tipo real. Para cada variável do tipo float utilizada são reservados 4 bytes (32 bits) na memória SRAM do ESP32. Portanto, é possível armazenar centena de milhares de números reais definidos como variáveis do tipo float utilizando somente a memória SRAM da plataforma.

3.1.1 IDE e linguagem de programação

O Ambiente de Desenvolvimento Integrado (*IDE - Integrated Development Environment*) é um software utilizado no desenvolvimento aplicações. O ESP32 pode ser programado em diferentes linguagens de programação, como, por exemplo, em linguagem C/C++ por meio da Arduino IDE e na linguagem Python através da utilização do *firmware* MicroPython e de uma Python IDE.

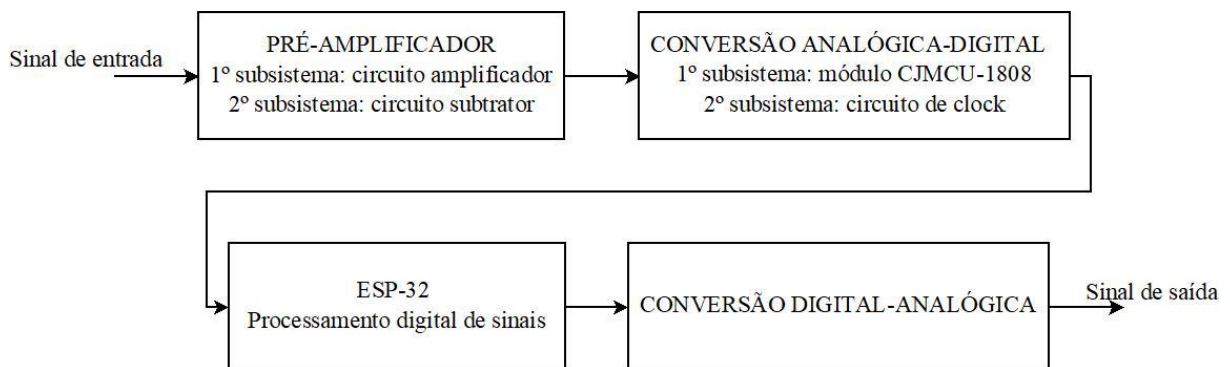
Nesse projeto foi empregada a linguagem de programação C e utilizada a Arduino IDE para configurar o ESP32 e desenvolver os algoritmos propostos. A frequência de operação do *clock* do processador do dispositivo foi definida em 240 MHz.

3.2 ETAPAS DA CONSTRUÇÃO DA UNIDADE DE EFEITO

O microcontrolador selecionado possui 18 canais de ADC de 12 bits de resolução e 2 canais de DAC de 8 bits de resolução. No entanto, sistemas de processamento digital de áudio de alta qualidade requerem a utilização de conversores de sinais de maior resolução e filtros para eliminação de ruídos. Sendo assim, foram utilizados módulos codificadores e decodificadores de sinais externos ao ESP32 e que possuem filtros de sinais incluso.

Além disso, visto que o sinal da guitarra elétrica tem amplitude muito baixa, a unidade de efeito implementada é composta por um circuito pré-amplificador cujo funcionamento pode ser dividido em dois subsistemas (conforme exemplificado na figura 9).

Figura 9: Diagrama em blocos da unidade de efeito.



Fonte: Elaborada pelo autor.

3.2.1 Sistema de pré-amplificador

O sistema de pré-amplificador desenvolvido é um amplificador de sinais de áudio advindos da guitarra elétrica anterior à realização da conversão analógica-digital. Tal sistema tem como objetivo amplificar o sinal e evitar que ruídos trafeguem e sejam processados pelas próximas etapas do circuito.

Para implementação do circuito pré-amplificador foram utilizados os componentes eletrônicos listados a seguir, além dos instrumentos utilizados para realização de testes: osciloscópio e gerador de funções.

- 1 circuito integrado AN7523N;
- 1 circuito integrado LM224N;
- 1 capacitor de poliéster de 100 nF;
- 1 capacitor eletrolítico de 1 μ F;

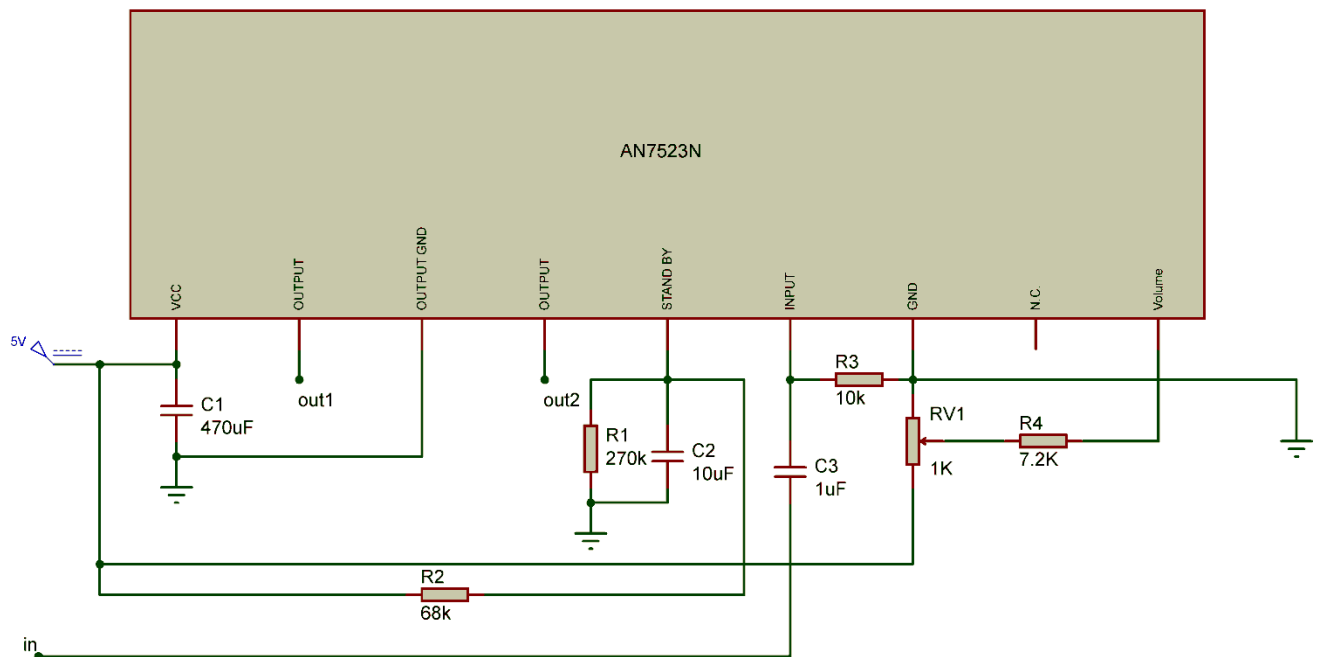
- 1 capacitor eletrolítico de 10 μF ;
- 1 capacitor eletrolítico de 470 μF ;
- 1 resistor de 10 Ω ;
- 1 resistor de 3,3 $\text{k}\Omega$;
- 1 resistor de 3,9 $\text{k}\Omega$;
- 1 resistor de 10 $\text{k}\Omega$;
- 1 resistor de 47 $\text{k}\Omega$;
- 1 resistor de 68 $\text{k}\Omega$;
- 1 resistor de 270 $\text{k}\Omega$;
- 4 resistores de 1 $\text{M}\Omega$;
- 1 fonte de alimentação DC de 5 V;
- 1 protoboard.

O sistema de pré-amplificador pode ser dividido em dois subsistemas, sendo eles: subsistema do amplificador diferencial e subsistema do circuito subtrator.

A subsistema do amplificador diferencial processa-se no circuito integrado AN7523N, um amplificador de áudio diferencial composto por amplificadores operacionais. Tal circuito integrado possui faixa de tensão de alimentação de 3,5 V à 13,5 V, sendo que quanto maior a tensão de alimentação maior será a capacidade do circuito amplificar o sinal de entrada sem que haja corte na forma de onda de saída. Dessa forma, analisando a faixa de alimentação dos próximos estágios da unidade de efeitos, 3,3 V e 5 V, foi definida uma tensão de alimentação de 5 V para o amplificador de áudio AN7523N.

O circuito do amplificador de áudio diferencial desenvolvido é constituído de um capacitor (C1) de desacoplamento, um resistor (R3) que compõe a impedância de entrada do sistema e um potenciômetro (RV1) disposto para controle de ganho do sinal de áudio, sendo utilizado para controlar o volume do som processado pela unidade de efeito. Tal configuração é indicada pelo fabricante do circuito integrado na folha de dados do componente AN7523N. A figura 10 apresenta o circuito implementado.

Figura 10: Esquemático circuito amplificador diferencial.



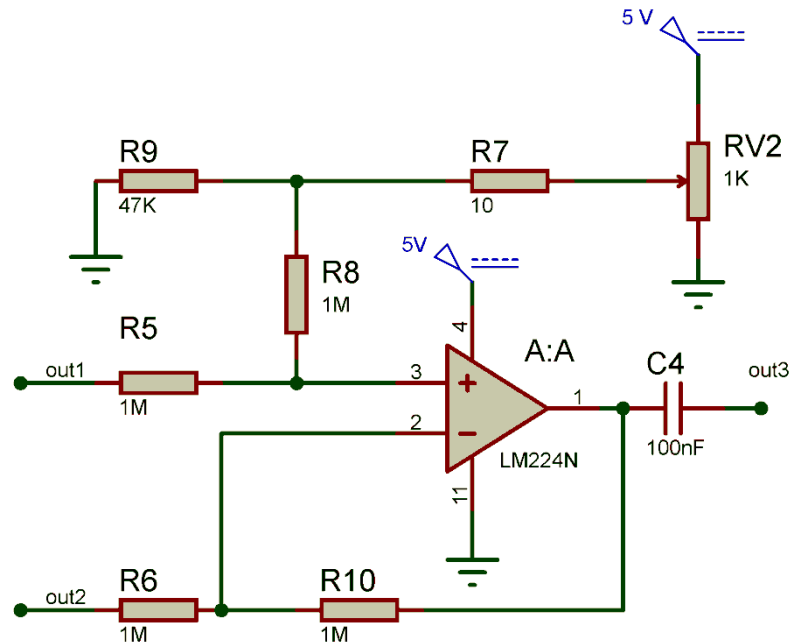
Fonte: Elaborada pelo autor.

O sinal de saída do amplificador AN7523N é um sinal de tensão diferencial (out1 – out2). No entanto, o sinal de entrada do sistema de conversão analógico-digital tem como referência o nó terra. Dessa forma, para que o sinal amplificado seja identificado corretamente, foi desenvolvido um circuito subtrator pelo qual o sinal diferencial é processado. O circuito subtrator implementado é composto pelo amplificador operacional LM224N.

O circuito subtrator desenvolvido possui ganho unitário. O amplificador operacional LM224N não foi utilizado como amplificador do sinal de áudio em razão de não possuir ganho linear em todo espectro de frequência audível. Portanto, a função desse circuito foi apenas transformar o sinal de áudio, anteriormente diferencial (out1 – out2), em uma única saída (out3) referenciada ao nó terra do sistema.

O LM224N presente no circuito subtrator não possui tensão de alimentação negativa. Por consequência, para não haver corte na forma de onda de saída (out3), o circuito subtrator é composto por um gerador de *offset*, o qual é ajustado a partir do trimpot (RV2) contido no circuito (conforme apresentado na figura 11).

Figura 11: Esquemático circuito subtrator.



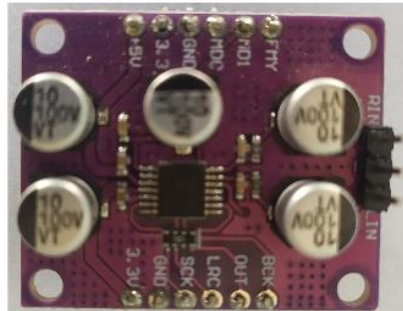
Fonte: Elaborada pelo autor.

Após o sinal ser amplificado, somado a um *offset* e referenciado ao nó terra, ele é processado por um capacitor (C4) para eliminação de sua componente de corrente contínua. Em seguida, o sinal é enviado para ser discretizado pelo conversor de sinal analógico-digital.

3.2.2 Sistema de conversão analógica-digital

Para o desenvolvimento do sistema de conversão analógica-digital da unidade de efeito se fez imprescindível a definição de um circuito integrado ADC de alto desempenho e com interface para comunicação em alta velocidade com o ESP32. À vista disso, foi selecionado o módulo CJMCU-1808 que possui interface de comunicação I2S. Tal módulo apresentado na figura 12.

Figura 12: Módulo CJMCU-1808.



Fonte: Elaborada pelo autor.

O módulo CJMCU-1808 é composto pelo circuito integrado conversor analógico-digital PCM-1808. Os principais recursos e requisitos para operação correta desse conversor são listados a seguir.

- conversão A/D Delta Sigma com 24 bits de resolução;
- frequência de sobreamostragem: 64x;
- formato de dados: 24 bits I2S ou 24 bits justificado à esquerda;
- filtro *antialiasing*;
- frequência de amostragem de 8 kHz à 96 kHz;
- *clock* do sistema: 256 fs, 384 fs ou 512 fs;
- dual fonte de alimentação: 5 V para alimentação do circuito analógico e 3,3 V para alimentação circuito digital;
- sinal de entrada de até 3 Vpp.

O CJMCU-1808 permite controlar o seu modo de operação, possibilitando configurá-lo para atuar como mestre ou escravo de um sistema de transferência serial de dados e o formato da transferência de dados conforme definido no protocolo de comunicação I2S ou justificado à esquerda. Essas configurações são realizadas via *hardware* pela alimentação digital nas portas de entrada MD0, MD1 e FMY presentes no módulo ADC (conforme é apresentado no quadro 3).

Quadro 3: Modo de operação CJMCU-1808.

MD0	MD1	Modo de operação
Baixo	Baixo	Modo escravo (256 fs, 384, fs, 512 fs – auto detecção)
Baixo	Alto	Modo mestre (384 fs)
Alto	Alto	Modo mestre (256 fs)
Alto	Baixo	Modo mestre (512 fs)
FMY		Formato de dados
Alto		Justificado à esquerda
Baixo		I2S

Fonte: Elaborado pelo autor.

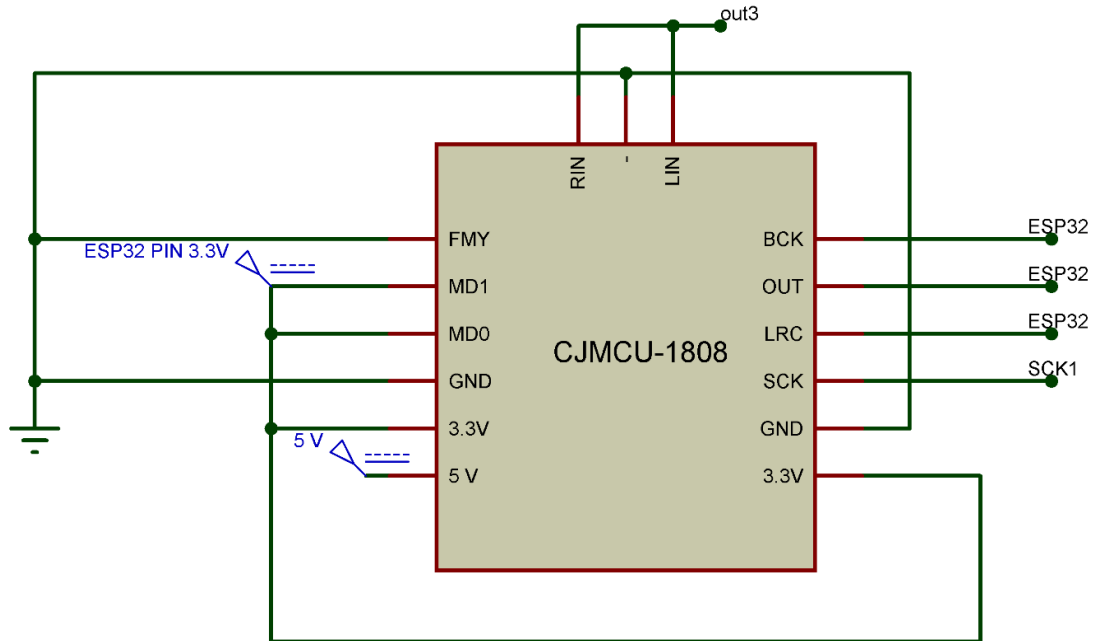
O PCM-1808 suporta 256 fs, 384 fs e 512 fs como frequência de *clock* do sistema, sendo que fs corresponde à frequência de amostragem de sinais. O circuito integrado conversor ADC selecionado possui um circuito interno que detecta qual a frequência do *clock* do sistema quando o conversor está operando no modo escravo. Em contrapartida, quando o dispositivo está operando no modo mestre, a frequência de *clock* do sistema deve ser gerada por um dispositivo externo e identificada através da alimentação digital nas portas de entrada MD0 e MD1 (conforme quadro 3).

A interface entre o módulo CJMCU-1808 e a plataforma ESP32 ocorre através das portas BCK, LRC e OUT do módulo. A porta BCK corresponde ao sinal de *clock* utilizado para sincronização na transferência serial de dados, a porta LRC à seleção do canal esquerdo (quando em nível lógico baixo) ou canal direito (quando nível lógico alto) e a porta OUT à saída serial dos sinais filtrados e discretizados pelo módulo conversor analógico-digital definido.

Dado a implementação do circuito (representado na figura 13), o CJMCU-1808 foi configurado para operar nos seguintes modos:

- modo de operação: mestre;
- formato de dados: 24 bits I2S;
- tamanho da palavra: 64 bits;
- frequência de *clock* do sistema: 256 fs;
- transferência de dados em estéreo (canal direito e esquerdo).

Figura 13: Esquemático circuito CJMCU-1808.



Fonte: Elaborada pelo autor.

É importante destacar que as portas BCK e LRC são portas de saída quando o PCM-1808 está operando no modo mestre. Por outro lado, quando ele está operando no modo escravo BCK e LRC se tornam portas de entrada. Além disso, em ambos os modos de operação, o *clock* do sistema deve ser gerado por um dispositivo externo e conectado à porta de entrada SCK (-0,3 a 6,5 V). Posto isto, foi necessário o desenvolvimento de um circuito de clock extrínseco ao módulo CJMCU-1808.

De acordo com teorema de Nyquist a frequência de amostragem deve ser superior à duas vezes a maior frequência do sinal amostrado para que não ocorra o fenômeno de *aliasing*. Com isso, considerando o espectro de frequência audível de até 20 kHz, a frequência de amostragem da unidade de efeito deve ser superior a 40 kHz.

A frequência de amostragem do ADC abordado corresponde 1/256 da frequência de *clock* do sistema na configuração implementada (apresentada na figura 13). Sendo assim, o *clock* do sistema deverá ser superior à 10,24 MHz no modo estéreo.

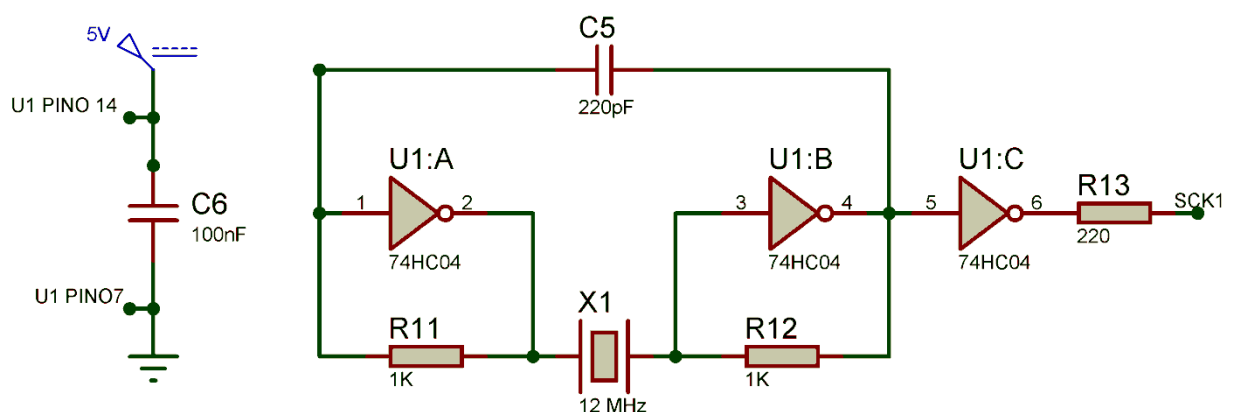
Para a implementação do circuito de *clock* foi utilizado os componentes eletrônicos abaixo.

- 1 circuito integrado inversor 74HC04;
- 1 capacitor cerâmico de 220 pF;
- 1 capacitor de poliéster de 100 nF;

- 1 resistor de 20 Ω ;
- 2 resistores de 1 k Ω ;
- 1 cristal oscilador de quartzo de 12 MHz;
- 1 fonte de alimentação de 5 V (a mesma utilizada nos estágios anteriores);
- 1 protoboard.

O circuito de *clock* do sistema do ADC foi implementado utilizando um cristal oscilador quartzo de 12 MHz, em razão de ser encontrado facilmente no mercado, e três portas inversoras do circuito integrado 74HC04. O capacitor C2 é um capacitor de desacoplamento que tem como objetivo filtrar o sinal de alimentação do circuito. O circuito de clock do sistema desenvolvido é apresentado na figura 14.

Figura 14: Esquemático circuito de clock do sistema CJMCU-1808.



Fonte: Elaborado pelo autor.

Dessa forma, o sinal de saída do circuito de clock (SCK1) é um sinal de onda quadrada de 12 MHz. Tal sinal de saída é conectado à porta SCK do CJMCU-1808 configurando o circuito integrado conversor PCM-1808 para operar em uma frequência de amostragem de 46,875 kHz no modo estéreo. No entanto, o sinal de saída da guitarra elétrica é um sinal de áudio mono, ou seja, de apenas um canal de áudio, que, após amplificado, é conectado aos dois canais de entrada do ADC (RIN e LIN) simultaneamente. Dessa forma, são realizadas duas leituras de dados a cada oscilação do sinal SCK1 resultando na transmissão de dados de áudio com o dobro da frequência de amostragem definida no modo estéreo: 93,75 kHz.

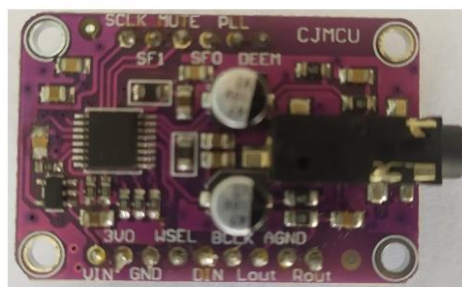
Após os dados serem transferidos para o microcontrolador e processados digitalmente, os mesmos são enviados ao sistema de conversão digital-analógica para que sejam

reconstruídos analogicamente antes de serem amplificados por um amplificador de maior ganho ou enviados a um fone de ouvido.

3.2.3 Sistema de conversão digital-analógica

Para o desenvolvimento do sistema de conversão digital-analógica da unidade de efeito se fez a definição de um circuito integrado DAC de alto desempenho e com interface para comunicação I2S para transferência de dados em alta velocidade. À vista disso, foi selecionado o módulo CJMCU-1334. Tal módulo é apresentado na figura 15.

Figura 15: Módulo CJMCU-1334.



Fonte: Elaborado pelo autor.

O módulo CJMCU-1334 é composto pelo circuito integrado conversor analógico-digital UDA-1334. Os principais recursos e requisitos para operação correta desse módulo são listados a seguir.

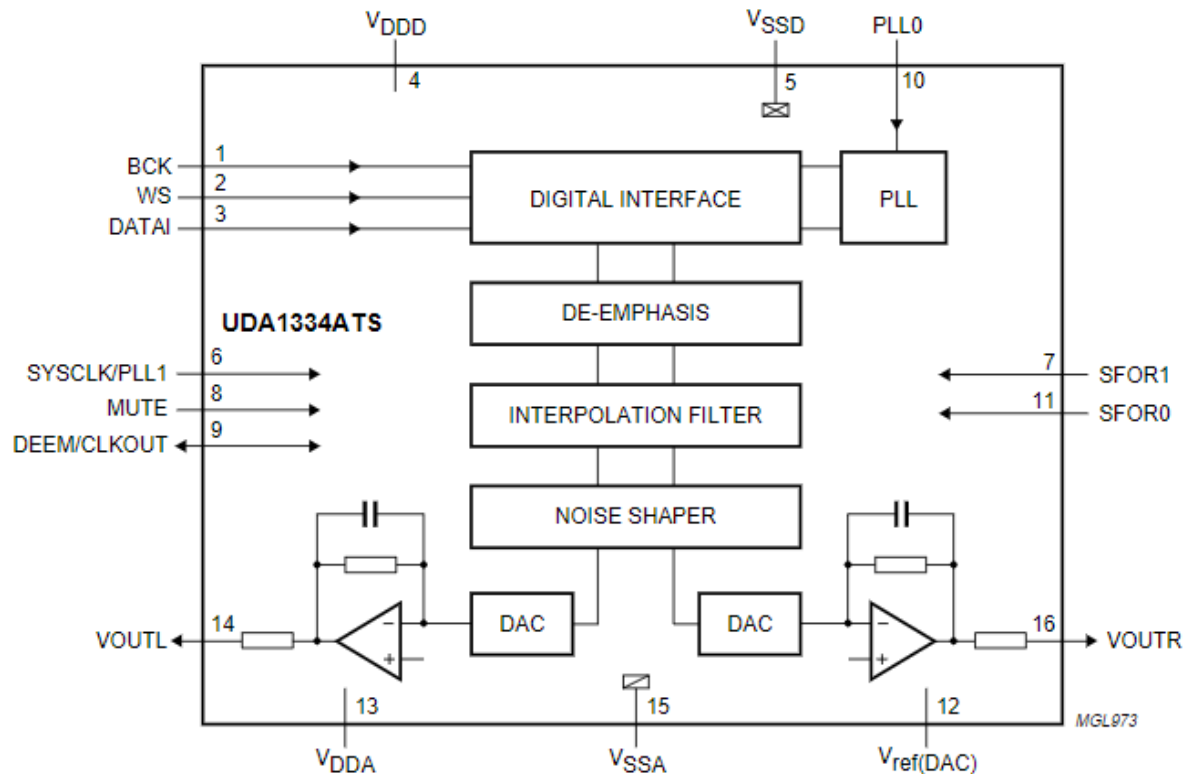
- conversão D/A de com comprimento de palavra de 24 bits;
- formato de dados: I2S (24 bits de resolução) ou justificado à esquerda (16, 20 ou 24 bits de resolução);
- frequência de amostragem de até 100 kHz;
- faixa de tensão de alimentação: 3,3 V à 5 V;
- modo de operação: mestre ou escravo.

O circuito integrado UDA-1334 é capaz de atuar no modo de operação mestre ou escravo. No modo mestre o conversor digital-analógico utiliza um circuito Loop de Fase Bloqueada (PLL - *Phase Locked Loop*) interno para a geração de sinais de *clock*. Por outro

lado, quando operado no modo escravo os sinais BCK e WS devem ser gerados por um dispositivo externo ao circuito integrado DAC.

Além de realizar a conversão de sinais, o UDA-1334 possui amplificadores operacionais internos que amplificam os sinais de áudio de saída e filtros para eliminação de ruídos (conforme figura 16).

Figura 16: Esquemático circuito integrado UDA-1334.

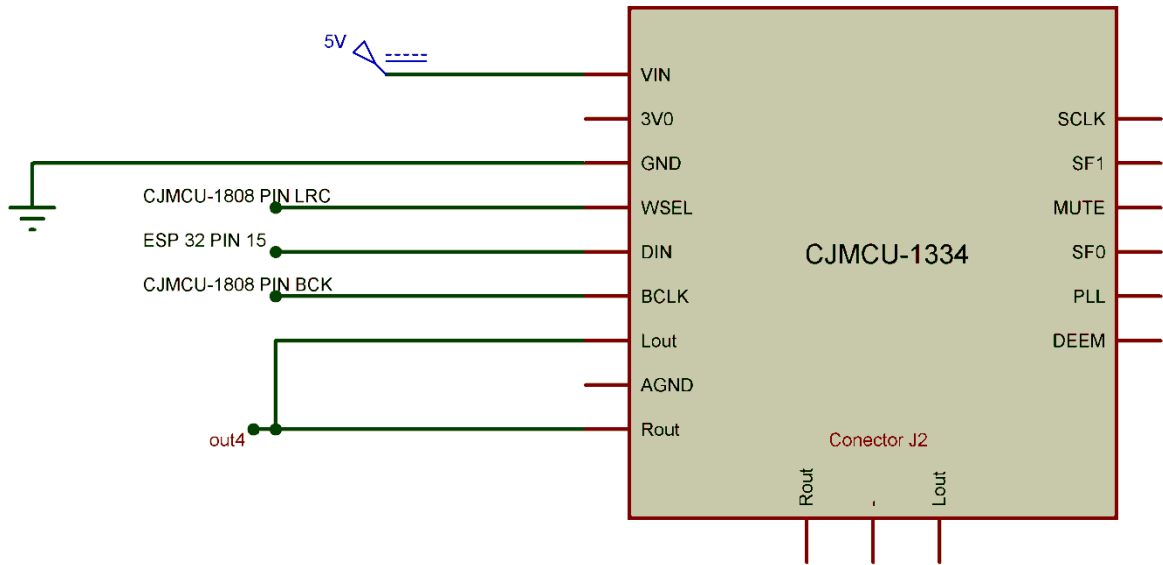


Fonte: NXP SEMICONDUCTORS (2000).

O circuito eletrônico implementado para o sistema de conversão analógica-digital é apresentado na figura 17. Nessa composição o módulo está configurado para operar em modo escravo e o formato de dados em 24 bits transferidos através do protocolo I2S. Dessa forma, os sinais BLCK e WS são fornecidos por um dispositivo externo mestre.

O dispositivo mestre da transferência de dados entre o ESP32 e o CJMCU-1334 é o dispositivo que fornece os sinais de clock para sincronização, ou seja, o CJMCU-1808. Dessa forma, a frequência de amostragem será a mesma definida para o sistema conversor analógico-digital, 93,75 kHz.

Figura 17: Esquemático circuito conversor digital-analógico.



Fonte: Elaborada pelo autor.

3.2.4 Interface eletrônica

A unidade de efeito para guitarra elétrica implementada possui interface com o usuário via apresentação de dados em tela e parametrização do efeito via menu. O circuito eletrônico implementado da interface possui os seguintes componentes eletrônicos:

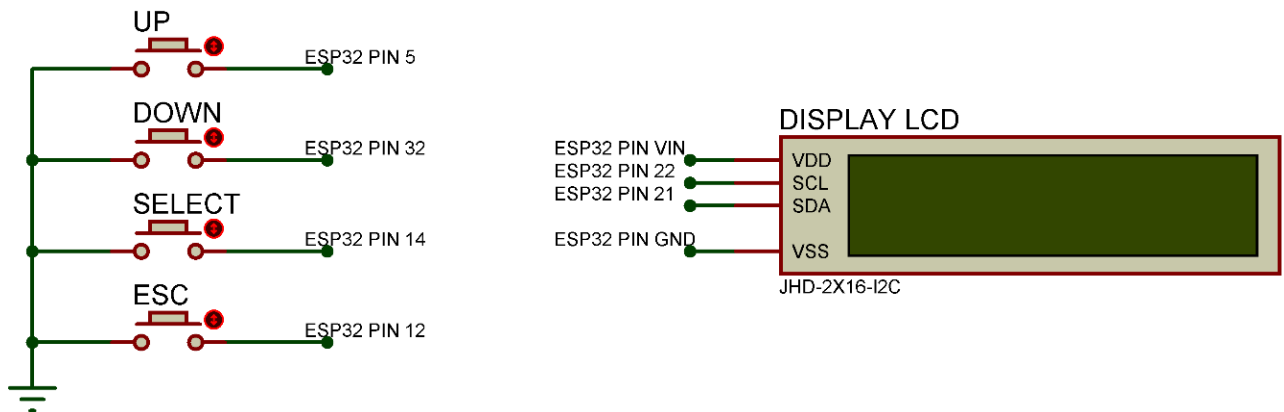
- 4 *push buttons*;
- 1 *display* LCD com módulo I2C acoplado;
- 1 ESP32 (o mesmo utilizado nas outras etapas da unidade de efeito).

A transferência de dados entre o ESP32 e o *display* LCD se dá por meio do protocolo de comunicação serial Circuito Inter Integrado (I2C - *Inter-Integrated Circuit*). Para isso são utilizadas duas linhas de sinais, sendo elas: SDA (linha de dados) e SCL (linha de *clock*). No entanto, conforme circuito implementado, apresentado na figura 18, não é necessário a inclusão de um circuito de *clock* externo para o sistema.

A interface desenvolvida possibilita que o usuário navegue em diferentes telas e visualize se o efeito está presente no sinal atual, parametrize o efeito, exclua e adicione efeito ao sinal. Para a navegação entres as informações e parâmetros contidos nos menus são utilizados os quatros botões apresentados a seguir.

- UP: navega para a próxima opção ou acresce o parâmetro selecionado;
- DOWN: navega para a opção anterior ou decresce o parâmetro selecionado;
- SELECT: seleciona a opção e direciona para a próxima tela;
- ESC: volta para a tela anterior.

Figura 18: Esquemático circuito interface *display* LCD.



Fonte: Elaborada pelo autor.

3.2.5 Configuração protocolo I2S

As características de como os dados são recebidos e enviados através do protocolo I2S pelo ESP32 também são configuradas via programa. Os parâmetros da comunicação I2S do microcontrolador foram definidos em:

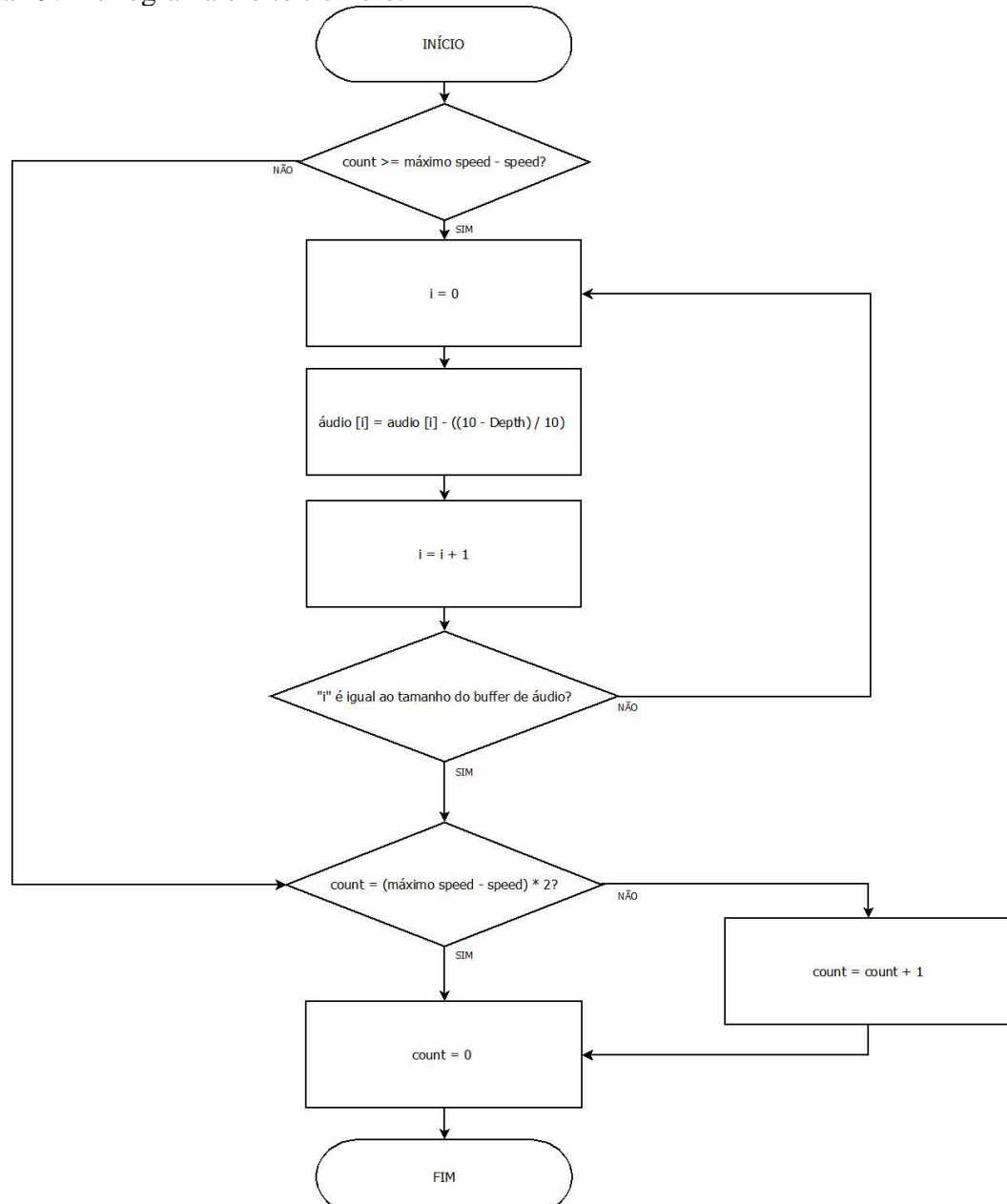
- Número de *buffers* DMA: 2;
- Tamanho *buffer* DMA = 1024 bytes;
- Frequência de amostragem = 46,875 kHz;
- Envio de dados: ambos os canais (esquerdo e direito);
- Número de bits por amostras: 24;
- Modo de operação do dispositivo: escravo, transmissor e receptor.

3.2.6 Programação efeito tremolo

O efeito de áudio implementado nesse projeto foi o efeito de tremolo que corresponde a modulação de volume. Por ser um efeito simples e não realizar modulação na frequência o processamento digital do mesmo foi realizado no domínio do tempo.

O efeito de tremolo é parametrizado pela definição de profundidade (*depth*) e velocidade de sua modulação (*speed*) da amplitude do sinal. No algoritmo implementado, a velocidade é definida em número de vetores de áudio e a profundidade em valores de 0 à 100%. A função tremolo é sempre executada após a leitura de um *buffer* de dados DMA. O algoritmo desenvolvido é representado pelo fluxograma na figura 19.

Figura 19: Fluxograma efeito tremolo.



Fonte: Elaborada pelo autor.

4 RESULTADOS E DISCUSSÕES

Este capítulo será dividido em duas partes. Na primeira serão apresentados os resultados obtidos a partir da análise dos circuitos eletrônicos que compõem o sistema da unidade digital de efeito. E na última parte serão apresentados os testes finais, tal como: avaliação do sinal de áudio amplificado por um amplificador de guitarra.

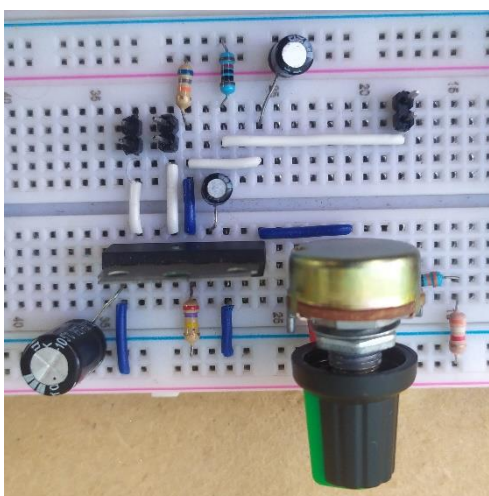
4.1 RESULTADOS E ANÁLISES DOS CIRCUITOS ELETRÔNICOS

Seguindo a sequência metodológica do trabalho e objetivando melhor compreensão dos resultados visualizou-se a avaliação dos circuitos eletrônicos que compõem o sistema da unidade digital de efeito de forma separada.

4.1.1 Pré-amplificador

Os testes realizados com o circuito pré-amplificador envolveram a inserção de um sinal senoidal na entrada do sistema e validação da amplificação realizada. Dessa forma, para a realização desses testes foi utilizado o gerador de funções e o osciloscópio. O circuito amplificador diferencial de áudio desenvolvido é apresentado na figura 20.

Figura 20: Circuito amplificador diferencial.

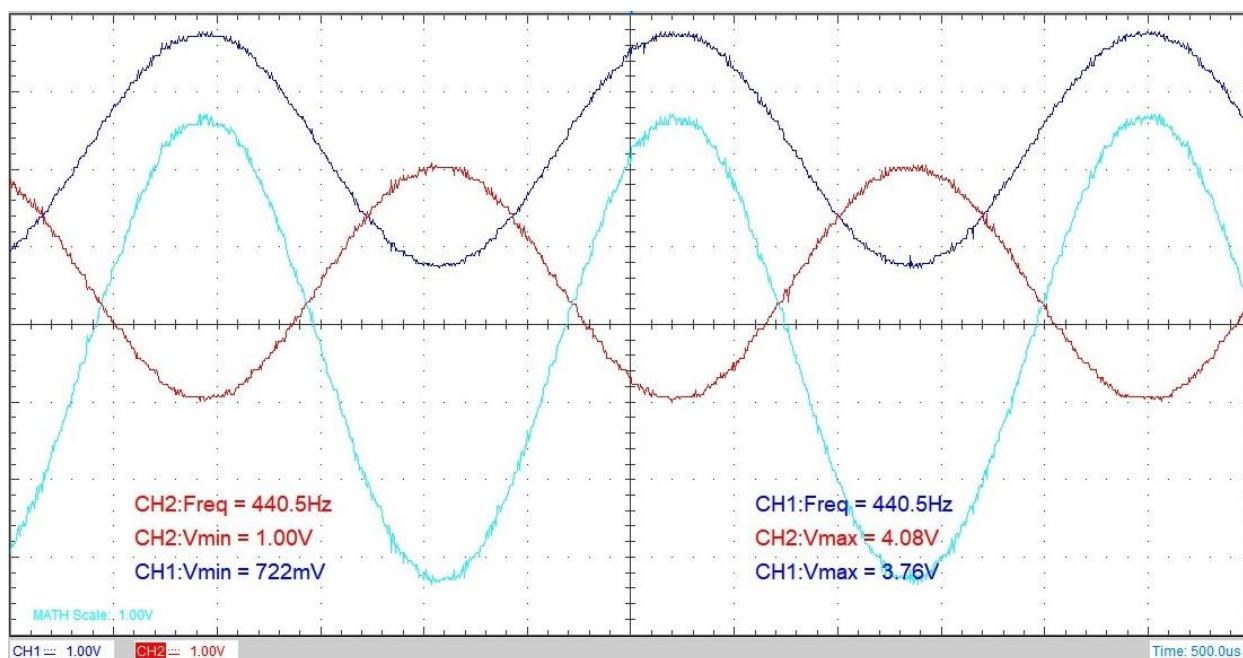


Fonte: Elaborada pelo autor.

O circuito integrado AN7523N é um amplificador diferencial de sinais de áudio. Portanto, o valor de tensão real do sinal amplificado corresponde-se da diferença de tensão entre os dois pinos de saída do amplificador (out1 - out2).

O primeiro teste realizado foi a verificação da maior tensão que pode ser obtida na saída do AN7523N nas configurações aplicadas nesse trabalho. Para isso, foi ajustado o trimpot (RV1), responsável por alterar o ganho do sistema, implementado no circuito até que o sinal de saída obtivesse a maior amplitude sem *clipping* na forma de onda que indicasse o limite de amplificação. A figura 21 apresenta o resultado obtido.

Figura 21: Saída diferencial AN7523N.



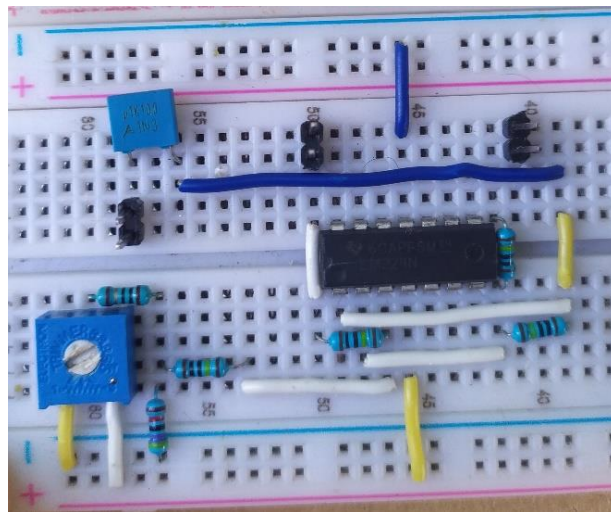
Fonte: Elaborada pelo autor.

A forma de onda de entrada do circuito AN7523N (in) para o teste acima foi uma senoide de aproximadamente 1,5 Vpp. Em contrapartida, o sinal de saída diferencial é dado por duas senoides: out1 (canal 1) e out2 (canal 2), defasadas de 180° uma da outra. Portanto, a amplitude real do sinal de saída é dada pela onda de maior amplitude, apresentada na figura acima, que corresponde-se da diferença: out1 (canal 1) – out2 (canal 2).

O resultado obtido demonstra que o circuito integrado AN7523N foi capaz de amplificar o sinal de entrada até que ele obtivesse aproximadamente 6 V de amplitude. Com isso, foi demonstrado que o amplificador diferencial de áudio utilizado é capaz de amplificar os sinais de áudio sem distorcê-los.

O segundo teste realizado com circuito pré-amplificador é a capacidade do circuito subtrator, apresentado na figura 22, referenciar o sinal diferencial de saída do AN7523N (out1 – out2) ao nó terra do sistema. Para isso, foi gerado um sinal senoidal de aproximadamente 440 mVpp como sinal de entrada do amplificador diferencial (in) e realizada a leitura em corrente contínua do sinal de saída do circuito subtrator, anterior ao capacitor de acoplamento C4, utilizando o osciloscópio.

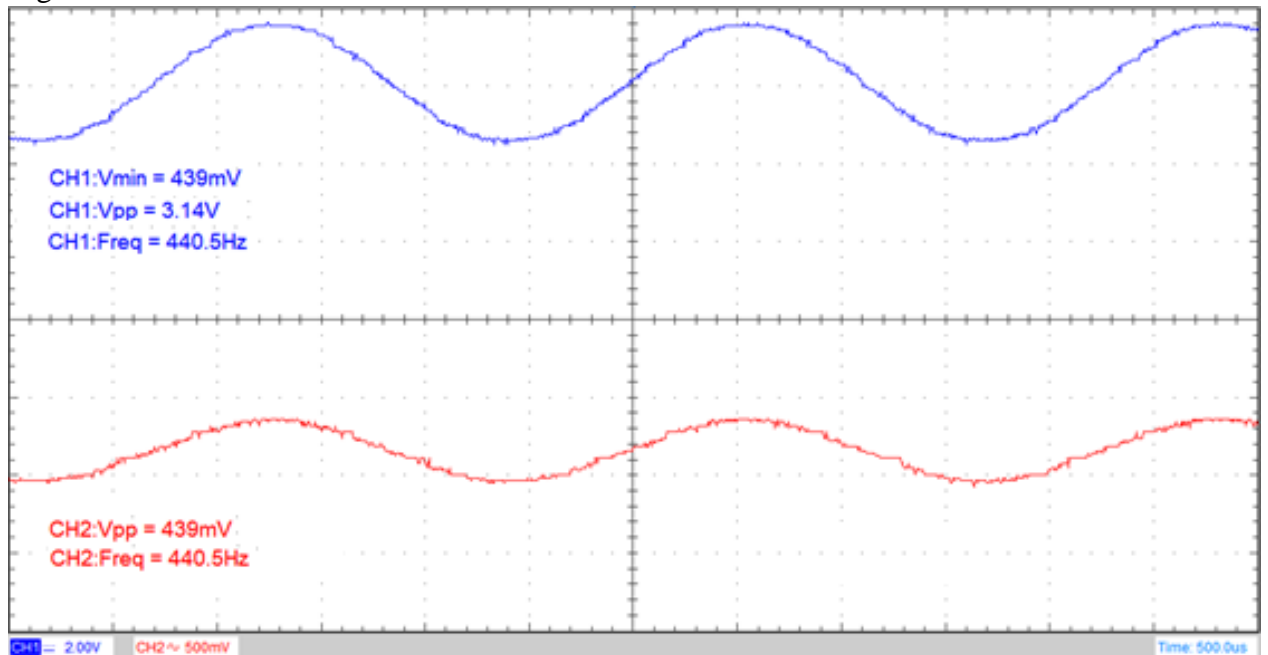
Figura 22: Circuito Subtrator.



Fonte: Elaborada pelo autor.

A figura 23 evidencia a maior amplitude obtida sem que ocorresse *clipping* na forma de onda de saída. O canal 2 apresenta o sinal de entrada do circuito pré-amplificador (in) e o canal 1 o sinal de saída analisado. Dessa forma, é válido afirmar que o valor do *offset* de aproximadamente de 1,8 V e ganho de aproximadamente de 7 vezes advindo do amplificador de áudio AN7523N obtido nessa configuração deverão ser os máximos utilizados para não comprometer a qualidade do sinal de áudio. O valor de *offset* pode ser ajustado pelo trimpot (RV2) incluso no sistema do circuito subtrator.

Figura 23: Offset do circuito subtrator.

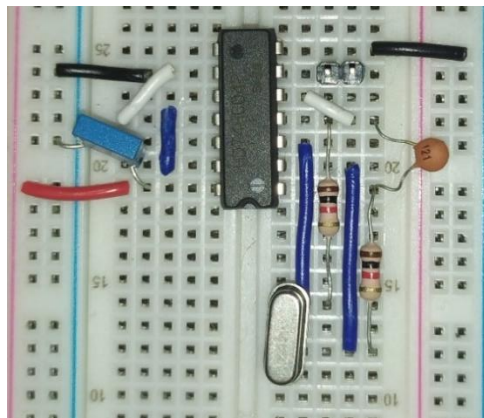


Fonte: Elaborada pelo autor.

4.1.2 Circuito de clock

Os testes do circuito de *clock* consistiram da validação das frequências dos sinais de *clock* que compõem o sistema da unidade de efeito. Para esse fim, foi utilizado o analisador lógico e feitas as análises das frequências e ciclos de trabalho dos três sinais de *clock*: sinal *clock* do sistema (SCK) utilizado somente pelo CJMCU-1808 e os sinais BKC e LRC gerados pelo CJMCU-1808 e utilizados pelo CJMCU-1334 e ESP32. O circuito gerador do *clock* do sistema (SCK) implementado é apresentado na figura 24.

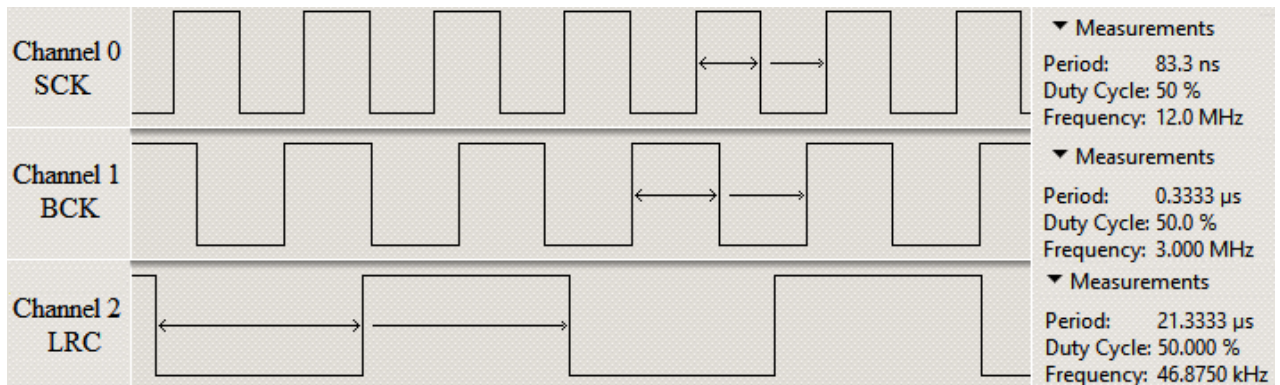
Figura 24: Circuito de Clock SCK.



Fonte: Elaborada pelo autor.

Primeiramente, foram realizadas as medições utilizando o analisador lógico e após verificado que todos os sinais de *clock* operam nas frequências projetadas (conforme apresentação na figura 25). O sinal SCK oscila em uma frequência de 12 MHz, enquanto o sinal LRC oscila em 46,875 kHz. Por outro lado, o sinal BCK possui frequência de 3 MHz (produto de 46,875 kHz por 64 bits), pois apesar da resolução dos conversores ser de 24 bits, o comprimento da palavra do ADC PCM-1808 é de 64 bits.

Figura 25: Sinais de clock.

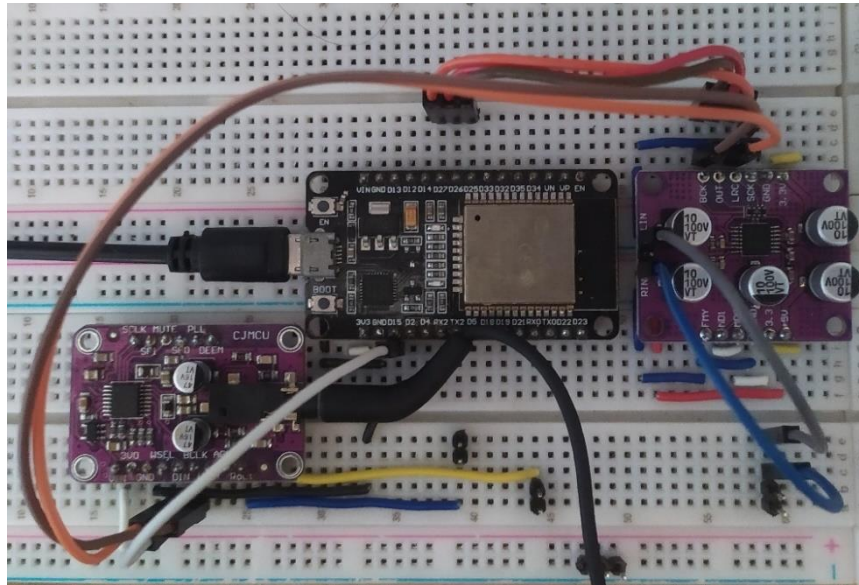


Fonte: Elaborado pelo autor.

4.1.3 Circuito de conversores de sinais

Os testes dos conversores de sinais consistiram na comparação entre os sinais de entrada do ADC e os sinais de saída do DAC. Para isso, foi gerado um sinal senoidal de aproximadamente 1,5 Vpp como sinal de entrada do módulo CJMCU-1808 (out3) através do gerador de funções, o qual foi comparado com o sinal gerado na saída do CJMCU-1334 (out4) a partir do uso do osciloscópio. A figura 26 apresenta o circuito conversores de sinais implementado.

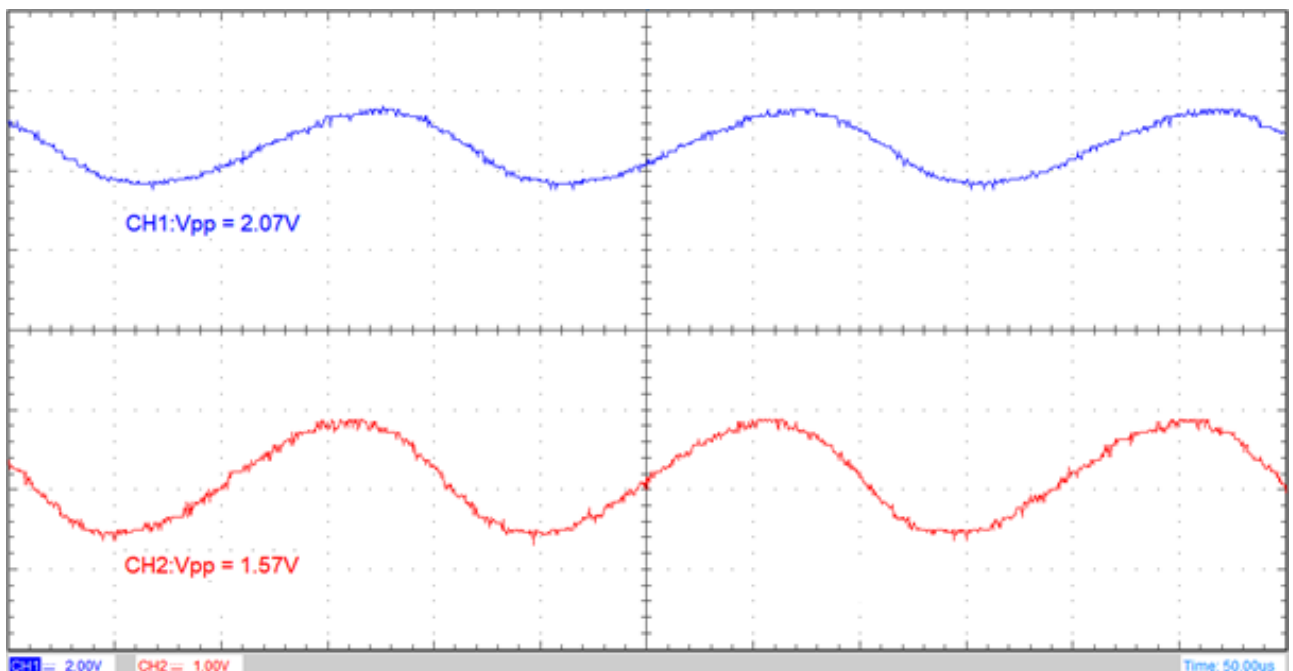
Figura 26: Circuito conversores de sinais.



Fonte: Elaborada pelo autor.

A figura 27 apresenta a relação entre o sinal de entrada do CJMCU-1808 (canal 2) e o sinal de saída do CJMCU-1334 (canal 1). O sinal de saída possui um pequeno ganho de tensão, aproximadamente 1,3 vezes, devido ao CJMCU-1334 conter um pequeno amplificador, porém são mantidas as características do sinal, tais como: formato e frequência da forma de onda.

Figura 27: Forma de onda conversores de sinais.

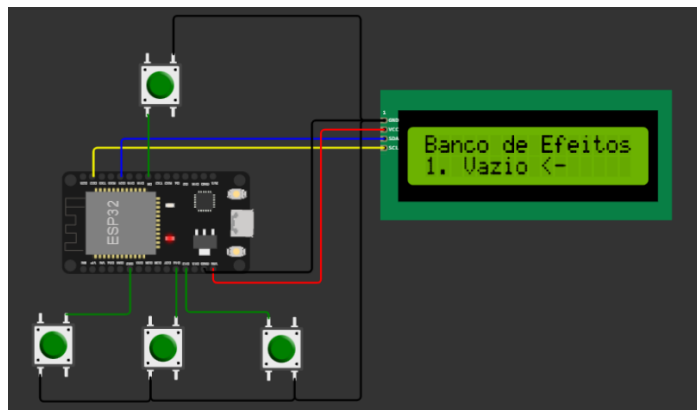


Fonte: Elaborada pelo autor.

4.1.4 Interface eletrônica

Os testes da interface eletrônica constituíram-se primeiramente da simulação online do circuito e da programação do menu em lista por meio da plataforma Wokwi. Para isso, primeiramente, foi necessário a implementação do circuito e do algoritmo na plataforma de simulação definida, e, após testes de funcionamento do sistema. A figura a 28 apresenta o circuito criado para a realização da simulação.

Figura 28: Circuito interface através de simulador.



Fonte: Elaborada pelo autor.

Após a simulação foi realizada a montagem do circuito eletrônico na protoboard, conforme figura 19, e verificado o funcionamento correto do circuito e programação.

Figura 29: Circuito interface elaborado na protoboard.

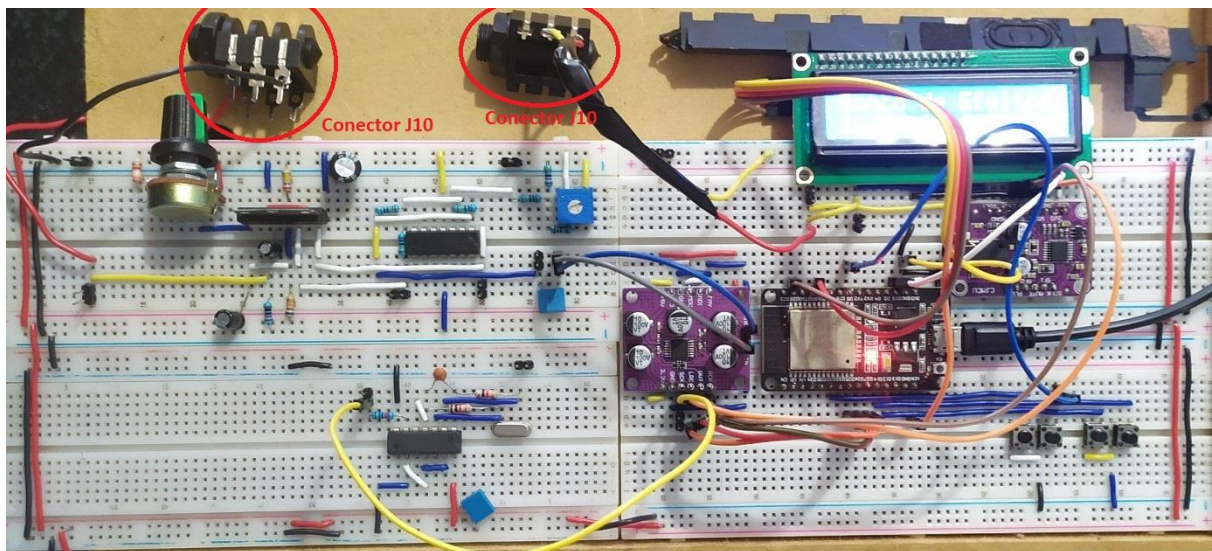


Fonte: Elaborada pelo autor.

4.2 TESTES FINAIS: EFEITO TREMOLO

Para a validação do efeito tremolo foi utilizado um amplificador de guitarra e uma guitarra elétrica. Inicialmente, foi conectado o sinal da guitarra à entrada da unidade de efeito (in) e o amplificador de guitarra ao sinal de saída da unidade (out4). Posteriormente, foram realizadas gravações de áudio com e sem a presença do efeito de tremolo. Após isso, foram feitas as considerações. Para isso foi utilizado dois conectores J10 (conforme a ilustração na figura 30).

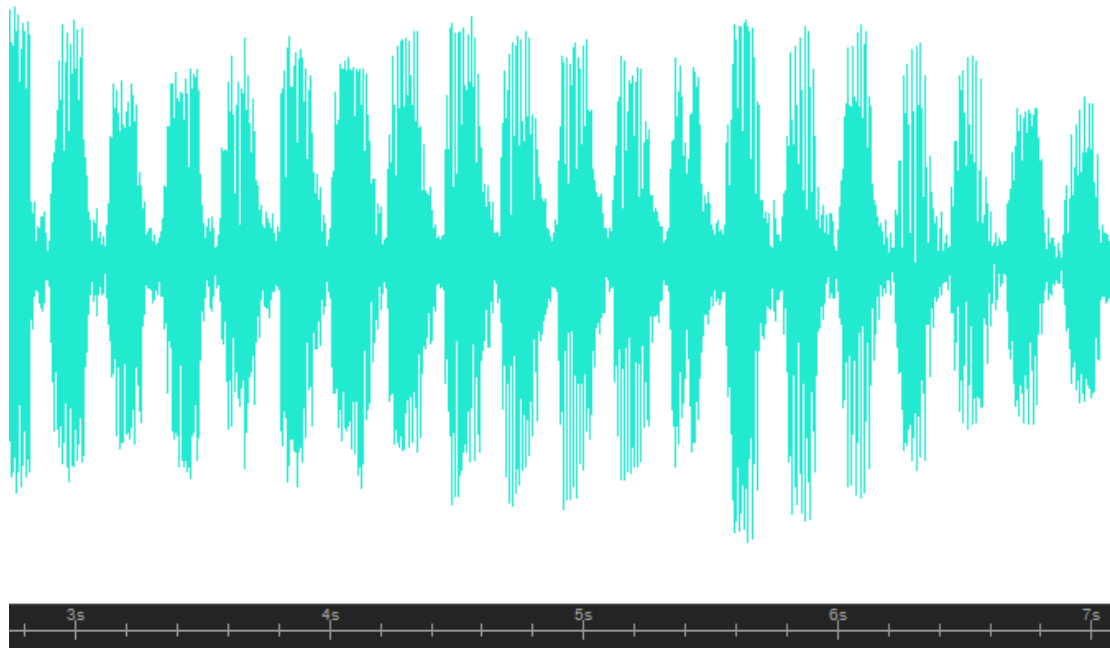
Figura 30: Circuito unidade de efeito.



Fonte: Elaborada pelo autor.

A figura 31 apresenta o formato da onda obtida com a presença do efeito tremolo. Tal forma de onda foi amostrada através da análise, realizada no software *WavePad Sound Editor*, do som gravado. O sinal de entrada da unidade de efeito foi o sinal de saída da guitarra elétrica, correspondente de um sinal de áudio contínuo. Dessa forma, é possível visualizar que o efeito de áudio tremolo resultou corretamente na modulação da amplitude da forma de onda do sinal de entrada do sistema (in). É possível visualizar a modulação do volume do som gravado em intervalos fixos de tempo. Portanto, pode ser concluído que é possível realizar processamento digital de sinais em alta frequência utilizando o ESP32.

Figura 31: Sinal de áudio com efeito tremolo.



Fonte: Elaborada pelo autor.

5 CONSIDERAÇÕES FINAIS

A pergunta de pesquisa que motivou esse estudo foi: de que maneira é possível desenvolver uma unidade digital de efeito tremolo para guitarra elétrica que seja reprogramável? Em busca da resposta, foi elaborado um sistema utilizando uma plataforma microcontroladora, reprogramável, de uma unidade de efeito que com os testes mostrou-se eficiente e possibilitou uma alta qualidade sonora.

Inicialmente buscou-se entender o processamento digital de sinais analógicos contínuos no tempo e os diferentes efeitos de áudio existentes, a fim de idealizar os diferentes estágios para a construção da unidade. Formulou-se assim os objetivos dessa pesquisa: o objetivo geral foi construir uma unidade digital, reprogramável, de efeito tremolo para guitarra elétrica utilizando sistema microcontrolado. O intento desse objetivo era o de utilizar um microcontrolador como processador de sinais discretizados, para que o mesmo pudesse ser reprogramável, possibilitando a utilização da unidade criada para produção do efeito em tempo real, alteração do efeito programado e programação futura de novos efeitos de áudio. Sendo assim foi possível alcançar o objetivo central do trabalho.

A plataforma microcontroladora selecionada foi a ESP-32 DevKit v1. Tal plataforma se mostrou uma boa escolha para o desenvolvimento desse projeto, devido presença de diversos protocolos de comunicação e bastante documentação disponível.

Para realizar a análise da unidade de efeito construída, foram realizadas leituras das formas de onda nos diferentes estágios do sistema, análise lógica dos sinais de *clock* do protocolo I2S, navegação no menu da interface construída e validação do efeito tremolo através de gravação do sinal de áudio produzido. Dessa forma, primeiramente foi averiguado a amplificação do sinal pelo circuito amplificador, analisando a máxima amplitude obtida sem que ocorresse *clipping* na forma de onda de saída. Em seguida, foram verificadas as frequências e os ciclos de trabalho das linhas de *clock* do protocolo I2S. Após, foi amostrado o sinal de saída do sinal do DAC e comparado com o sinal de entrada do ADC. Posteriormente, foi realizada análise do som de saída da unidade com a presença do efeito tremolo.

Depois de alguns testes o funcionamento do protótipo ocorreu como esperado. O sistema permitiu parametrizar, adicionar ou excluir o efeito através da interface criada com display LCD. O efeito tremolo modulou corretamente a amplitude do sinal, resultando em uma modulação de volume do som.

O presente trabalho se inseriu no campo de conhecimento da Engenharia de Automação Industrial, especificamente em processamento digital de sinais. Além disso, apresentou conceitos de elétrica, eletrônica, música e sonorização.

Como sugestão de trabalho futuro, apresenta-se a ideia da realização de simulação de sistemas lineares invariantes no tempo, como, por exemplo, de ambientes, equipamento de áudio, através da unidade implementada, uso da FFT e da convolução de sinais no domínio da frequência. Ademais, tem-se como sugestão a implementação de novos efeitos.

Por fim, a resposta da pergunta de pesquisa pode ser assim resumida: é possível construir uma unidade digital reprogramável de efeito para guitarra elétrica de alta qualidade sonora utilizando uma plataforma microcontroladora.

REFERÊNCIAS

AGNOL, C. D. **Comparação entre microcontroladores e aplicação do FPGA no controle do conversor Boost**. Orientadora: Franciéli Lima de Sá. 2018. 112 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) - Centro Universitário UNIFACVEST, Lages, 2018.

ALVES, G. C. **Projeto de unidade de efeito sonoro para guitarras**. Orientador: Marcelo de Oliveira Rosa. 2018. 67 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) - Universidade Tecnológica Federal do Paraná, Curitiba, 2018.

BORGES, A. N.; RODRIGUES, C. G. **Introdução à Física Acústica**. São Paulo: Livraria da Física, 2017.

BOSS. 2021. Disponível em: <https://www.boss.info/br/categories/discontinued/>. Acesso em: 10 fev. 2021.

DAVIS, G.; JONES, R. **Sound Reinforcement Handbook**. 2. ed. Milwaukee: Yamaha, 1990.

DINIZ, P. S. R.; SILVA, E. A. B.; NETTO, S. L. **Processamento Digital de Sinais: projeto e análise de sistemas**. 2. ed. p. 1-55, 123-129-154-242. Porto Alegre: Bookman, 2014.

ESPRESSIF SYSTEMS. **Inter-IC Sound (I2S)**. 2022. Disponível em: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2s.html>. Acesso em: 04 maio 2022.

FERNANDO K TECNOLOGIA. **Pequeno, mas Poderoso: STM32**. 2018. Disponível em: <https://www.fernandok.com/2018/08/pequeno-mas-poderoso-stm32.html>. Acesso em: 23 fev. 2021.

LATHI, B.P. **Sinais e Sistemas Lineares**. Tradução: Guilherme Guimarães Parma. 2. ed. Porto Alegre: Bookman, 2007. 856 p.

MATSUSHITA ELECTRONICS CORPORATION. **Product Specifications: AN7523N**. 2000. Disponível em: <https://storage.googleapis.com/baudaeletronicadatasheet/AN7523N.pdf>. Acesso em: 07 jul. 2021.

NXP SEMICONDUCTORS. Datasheet: **UDA1334ATS**. 2000. Disponível em: https://storage.googleapis.com/baudaeletronicadatasheet/Datasheet_UDA1334ATS.pdf. Acesso em: 05 jul. 2021.

OPPENHEIM, A. V.; SCHAFER, R. W. **Processamento em tempo discreto de sinais**. Tradução: Daniel Vieira. 3. ed. São Paulo: Pearson, 2013.

QUEIRÓZ FILHO, E. S. **Implementação e análise de um pedal de efeito para utilização em guitarras elétricas**. Orientador: Sérgio Daher. 2017. 79 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) - Universidade Federal do Ceará, Fortaleza, 2017.

Rei de Les Paul. 2015. Disponível em: <http://www.reidalespaul.com.br/2015/09/historia-dos-captadores-capitulo-1-em.html>. Acesso em: 15 fev. 2021

ROCHA, M. E. L. **A Tecnologia como meio expressivo do guitarrista atuante no mercado musical pop**. Orientador: José Eduardo Ribeiro de Paiva. 2011. 159 f. Tese (Doutorado em Música) - Universidade Estadual de Campinas, Campinas, 2011.

SOUZA, I. M. R. **Proposta de ensino investigativo usando a indução eletromagnética e piezoeletricidade aplicados aos sensores de guitarras e violões**. Orientador: Antônio Maia de Jesus Chaves Neto. 2018. 111 f. Dissertação (Mestrado Profissional em Ensino de Física) - Universidade Federal do Pará, Belém, 2018.

TEXAS INSTRUMENTS. Datasheet: **LMx24-N, LM2902-N Low-Power, Quad-Operational Amplifiers**. 2015. Disponível em: <https://www.ti.com/lit/ds/symlink/lm124-n.pdf>. Acesso em: 03 abr. 2021.

TEXAS INSTRUMENTS. Datasheet: **PCM1808 Single-Ended, Analog-Input 24-Bit, 96-kHz Stereo ADC**. 2015. Disponível em: <https://datasheetspdf.com/pdf-file/560335/Burr-Brown/PCM1808/1>. Acesso em: 03 jun. 2021.

TOCCI, R. J.; WIDMER, N. S.; MOSS, G. L. **Sistemas digitais: princípios e aplicações**. 10. ed. São Paulo: Pearson Education do Brasil, 2007.

XProjetos.net. Arduino, Esp32 e ESP8266 - Comparação. 2019. Disponível em: <https://xprojetos.net/arduino-esp32-e-esp8266-comparacao/>. Acesso em: 03 jun. 2021.

APÊNDICE A – PROGRAMAÇÃO DO SISTEMA ESP32

```

// Valores constantes - pinos display LCD
#define upButton 5           // botão up interface
#define downButton 32       // botão down interface
#define selectButton 14     // botão select interface
#define escButton 12        // botão esc interface
#define speedM 20           // Máximo speed tremolo

// Variáveis globais
byte buffer[3072];          // buffer I2S -> 1024 amostras (24
bits de resolução)
double audioRe[1024];      // vetor de audio
int i, j;                   // variável auxiliar laços FOR
byte screen = 1;           // variável auxiliar para controlar a
mudança de tela
char *banco[1] = {"Vazio"}; // variável auxiliar para escrita dos
nomes dos efeitos presentes no banco
int tremoloSpeed = 1;      // controla a velocidade do tremolo
int tremoloDepth = 5;      // controla a profundidade do tremolo
byte countTremolo = 0;     // variável auxiliar passo tremolo

// Inclusão de bibliotecas
#include <Wire.h>            // comunicação I2C display LCD
#include <LiquidCrystal_I2C.h> // biblioteca display LCD
#include "driver/i2s.h"      // comunicação I2S - ADC e DAC

// Objetos
LiquidCrystal_I2C lcd(0x27, 16, 2); // display LCD
static const i2s_port_t i2s_num = I2S_NUM_0; // protocolo I2S

// Configuração estrutura I2S
static const i2s_config_t i2s_config = {
    .mode = (i2s_mode_t)(I2S_MODE_SLAVE | I2S_MODE_TX | I2S_MODE_RX),
    // modo escravo-transmissor-receptor
    .sample_rate = 46875,
    // frequência de amostragem
    .bits_per_sample = I2S_BITS_PER_SAMPLE_24BIT,
    // resolução da amostra
    .channel_format = I2S_CHANNEL_FMT_RIGHT_LEFT,
    // canais direito e esquerdo
    .communication_format = (i2s_comm_format_t)(I2S_COMM_FORMAT_I2S |
I2S_COMM_FORMAT_I2S_MSB),
    // formato de dados I2S
    .intr_alloc_flags = ESP_INTR_FLAG_LEVEL1,
    // prioridade interrupção alta
    .dma_buf_count = 2,
    // números de buffers de dados
    .dma_buf_len = 1024,
    // tamanho dos buffers - 2 buffers de 1024 bytes
    .use_apll = 0,
    .tx_desc_auto_clear = true,
    .fixed_mclk = -1
};

// Configuração portas I2S
static const i2s_pin_config_t pin_config = {
    .bck_io_num = 25, // porta bck - clock frequência de amostragem
    .ws_io_num = 27, // porta ws - seleciona canal direito ou esquerdo

```

```

    .data_out_num = 15, // porta de saída de dados serial I2S
    .data_in_num = 26 // porta de entrada de dados serial I2S
};

void setup() {
    Serial.begin(115200);

    // Inicializa display LCD
    lcd.init();
    lcd.backlight();

    // PinMode display LCD
    pinMode(upButton, INPUT_PULLUP);
    pinMode(downButton, INPUT_PULLUP);
    pinMode(selectButton, INPUT_PULLUP);
    pinMode(escButton, INPUT_PULLUP);

    // Tela inicial display LCD
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(F("Banco de Efeitos"));
    lcd.setCursor(0, 1);
    lcd.print(F("1. Vazio <-"));

    // Instalação driver I2S
    esp_err_t err;
    err = i2s_driver_install(i2s_num, &i2s_config, 0, NULL);
    if (err != ESP_OK) {
        Serial.printf("Failed installing driver: %d\n", err);
        while (true);
    }
    err = i2s_set_pin(i2s_num, &pin_config);
    if (err != ESP_OK) {
        Serial.printf("Failed setting pin: %d\n", err);
        while (true);
    }
    Serial.println("I2S driver installed.");
}

void loop() {
    // Atualiza menu LCD
    if (!digitalRead(upButton) || !digitalRead(downButton) ||
        !digitalRead(selectButton) || !digitalRead(escButton)) {
        Menu();
    }

    size_t BytesWritten;

    i2s_read(i2s_num, &buffer, sizeof(buffer), &BytesWritten, portMAX_DELAY);
    // leitura de dados serial I2S

    readDados(); // acoplamento 24 bits

    if (banco[0] == "Tremolo") {
        tremolo(); // efeito tremolo
    }

    writeDados(); // desacoplamento 8 bits
}

```

```

    i2s_write(i2s_num, &buffer, sizeof(buffer), &BytesWritten,
portMAX_DELAY);
// escrita de dados serial I2S
}

// Acoplamento de dados -> 8 para 24 bits
void readDados() {
    for (i = 0; i < 1024; i++) {
        audioRe[i] = (buffer[i * 3] << 16) | (buffer[i * 3 + 1] << 8) |
(buffer[i * 3 + 2]);
    }
}

// Desacoplamento de dados -> 24 para 8 bits
void writeDados() {
    for (i = 0; i < 1024; i++) {
        buffer[i * 3 + 2] = int(audioRe[i]) & 0xFF;
        buffer[i * 3 + 1] = (int(audioRe[i]) & 0xFFFF) >> 8;
        buffer[i * 3 + 0] = (int(audioRe[i]) & 0xFFFFFFFF) >> 16;
    }
}

// Controla Menu LCD
void Menu() {
    // Ação para o botão upButton
    if (!digitalRead(upButton)) {
        // Menu 1
        if (screen == 1)
            screen = 2;
        else if (screen == 2)
            screen = 1;
        // Menu 2
        else if (screen == 11)
            screen = 12;
        else if (screen == 12)
            screen = 11;
        // Menu Tremolo
        else if (screen == 21)
            screen = 22;
        else if (screen == 22)
            screen = 23;
        else if (screen == 23)
            screen = 24;
        else if (screen == 24)
            screen = 21;
        // Parâmetros Tremolo
        else if (screen == 32) {
            if (tremoloSpeed != 10)
                tremoloSpeed = tremoloSpeed + 1;
        }
        else if (screen == 33) {
            if (tremoloDepth != 10)
                tremoloDepth = tremoloDepth + 1;
        }
    }
    while (!digitalRead(upButton));
}

// Ação para o botão downButton
if (!digitalRead(downButton)) {
    // Menu 1
    if (screen == 1)

```

```

    screen = 2;
else if (screen == 2)
    screen = 1;
// Menu 2
else if (screen == 11)
    screen = 12;
else if (screen == 12)
    screen = 11;
// Menu Tremolo
else if (screen == 21)
    screen = 24;
else if (screen == 22)
    screen = 21;
else if (screen == 23)
    screen = 22;
else if (screen == 24)
    screen = 23;
// Parâmetros Tremolo
else if (screen == 32) {
    if (tremoloSpeed != 0)
        tremoloSpeed = tremoloSpeed - 1;
}
else if (screen == 33) {
    if (tremoloDepth != 0)
        tremoloDepth = tremoloDepth - 1;
}

while (!digitalRead(downButton));
}

// Ação para o Botão selectButton
if (!digitalRead(selectButton)) {
    switch (screen) {
        case 24:
            banco[0] = "Tremolo";
            screen = 7; // valor menor que 10 para voltar para o Menu 2
            break;
        case 12:
            banco[0] = "Vazio"; // Remove efeito tremolo
            break;
    }
    if (screen < 10) // seleciona posição do banco de efeitos
        screen = 11;
    else if (screen == 11) // seleciona tremolo
        screen = 21;
    else if (screen == 12) // volta para o Menu 1
        screen = 1;
    else if (screen == 21) // seleciona Wave tremolo
        screen = 31;
    else if (screen == 22) // seleciona Speed tremolo
        screen = 32;
    else if (screen == 23) // seleciona Depth tremolo
        screen = 33;

    while (!digitalRead(selectButton));
}

// Ação para o Botão escButton
if (!digitalRead(escButton)) {
    if (screen > 10 && screen < 20)
// volta para o Menu 1 (banco de efeitos)

```

```

    screen = 1;
    if (screen > 20 && screen < 30)
    // volta para o Menu 2 (escolha do efeito)
        screen = 11;
    if (screen > 30)
    // volta para Menu Parâmetros Tremolo
        screen = 21;

    while (!digitalRead(escButton));
}

// Escreve na tela selecionada
switch (screen) {
    // Menu 1
    case 1:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("Banco de Efeitos"));
        lcd.setCursor(0, 1);
        lcd.print(F("1. "));
        lcd.setCursor(3, 1);
        lcd.print(F(banco[0]));
        break;
    // Menu 2
    case 11:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("A. Tremolo <-"));
        lcd.setCursor(0, 1);
        lcd.print(F("B. Remover"));
        break;
    case 12:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("B. Remover <-"));
        lcd.setCursor(0, 1);
        lcd.print(F("A. Tremolo"));
        break;

    // Menu Tremolo
    case 21:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("Tremolo"));
        lcd.setCursor(0, 1);
        lcd.print(F("a. Wave <-"));
        break;
    case 22:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("Tremolo"));
        lcd.setCursor(0, 1);
        lcd.print(F("b. Speed <-"));
        break;
    case 23:
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print(F("Tremolo"));
        lcd.setCursor(0, 1);
        lcd.print(F("c. Depth <-"));
        break;
}

```



```

case 24:
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(F("Tremolo"));
    lcd.setCursor(0, 1);
    lcd.print(F("d. Adicionar <-"));
    break;

// Parâmetros Tremolo
case 31:
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(F("Wave"));
    lcd.setCursor(0, 1);
    lcd.print(F("Quadrada"));
    break;
case 32:
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(F("Speed"));
    lcd.setCursor(0, 1);
    lcd.print((tremoloSpeed));
    break;
case 33:
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print(F("Depth"));
    lcd.setCursor(0, 1);
    lcd.print((tremoloDepth));
    break;
}
}

// Efeito Tremolo
void tremolo() {
    if (countTremolo >= (speedM - tremoloSpeed)) {
        for (i = 0; i < 1024; i++)
            audioRe[i] = audioRe[i] * ((10 - tremoloDepth) / 10);
    }
    if (countTremolo == (speedM - tremoloSpeed) * 2) {
        countTremolo = 0;
    }
    else {
        countTremolo++;
    }
}
}

```



Emitido em 09/09/2022

CÓPIA DO TRABALHO Nº 171/2022 - DELMAX (11.57.05)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 12/09/2022 10:16)

FREDERICO DUARTE FAGUNDES
PROFESSOR ENS BASICO TECN TECNOLOGICO
DELMAX (11.57.05)
Matrícula: 1107165

Para verificar a autenticidade deste documento entre em <https://sig.cefetmg.br/documentos/> informando seu número: **171**, ano: **2022**, tipo: **CÓPIA DO TRABALHO**, data de emissão: **12/09/2022** e o código de verificação: **fa37e7cbdc**