Humberto Lopes Ferreira de Souza

SoniCUBE: Unindo música e jogos através de um sistema embarcado

Universidade Federal de Pernambuco
secgrad@cin.ufpe.br
www.cin.ufpe.br/~graduacao

Recife
2024

Humberto Lopes Ferreira de Souza

SoniCUBE: Unindo música e jogos através de um sistema embarcado

Trabalho apresentado ao Programa de Graduação em Engenharia da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Engenharia da Computação.

Área de Concentração: *Sistemas Embarcados*
Orientador: *Adriano Augusto de Moraes Sarmento*

Recife
2024

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Souza, Humberto Lopes Ferreira de.
SoniCUBE: Unindo música e jogos através de um sistema embarcado / Humberto Lopes Ferreira de Souza. - Recife, 2024.
36 p. : il., tab.

Orientador(a): Adriano Augusto de Moraes Sarmento
Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Pernambuco, Centro de Informática, Engenharia da Computação - Bacharelado, 2024.
Inclui referências.

1. Sistemas Embarcados. 2. Gamificação. 3. Música. I. Sarmento, Adriano Augusto de Moraes. (Orientação). II. Título.

000 CDD (22.ed.)

Humberto Lopes Ferreira de Souza

SoniCUBE: Unindo música e jogos através de um sistema embarcado

Trabalho de Conclusão de Curso apresentado como pré-requisito para conclusão do Curso de Bacharelado em Engenharia da Computação do Centro de Informática da Universidade Federal de Pernambuco. Defendida e aprovada em 6 de agosto de 2024 pela seguinte banca examinadora:

Adriano Augusto de Moraes Sarmento
Orientador/CIn-UFPE

Filipe Carlos De Albuquerque Calegario
Examinador/CIn-UFPE

AGRADECIMENTOS

Ao meu orientador, Prof. Adriano Augusto de Moraes Sarmento, pela orientação segura e crítica que me permitiu concluir este percurso.

À minha mãe, Maria da Luz, por seu apoio e companheirismo ao longo de toda minha jornada acadêmica.

Aos professores do CIn - UFPE que fizeram parte da minha trajetória, pelos valiosos ensinamentos e pelo compromisso com a educação.

A todos os meus amigos, com quem compartilho as alegrias e desafios da vida, que me acompanharam e apoiaram durante este percurso.

Aos meus colegas e amigos de curso que dividiram esta caminhada comigo.

RESUMO

A música e o aprendizado musical são conhecidos por desenvolver habilidades cognitivas, motoras, mentais e sociais do indivíduo. No entanto, muitos métodos tradicionais podem ser desafiadores e desmotivadores para iniciantes. Tornar o aprendizado musical mais divertido e envolvente é essencial para manter as pessoas motivadas e engajadas. Este trabalho explora a interseção entre música e tecnologia, oferecendo uma nova experiência de entretenimento e aprendizado musical, permitindo aos usuários utilizar notas musicais para interagir com jogos eletrônicos. Para isso, foi desenvolvido o SoniCUBE, um sistema embarcado capaz de capturar e interpretar notas musicais e mapeá-las corretamente para os botões de um joystick, utilizando uma plataforma ESP32 e um sensor de som MAX9814, sendo necessária apenas uma conexão Bluetooth. O sistema foi testado em diferentes cenários de uso, utilizando diversos instrumentos musicais e jogos eletrônicos. Para avaliar a experiência de jogo e a eficácia do mapeamento de notas e da conexão Bluetooth, foram observadas a responsividade do sistema e a precisão na interpretação das notas musicais. Os resultados indicaram que o SoniCUBE alcançou os objetivos propostos, oferecendo uma experiência de jogo e confirmando a eficácia das funcionalidades implementadas.

Palavras-chave: Sistemas Embarcados. Gamificação. Música.

LISTA DE FIGURAS

Figura 1 – Funcionamento do SoniCUBE 12

Figura 2 – Wii Fit Balance Board 20
Figura 3 – Arquitetura do projeto (Melo *et al.*, 2023) 21

Figura 4 – Sensor de som KY-038 23
Figura 5 – Sensor de som MAX9814 24
Figura 6 – Esquema de circuito do SoniCUBE 25
Figura 7 – Fluxo completo de funcionamento do SoniCUBE 25
Figura 8 – Fluxo da comunicação com sensor e amostragem de dados do SoniCUBE 26
Figura 9 – Fluxo de processamento dos dados e funcionamento do SoniCUBE . . . 26
Figura 10 – Circuito do SoniCUBE 27
Figura 11 – Forma de onda do sinal de áudio lido pelo sensor 28
Figura 12 – Saída do SoniCUBE com frequências 29
Figura 13 – Saída do SoniCUBE com notas mapeadas 29
Figura 14 – Saída do SoniCUBE com mapeamento do joystick 30

Figura 15 – Teste básico do SoniCUBE 31
Figura 16 – Testes no computador com diferentes instrumentos 33

LISTA DE TABELAS

Tabela 1 – Resultado da pesquisa às bases de dados 19

Tabela 2 – Mapeamento de notas para botões de joystick 30

LISTA DE ACRÔNIMOS

ADC	Conversor Analógico Digital
AGC	Controle Automático de Ganho
BDTD	Biblioteca Digital Brasileira de Teses e Dissertações
CI	Circuito Integrado
CPU	Unidade Central de Processamento
DFT	Transformada Discreta de Fourier
DMA	Acesso Direto à Memória
ESPM	Escola Superior de Propaganda e Marketing
FFT	Transformada Rápida de Fourier
GERSE	Guia de Elicitação de Requisitos para Sistemas Embarcados
HID	Dispositivo de Interface Humana
Hz	Hertz
IoT	Internet das Coisas
MIDI	Musical Instrument Digital Interface
NIME	New Interfaces for Musical Expression
PCB	Placa de Circuito Impresso
PGB	Pesquisa Game Brasil

SUMÁRIO

1 INTRODUÇÃO . . . 11

2 REVISÃO TEÓRICA . . . 13
2.1 PLATAFORMA DE DESENVOLVIMENTO . . . 13
2.1.1 Sistemas Embarcados . . . 13
2.1.2 Plataforma ESP32 . . . 14
2.1.3 Dispositivo de Interface Humana (HID) . . . 14
2.2 COMUNICAÇÃO COM O SENSOR . . . 15
2.2.1 Amostragem de Dados . . . 15
2.2.2 Conversor Analógico Digital (ADC) . . . 15
2.2.3 Inter-IC Sound (I2S) . . . 16
2.2.4 Acesso Direto à Memória (DMA) . . . 16
2.3 TRANSFORMADA RÁPIDA DE FOURIER (FFT) . . . 17
2.4 TEORIA MUSICAL . . . 18
2.4.1 Oitava . . . 18
2.4.2 Temperamento Igual . . . 18
2.4.3 Número de Notas MIDI . . . 18

3 ESTADO DA ARTE . . . 19
3.1 WII FIT DA NINTENDO . . . 19
3.2 GAMIFICAÇÃO NA REABILITAÇÃO FÍSICA DE AMPUTADOS . . . 20
3.3 RELEVÂNCIA PARA O SONICUBE . . . 21

4 DESENVOLVIMENTO . . . 22
4.1 LEVANTAMENTO DE REQUISITOS . . . 22
4.2 PESQUISA E SELEÇÃO DE TECNOLOGIAS . . . 23
4.2.1 Microcontroladores . . . 23
4.2.2 Sensores de som . . . 23
4.3 PROJETO DO SISTEMA . . . 24
4.3.1 Hardware . . . 24
4.3.2 Software . . . 25
4.4 IMPLEMENTAÇÃO DO SISTEMA . . . 27

5 RESULTADOS E ANÁLISE . . . 31

6 CONCLUSÕES E TRABALHOS FUTUROS . . . 34

REFERÊNCIAS 35

1

INTRODUÇÃO

Jogos digitais sempre fizeram muito sucesso entre os jovens, mas nos últimos anos essa popularidade vem aumentando cada vez mais. Conforme a Pesquisa Game Brasil (PGB) 2024, 73,9% da população no Brasil afirma jogar algum tipo de jogo digital. Comparado ao ano de 2023, o dado apresenta um aumento de 3,8 pontos percentuais. A pesquisa também mostra que essa porcentagem é ainda maior entre crianças e adolescentes menores de 15 anos, alcançando 82,8% deles, sendo essa uma fase em que a educação está muito presente. Segundo Mauro Berimbau, Consultor da Go Gamers e professor da Escola Superior de Propaganda e Marketing (ESPM) "Percebemos que os games têm aparecido com maior frequência no cotidiano do brasileiro e isso mostra o grande impacto que os jogos digitais têm na vida dos jogadores. As pessoas têm feito amizades e construído uma identidade da qual elas se orgulham" (GOGAMERS, 2024). Isso demonstra uma tendência em inserir os jogos em vários aspectos da nossa vida, do entretenimento à educação.

Assim, há um crescente interesse pela gamificação e pela criação de interfaces naturais de usuário. O Gamification Global Market Report 2024 indica que, este ano o mercado de gamificação irá crescer de 18 bilhões para cerca de 23 bilhões de dólares, um crescimento de cerca de 25%, e espera-se que cresça até cerca de 57 bilhões de dólares até 2028. Na educação, um estudo científico revela que alunos que tiveram um ensino utilizando a gamificação tiveram uma melhoria de desempenho de até 90% em relação àqueles que só tiveram palestras (Legaki *et al.*, 2020).

A música e o aprendizado musical são conhecidos por desenvolver habilidades cognitivas, motoras, mentais e sociais do indivíduo (Silva *et al.*, 2022). No entanto, muitos métodos tradicionais podem ser desmotivadores para iniciantes. Então, ao tornar o aprendizado musical mais divertido e envolvente, utilizando a gamificação para manter os usuários motivados e engajados e transformar a prática musical em uma experiência de jogo, espera-se que mais pessoas, especialmente jovens, se interessem em aprender música.

O avanço tecnológico tem permitido a criação de interfaces cada vez mais inovadoras e interativas, transformando a maneira como interagimos com dispositivos e sistemas. Dentro desse contexto, o presente projeto, denominado SoniCUBE, propõe o desenvolvimento de um sistema embarcado que integra música e jogos eletrônicos, proporcionando uma nova experiência

de entretenimento e estimulando o aprendizado musical através da prática lúdica.

O principal objetivo deste projeto é explorar a interseção entre música e tecnologia, criando uma ferramenta que não apenas diverte, mas também educa. O SoniCUBE permite aos usuários controlar jogos eletrônicos, através de instrumentos musicais, em dispositivos que reconhecem joysticks bluetooth, mapeando notas musicais para os botões de um joystick, como ilustra a figura 1. Utilizando uma plataforma ESP32 e um sensor de som MAX9814 para captar e interpretar notas musicais, esta abordagem visa atender a músicos, gamers e estudantes de música, oferecendo uma alternativa criativa e interativa para o aprendizado e entretenimento.

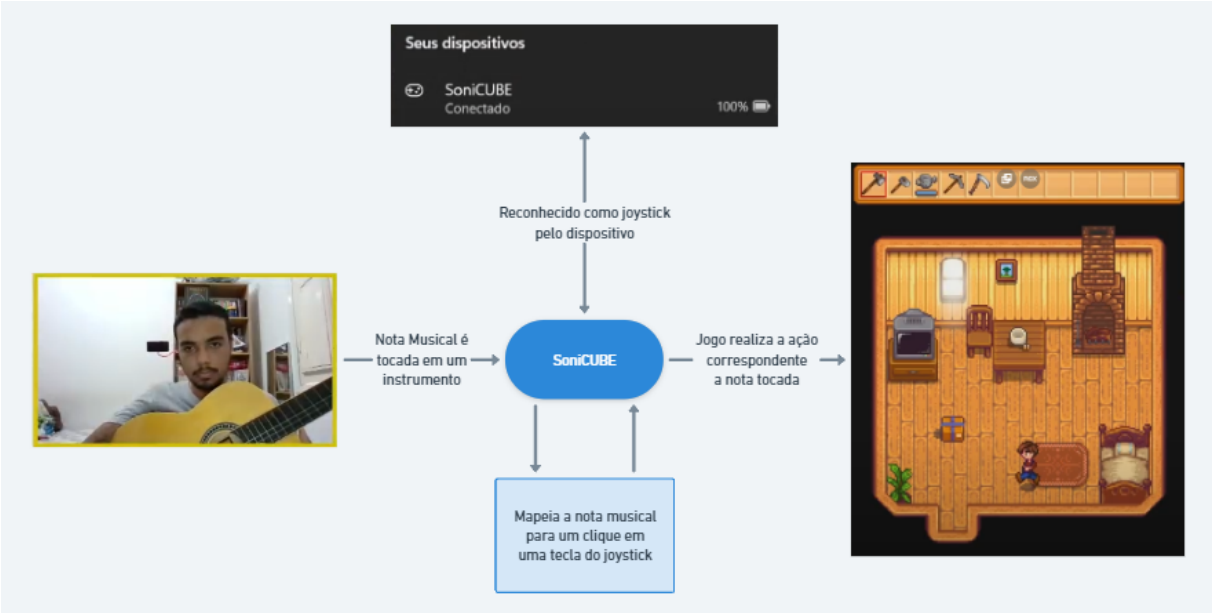

Figura 1: Funcionamento do SoniCUBE

Esse trabalho toma como inspiração o projeto Musical Gamepad realizado por mim e colegas, na disciplina de Computação Musical, do CIn-UFPE, lecionada pelo professor Giordano Cabral, onde tentamos outra abordagem para solucionar o problema. O Musical Gamepad utiliza o próprio computador e um microfone através de um projeto python para captar e interpretar as notas musicais e mapeá-las para os botões do joystick. Então a ideia por trás dessa nova abordagem utilizando a ESP32 é trazer uma série de melhorias para o projeto, como custo, portabilidade, acessibilidade, maior tolerância a ruídos, consumo de energia, etc.

Esta introdução apresenta uma visão geral do SoniCUBE, destacando sua relevância e inovação. Nas seções subsequentes, serão detalhadas as bases teóricas e tecnológicas do projeto, a metodologia de desenvolvimento, os resultados obtidos e as conclusões. A seção 2 aborda os fundamentos teóricos e tecnologias utilizadas. O estado da arte é demonstrado na seção 3. A metodologia e desenvolvimento são apresentados na seção 4. Os resultados e análises estão na seção 5. Por fim, a seção 6 traz as conclusões e perspectivas futuras do trabalho.

2

REVISÃO TEÓRICA

Conforme apresentado no capítulo anterior, a gamificação tem se mostrado uma área de grande potencial, oferecendo novas formas de ensino e entretenimento. No âmbito do aprendizado musical, em particular, a gamificação tem o potencial de engajar e motivar os alunos. O projeto SoniCUBE surge neste contexto, propondo uma nova maneira de interagir com jogos eletrônicos através da música. Dessa forma, a proposta se encaixa como um problema de processamento de audio em tempo real, em que é preciso ter baixa latência e precisão, assim, a implementação foi dividida em algumas partes, cada uma com seus próprios conceitos teóricos e técnicos que serão explicados nesta revisão.

2.1 PLATAFORMA DE DESENVOLVIMENTO

Pode-se afirmar que o SoniCUBE, em sua essência, é um sistema embarcado para uma aplicação em tempo real. Assim, alguns conceitos precisam ser tratados nesse tópico.

2.1.1 Sistemas Embarcados

Um sistema embarcado é um computador especializado que faz parte de um dispositivo maior, frequentemente de uso específico, dedicado a realizar funções particulares. Ao contrário dos computadores de propósito geral, como PCs ou laptops, os sistemas embarcados são projetados para executar uma ou poucas tarefas específicas, otimizando-se para eficiência, confiabilidade e desempenho em tempo real.

Nesses sistemas, todos os componentes de hardware tendem a ficar organizados em uma mesma placa, conhecida como Placa de Circuito Impresso (PCB), essas feitas de finas tiras de fibra de vidro, e com as conexões elétricas impressas em cobre.

Os sistemas embarcados têm como sua unidade central um microcontrolador, que por sua vez são circuitos integrados compactos projetados para realizar tarefas específicas de controle em sistemas embarcados. Os microcontroladores combinam um processador, memórias voláteis e não voláteis e periféricos de entrada/saída em um único chip. Alguns dos mais famosos são os PIC da microchip, AVR da atmel e ESP da espressif, cada um com suas particularidades, fazendo

com que cada um se sobressaia em determinadas aplicações.

Mas, o hardware não é a única coisa limitada nos sistemas embarcados, o software deve agir de forma determinística, fazendo sempre a mesma coisa ou sempre reagindo a um evento rápido o suficiente. Resumindo, um sistema embarcado é um sistema computadorizado que o seu propósito depende de sua aplicação. (White, 2011)

Exemplos de sistemas embarcados utilizados em jogos seriam os próprios joystics dos videogames que no seu interior possuem uma pcb com todo o sistema e a plataforma Wii Fit da Nintendo.

2.1.2 Plataforma ESP32

A plataforma ESP32, desenvolvida pela Espressif Systems, é uma solução de baixo custo e alta performance, altamente adequada para projetos que requerem conectividade e processamento de dados em tempo real. O ESP32 é um microcontrolador com capacidades de Wi-Fi e Bluetooth integradas, projetado para ser eficiente em termos de energia, o que o torna ideal para aplicações móveis e dispositivos Internet das Coisas (IoT).

O ESP32 é equipado com um processador dual-core Xtensa LX6, capaz de operar a uma frequência de até 240 MHz. Ele possui uma memória SRAM de até 520 KB e suporta diversos protocolos de comunicação, incluindo SPI, I2C, I2S, UART, e PWM, o que o torna extremamente versátil para diversas aplicações. Além disso, o ESP32 integra um Conversor Analógico Digital (ADC) de 12 bits, essencial para a leitura de sensores analógicos como o módulo de som MAX9814 utilizado no SoniCUBE.

Uma das principais vantagens do ESP32 é sua capacidade de comunicação sem fio. Com suporte integrado para Wi-Fi 802.11 b/g/n e Bluetooth v4.2 BR/EDR e BLE, o ESP32 pode facilmente se conectar a redes sem fio e a dispositivos Bluetooth, proporcionando grande flexibilidade na troca de dados.

Projetado para ser eficiente em termos de consumo de energia, o ESP32 possui vários modos de operação de baixa potência, permitindo que ele funcione por longos períodos em dispositivos alimentados por bateria. Os modos de baixa potência, como o modo deep sleep, são essenciais para prolongar a vida útil da bateria em aplicações móveis.

A plataforma ESP32 destaca-se como uma solução robusta e acessível para o desenvolvimento de sistemas embarcados, proporcionando uma base sólida para o SoniCUBE e outros projetos que exigem conectividade e processamento eficiente de dados.

2.1.3 Dispositivo de Interface Humana (HID)

O Dispositivo de Interface Humana (HID) é uma classe de dispositivo projetada para substituir os conectores tradicionais de estilo PS/2 por um driver USB genérico, oferecendo suporte a uma ampla gama de dispositivos de interface humana, como mouses, teclados, joysticks, entre outros. Antes do desenvolvimento do HID, os dispositivos precisavam se basear em drivers

predefinidos, como os de teclado e mouse, o que limitava a inovação, pois novas funcionalidades precisavam ser implementadas sobre protocolos existentes ou exigiam a criação de drivers específicos, uma tarefa complexa e demorada. O HID facilita o desenvolvimento de novas tecnologias, fornecendo interfaces padronizadas, extensíveis e facilmente programáveis.

Embora tenha começado como um padrão USB, o HID foi projetado para ser independente de barramento, sendo hoje utilizado em outros tipos de comunicação, como Bluetooth. Com o suporte HID, um dispositivo programável, como um microcontrolador, pode ser identificado e operado como um joystick ou outro dispositivo de interface humana por outros sistemas, ampliando as possibilidades de interação e controle em diversos contextos.

2.2 COMUNICAÇÃO COM O SENSOR

Para termos informação de som, um microfone precisa ser utilizado e integrado ao sistema, assim, alguns conceitos são extremamente necessários:

2.2.1 Amostragem de Dados

A amostragem de dados é um passo crítico no processamento de sinais, garantindo que os dados capturados pelos sensores sejam representativos das frequências sonoras presentes no ambiente.

Para isso, os sinais do sensor são lidos várias vezes por segundo, a uma taxa chamada taxa de amostragem, geralmente expressa em Hertz (Hz), que significa a quantidade de amostras lidas em um segundo, ou seja, 1 Hz significa 1 amostra por segundo, já 50 Hz, 50 amostras por segundo. A taxa de amostragem afeta diretamente a qualidade do sinal lido, se ela for muito baixa, você pode perder informação e acabar gerando erro ou distorções e se ela for muito alta, acontece o que é chamado de oversample, que pode gerar ruído e/ou redundância aos seus dados.

Para saber a taxa de amostragem ideal, Nyquist provou através de um teorema que se a taxa de amostragem for duas vezes maior que a maior frequência do sinal que se quer amostrar, esse sinal pode ser perfeitamente recuperado.

2.2.2 Conversor Analógico Digital (ADC)

Um ADC é um componente essencial em sistemas embarcados que interagem com sinais do mundo real, como som, luz, temperatura, etc. O ADC converte sinais analógicos, que são contínuos, podem assumir qualquer valor dentro de um intervalo de tempo e muito complexos para serem processados, em sinais digitais, que são discretos e podem assumir apenas valores específicos.

O ADC funciona amostrando o sinal analógico em intervalos de tempo regulares e atribuindo a cada amostra um valor digital correspondente, com base na amplitude do sinal no momento da amostragem, esses valores podem variar de 0 a 1024 ou até de 0 a 4095 ou

outras faixas de valor, dependendo da quantidade de bits que o ADC utiliza. Este processo de amostragem e quantização permite que o sinal analógico contínuo seja representado como uma sequência de valores digitais, que podem ser processados pelo microcontrolador.

2.2.3 Inter-IC Sound (I2S)

Após o sinal de áudio ser convertido para digital pelo ADC, um protocolo de comunicação é responsável por entregar esses dados a memória do sistema.

O I2S é um padrão de comunicação serial usado para transmitir áudio digital entre dispositivos integrados. Diferente de outros protocolos de comunicação como I2C ou SPI, que são usados para transmitir dados gerais, o I2S é especificamente projetado para áudio, garantindo alta fidelidade e sincronização precisa.

O I2S utiliza três linhas principais para comunicação:

- **Clock de Bit (BCLK):** Sincroniza a transmissão dos bits de dados.
- **Clock de Quadro (LRCLK):** Indica o início de um novo quadro de áudio (esquerdo ou direito).
- **Linha de Dados (SD):** Transporta os dados de áudio.

O I2S permite a transmissão de dados de áudio digital de alta qualidade com baixa latência, o que é essencial para aplicações que exigem processamento de som em tempo real.

2.2.4 Acesso Direto à Memória (DMA)

Acesso Direto à Memória (DMA) é uma característica de sistemas de computador que permite que certos subsistemas de hardware acessem a memória principal (RAM) de forma independente da Unidade Central de Processamento (CPU). Isso significa que a transferência de dados entre a memória e outros componentes do sistema pode ocorrer sem a intervenção constante da CPU, liberando-a para executar outras tarefas.

Em um sistema com DMA, um controlador DMA gerencia a transferência de dados. Quando um dispositivo de hardware, como um sensor de som, precisa enviar dados para a memória, ele solicita ao controlador DMA para iniciar a transferência. O controlador DMA então executa a transferência de dados diretamente entre a memória e o dispositivo de hardware, sem envolver a CPU. Quando o buffer do DMA está cheio, ocorre uma interrupção, e a CPU pode processar esses dados. Este processo é realizado de forma eficiente e com mínima latência.

Para utilizar o DMA, é necessário configurá-lo adequadamente, o que envolve a definição de duas variáveis principais: a largura do buffer e a quantidade de buffers utilizados. Se a largura do buffer for pequena, o número de interrupções aumenta, pois o buffer enche mais rapidamente. Por outro lado, se o buffer for grande, pode haver um problema de latência, pois a CPU terá que esperar mais tempo para acessar os dados. Durante o tempo em que o controlador

DMA está escrevendo em um buffer, a CPU não pode ler esse buffer e vice-versa. Portanto, é essencial utilizar pelo menos dois buffers para otimizar o uso do DMA, permitindo que, enquanto um buffer está sendo preenchido, o outro possa ser lido pela CPU. Esta configuração garante uma transferência de dados mais contínua e eficiente, minimizando a latência e melhorando o desempenho geral do sistema.

2.3 TRANSFORMADA RÁPIDA DE FOURIER (FFT)

O processamento dos dados capturados é uma etapa crucial para transformar as informações coletadas pelos sensores em dados significativos e úteis para a aplicação. Nesse contexto, a Transformada Rápida de Fourier (FFT) é uma ferramenta ideal para a análise e identificação das frequências presentes nas ondas sonoras capturadas.

A FFT é, na prática, uma implementação eficiente da Transformada Discreta de Fourier (DFT), que reduz significativamente o número de operações necessárias para calcular a DFT, tornando possível o processamento em tempo real das amostras de áudio. Isso permite uma análise rápida e precisa das componentes de frequência que compõem o sinal.

A DFT, por sua vez, é uma técnica matemática que converte um sinal discreto do domínio do tempo para o domínio da frequência. Em essência, a DFT decompõe um sinal em uma soma de senos e cossenos com diferentes frequências, permitindo a análise detalhada das componentes de frequência presentes no sinal original.

Essa técnica transforma uma sequência finita de amostras de um sinal em uma sequência finita de coeficientes de frequência. Dados N pontos de amostra $x[n]$ no domínio do tempo, a DFT é definida por:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi kn/N} \quad \text{para } k = 0, 1, \ldots, N-1 \tag{2.1}$$

onde:

- $x[n]$ representa o sinal de entrada no domínio do tempo.
- $X[k]$ é o coeficiente da frequência k no domínio da frequência.
- N é o número de pontos considerados na DFT.
- e é a base do logaritmo natural.
- j é a unidade imaginária.

A aplicação da FFT nesse projeto permite uma análise eficiente das frequências dominantes no sinal capturado, sendo um passo essencial para o correto mapeamento das notas musicais em botões de controle no sistema desenvolvido.

2.4 TEORIA MUSICAL

2.4.1 Oitava

Na música, uma oitava refere-se ao intervalo entre duas notas em que a frequência de vibração da nota superior é exatamente o dobro da frequência da nota inferior. Essa relação é percebida como a mesma nota, mas em alturas diferentes, e é fundamental na construção de escalas musicais e na afinação de instrumentos (of Encyclopaedia Britannica, 2024).

2.4.2 Temperamento Igual

O temperamento igual é um sistema de afinação amplamente utilizado na música ocidental, onde a oitava é dividida em 12 semitons de tamanhos iguais. Esse sistema permite que instrumentos de teclado possam tocar em todas as tonalidades com perdas mínimas de entonação, facilitando a modulação entre diferentes tons. Historicamente, substituiu sistemas de afinação anteriores baseados em intervalos acústicos puros, que, embora mais consonantes em certas tonalidades, apresentavam desafinações perceptíveis em outras. No temperamento igual, a relação de frequência entre notas adjacentes (semitons) é constante e pode ser expressa pela razão $2^{\frac{1}{12}}$, que é aproximadamente 1.05946 (of Encyclopaedia Britannica, 2019).

2.4.3 Número de Notas MIDI

O padrão Musical Instrument Digital Interface (MIDI) adota o sistema de temperamento igual e segue o padrão internacional onde o tom A4 (Lá) — a nota acima do C4 (Dó central) em um teclado — vibra a 440 Hz. No padrão MIDI, A4 corresponde ao número 69. As frequências das notas MIDI são determinadas pela equação:

$$f = 440 \cdot 2^{\frac{n-69}{12}} \tag{2.2}$$

Essa equação permite calcular a frequência de qualquer nota MIDI, estabelecendo uma correspondência direta entre o número MIDI e a altura musical. Assim, o sistema MIDI padroniza a afinação de instrumentos digitais e sintetizadores, assegurando consistência na produção musical (of Inpired Acoustics, 2024).

3

ESTADO DA ARTE

Nesta seção, são explorados estudos e tecnologias que se relacionam de alguma forma ao SoniCUBE. A pesquisa foi realizada em três bases de dados distintas: Scielo, Biblioteca Digital Brasileira de Teses e Dissertações (BDTD) e o Catálogo de Teses e Dissertações da Capes, abrangendo tanto publicações em português quanto em inglês. Os resultados não revelaram trabalhos com a mesma abordagem específica proposta neste projeto, destacando a originalidade e inovação do SoniCUBE.

Tabela 1: Resultado da pesquisa às bases de dados

Palavras-Chave	**SciELO**	**BDTD**	**Catálogo da CAPES**
Sistemas Embarcados	277	1602	2345
Sistemas Embarcados e Gamificação	0	2	1
Sistemas Embarcados e Música	0	2	2
Sistemas Embarcados, Gamificação e Música	0	0	0

Além das palavras-chave principais, termos como New Interfaces for Musical Expression (NIME) e Digital Musical Instrument foram utilizados em combinação com Sistemas Embarcados. No entanto, essas buscas também não produziram resultados diretamente relevantes ao escopo do projeto.

Embora não tenham sido encontrados estudos diretamente alinhados com o SoniCUBE, exemplos notáveis de sistemas embarcados aplicados à gamificação incluem o Wii Fit da Nintendo e o trabalho acadêmico intitulado "Gamificação na Reabilitação Física de Amputados: Uma Abordagem Baseada em Jogo Sério". Esses exemplos demonstram como sistemas embarcados para gamificação têm sido aplicados em diversas áreas para melhorar a experiência do usuário e alcançar objetivos específicos.

3.1 WII FIT DA NINTENDO

O Wii Fit, desenvolvido pela Nintendo, é um exemplo clássico de como sistemas embarcados podem ser utilizados para gamificação. Lançado em 2007, o Wii Fit utiliza uma plataforma

chamada Wii Balance Board, equipada com sensores de pressão que detectam o peso e o centro de gravidade do usuário. Este sistema embarcado permite que os jogadores interajam com jogos de fitness de uma maneira intuitiva e envolvente.

O Wii Fit combina exercícios físicos com elementos de jogo, incentivando os usuários a se movimentarem e melhorarem sua condição física enquanto se divertem. A plataforma oferece uma variedade de atividades, incluindo yoga, aeróbica, equilíbrio e jogos de força, cada uma projetada para desafiar e motivar os usuários a manterem uma rotina de exercícios. Estudos mostram que a gamificação de exercícios físicos, como visto no Wii Fit, pode aumentar a adesão e o engajamento em programas do campo da saúde, ou até em casos de reabilitação (Tripette *et al.*, 2017).

Figura 2: Wii Fit Balance Board

3.2 GAMIFICAÇÃO NA REABILITAÇÃO FÍSICA DE AMPUTADOS

Outro exemplo significativo é o trabalho "Gamificação na Reabilitação Física de Amputados: Uma Abordagem Baseada em Jogo Sério"(Melo *et al.*, 2023), que investiga o uso de jogos sérios para melhorar a reabilitação física de pacientes amputados. Este estudo desenvolveu um sistema embarcado que utiliza sensores de movimento para capturar os movimentos dos pacientes durante as sessões de fisioterapia. Os dados coletados são então utilizados para controlar um jogo sério, transformando exercícios de reabilitação em atividades lúdicas e motivadoras.

A abordagem baseada em jogo sério visa aumentar a motivação e o engajamento dos pacientes, tornando o processo de reabilitação mais interessante e menos oneroso. Resultados preliminares indicam que a gamificação pode melhorar significativamente a adesão dos pacientes aos programas de reabilitação e levar a melhores resultados clínicos. Este estudo exemplifica como sistemas embarcados podem ser aplicados em contextos de saúde para criar soluções inovadoras que combinam terapia e entretenimento.

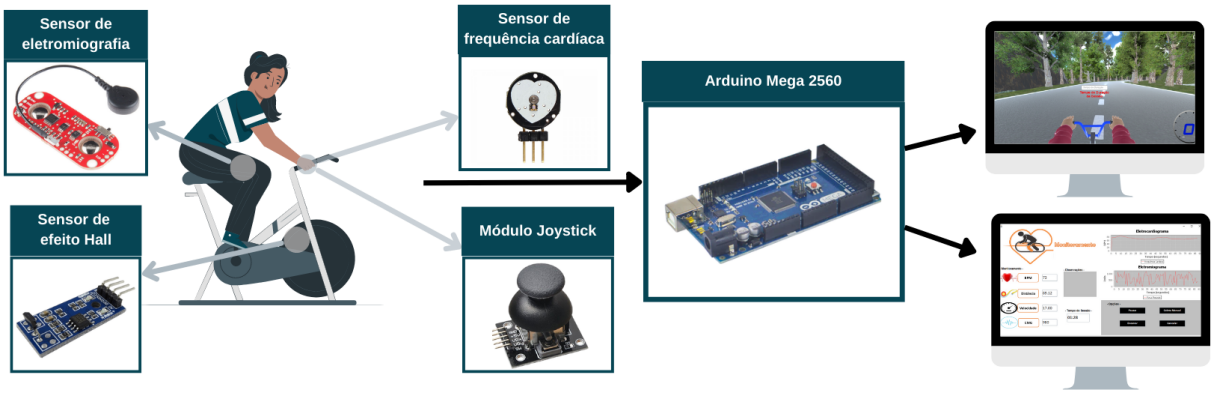

Figura 3: Arquitetura do projeto (Melo *et al.*, 2023)

3.3 RELEVÂNCIA PARA O SONICUBE

Os exemplos do Wii Fit e da gamificação na reabilitação física ilustram o potencial dos sistemas embarcados em aplicações de gamificação, evidenciando como essas tecnologias podem ser utilizadas para transformar atividades tradicionais em experiências interativas e envolventes. O SoniCUBE segue essa linha, utilizando um sistema embarcado para integrar música e jogos eletrônicos, oferecendo uma nova ferramenta para aprendizado musical e entretenimento.

Ao incorporar conceitos de gamificação e utilizando sensores avançados, o SoniCUBE visa não apenas entreter, mas também educar os usuários, proporcionando uma maneira divertida e eficaz de aprender música. A relevância desses estudos para o SoniCUBE reside na demonstração de que a gamificação, quando aplicada corretamente, pode aumentar o engajamento, a motivação e os resultados em várias áreas, desde o fitness até a reabilitação e educação.

Esta seção forneceu uma visão geral do estado da arte em sistemas embarcados para gamificação, destacando a importância e os benefícios dessas abordagens. Nas seções seguintes, discutiremos a metodologia específica utilizada para o desenvolvimento do SoniCUBE e os resultados obtidos.

4

DESENVOLVIMENTO

O desenvolvimento do SoniCUBE foi estruturado de acordo com um cronograma, abrangendo desde a definição de requisitos até a realização de testes e validação. Este cronograma assegura que cada etapa do projeto seja devidamente planejada e executada dentro dos prazos estabelecidos. A seguir, detalhamos cada uma das fases de desenvolvimento.

4.1 LEVANTAMENTO DE REQUISITOS

Na fase inicial, foi realizado o levantamento de requisitos do SoniCUBE. Este processo envolveu a coleta de informações sobre as necessidades dos usuários, funcionalidades esperadas do sistema e especificações técnicas. Como esse projeto foi baseado em outra abordagem de solução do problema. Foram realizadas seções de brainstorming para idealizar em como solucionar o problema no contexto de sistemas embardados. Para auxiliar no levantamento de requisitos foi utilizado, como um guia, o Guia de Elicitação de Requisitos para Sistemas Embarcados (GERSE) (Ossada *et al.*, 2012), um artigo publicado em um workshop de engenharia de requisitos na PUC-Rio em que é proposto uma série de atividades de organização, definição, etc, em que cada artefato de saída é uma parte importante para os requisitos.

Assim, foi gerado, de forma resumida, o seguinte artefato: O principal objetivo do produto é combinar entretenimento e educação musical, permitindo que os usuários joguem utilizando instrumentos musicais e aprendam música de forma interativa. Espera-se que o produto mapeie notas musicais para um joystick, oferecendo uma experiência inovadora de jogo e aprendizado. A plataforma será direcionada a músicos, gamers e alunos de música, e deverá ser reconhecida como um joystick pelo dispositivo em que for conectado. Os principais stakeholders incluem instituições de ensino, professores de música, músicos e gamers. Os requisitos funcionais incluem a capacidade de ler ondas sonoras, transformar essas ondas em frequências, identificar notas musicais e mapear essas notas para os botões do joystick. Em termos de requisitos não-funcionais, o produto deve respeitar restrições de memória e velocidade de processamento, e ser capaz de lidar com ruídos sonoros do ambiente. O SoniCUBE será alimentado via USB, terá uma interface Bluetooth, e deverá ser pequeno e portátil, com um microfone integrado.

4.2 PESQUISA E SELEÇÃO DE TECNOLOGIAS

Baseada no levantamento de requisitos e em conjunto ao GERSE, foi conduzida uma pesquisa abrangente em busca de microcontroladores que funcionassem como HID e sensores de som (microfones) que fossem mais adequados para o desenvolvimento do SoniCUBE.

4.2.1 Microcontroladores

Durante a pesquisa, os dois principais microcontroladores que poderiam ser utilizados seriam o Atmega com a placa Arduino e o ESP com a placa ESP32, pois ambas possuem interface com os sensores, e podem funcionar como HID, o Arduino através do USB e a ESP32 através do bluetooth, então o critério de escolha foi a capacidade de processamento e a memória disponível, como precisaríamos realizar transformadas de Fourier que é um algoritmo muito custoso espacialmente e em tempo real, o ideal seria utilizar o ESP como microcontrolador.

4.2.2 Sensores de som

Já em relação aos sensores de som, a escolha era entre o KY-038 e o MAX9814. O KY-038, mostrado na figura 4, possui um microfone sensível à capacitância com alcance de 50-10KHz, um circuito de amplificação e, para definir o nível de som, um potenciômetro.

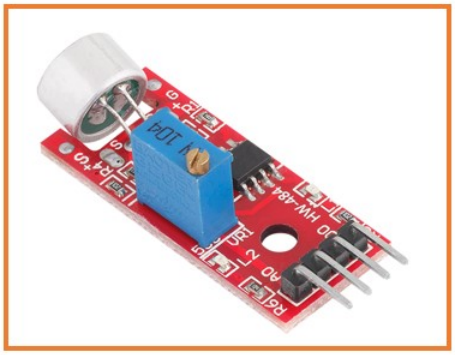

Figura 4: Sensor de som KY-038

O MAX9814 conta com um microfone de eletreto de 20-20KHz, sendo que o módulo é projetado com um Circuito Integrado (CI) destinado a amplificação, o MAX9814 e um Controle Automático de Ganho (AGC). De forma mais detalhada, o CI garante ao módulo AGC para equilibrar o valor lido na entrada, sendo que quando o som detectado é excessivamente alto, o controle o abaixa para que não haja uma sobrecarga no amplificador, e em caso contrário, em que a onda sonora está muito baixa, o amplificador aumenta para que seja realizada a perfeita leitura.

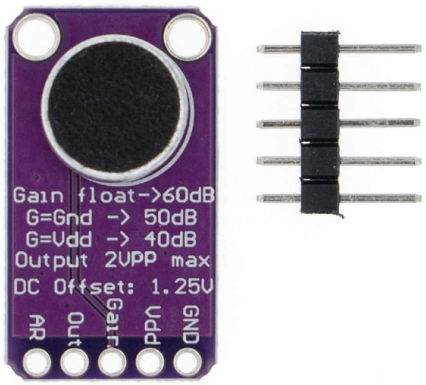

Figura 5: Sensor de som MAX9814

Então, devido ao MAX9814 ter um alcance maior de frequências e possuir um AGC, ele foi escolhido para o projeto em vez do KY-038, que requereria ajustes manuais do potenciômetro dependendo do ruído ambiente. A Figura 5 ilustra o sensor escolhido para o projeto.

Resultando de forma resumida no seguinte artefato: Para o desenvolvimento do SoniCUBE, foi definida a necessidade de um módulo sensor de som MAX9814, composto por cinco pinos: Vcc, Gnd, Out, Ar e Gain. O atuador principal será o próprio microcontrolador, que utilizará a interface bluetooth para se conectar aos dispositivos externos. A interação com o usuário será realizada por meio do dispositivo conectado via bluetooth, permitindo ao usuário utilizar o som como um joystick. A porta de comunicação externa será exclusivamente bluetooth. O microcontrolador escolhido para o projeto é o ESP através da placa de prototipação ESP32. As variáveis de ambiente incluem a amplitude do som medida pelo sensor, variando entre 0 e 4095.

4.3 PROJETO DO SISTEMA

Com os requisitos definidos e as tecnologias selecionadas, iniciou-se o projeto detalhado do sistema. Esta fase envolveu a criação de esquemas de circuitos, e a definição das interfaces de comunicação entre os componentes. Foram elaborados planos para a implementação do software embarcado, incluindo o desenvolvimento dos algoritmos para a detecção de notas musicais e o mapeamento para os botões do joystick.

4.3.1 Hardware

O esquema mostrado na figura 6 será o circuito a ser implementado no projeto, onde é utilizado a ESP32 e o sensor MAX9814, com os fios vermelhos representando a conexão com o Vdd 3.3 V e os fios pretos com o GND, com um detalhe importante que é a ligação da porta Gain do sensor em Vdd para um ganho de 40db e a saída de dados do sensor (Out) conectada a porta 34 da ESP32.

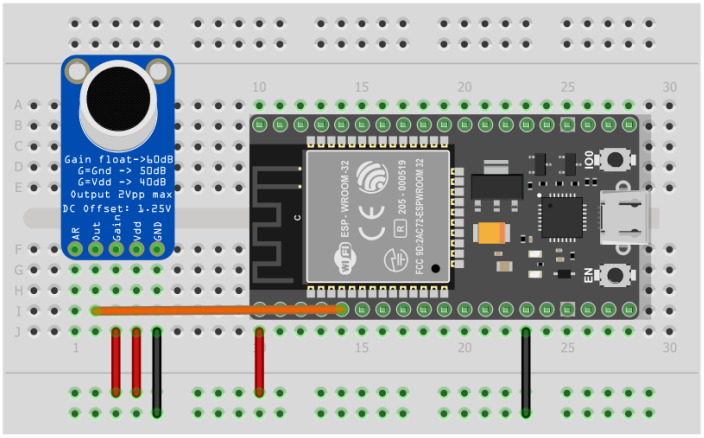

Figura 6: Esquema de circuito do SoniCUBE

4.3.2 Software

A aplicação terá dois fluxos concorrentes e que funcionam sem interrupção, a figura 7 mostra esse fluxo de forma completa.

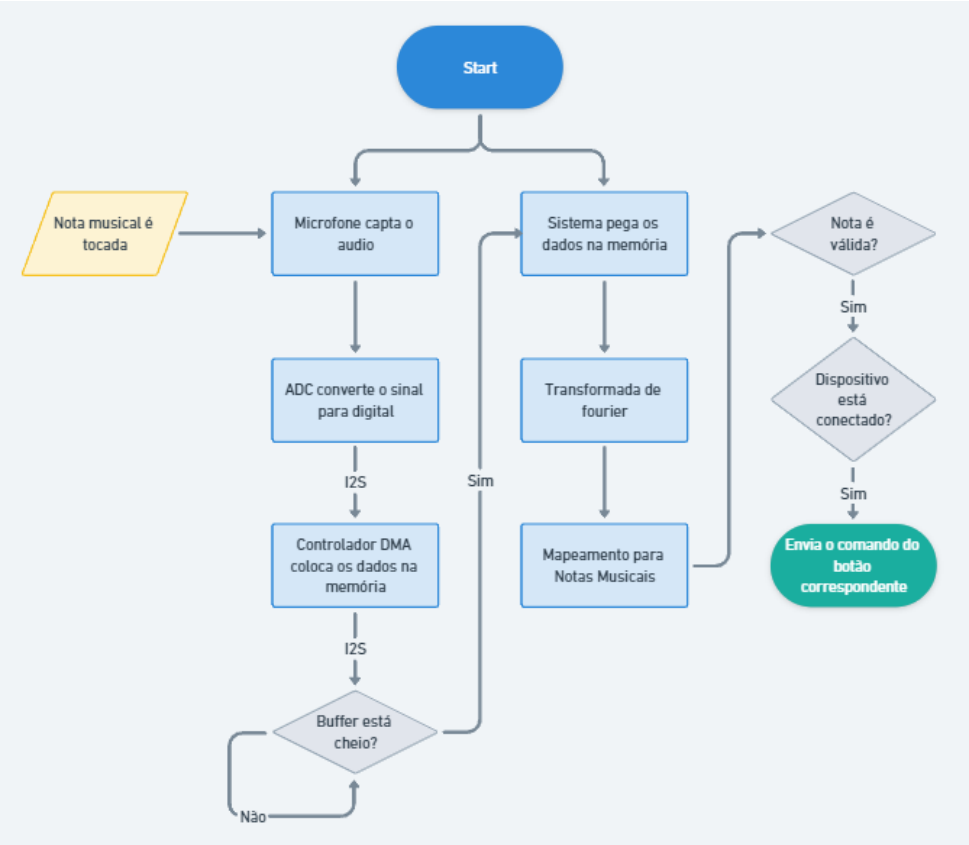

Figura 7: Fluxo completo de funcionamento do SoniCUBE

A figura 8 mostra o primeiro fluxo que é responsável pela parte de captação de dados da aplicação, onde de forma ininterrupta vai captar o áudio através do microfone e depositar esses

dados em um buffer na memória, quando esse buffer está cheio, esses dados são automaticamente levados para a próxima parte da aplicação em que serão tratados.

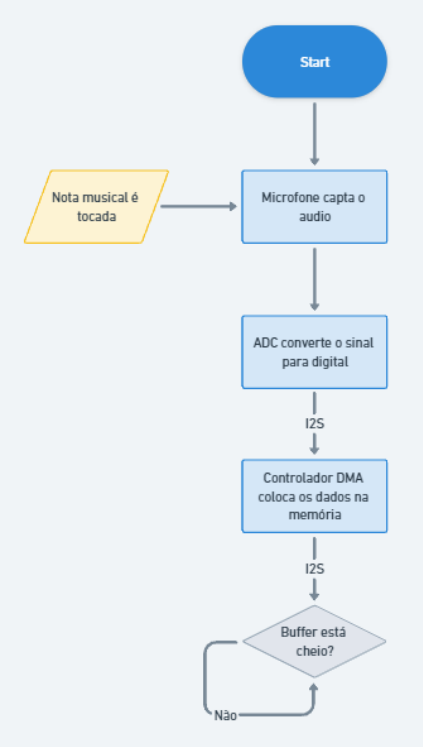

Figura 8: Fluxo da comunicação com sensor e amostragem de dados do SoniCUBE

Já o segundo fluxo, mostrado na figura 9, também funciona de forma ininterrupta e independente, sempre que os dados estiverem disponíveis a aplicação irá tratá-los através da transformada de fourier e mapeá-los para as notas musicais, e se possível um comando será enviado para o dispositivo conectado ao SoniCUBE.

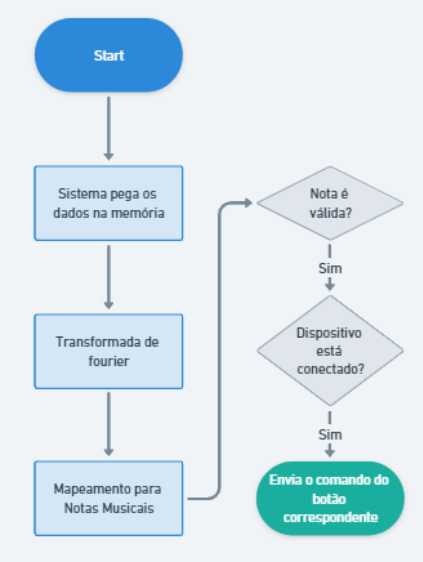

Figura 9: Fluxo de processamento dos dados e funcionamento do SoniCUBE

4.4 IMPLEMENTAÇÃO DO SISTEMA

Durante a fase de implementação, o foco foi a construção física e a codificação do SoniCUBE. A Figura 10 ilustra como os componentes eletrônicos foram montados e conectados conforme o projeto desenvolvido.

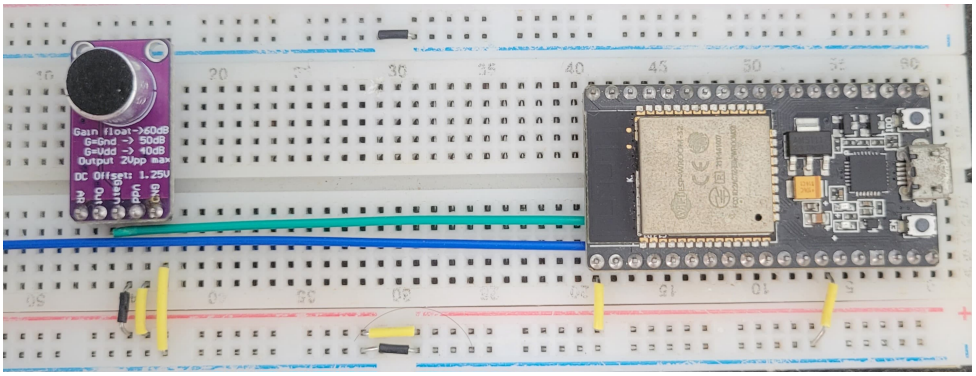

Figura 10: Circuito do SoniCUBE

O código-fonte para a ESP32 foi escrito e testado, assegurando que a captura e o processamento das notas musicais funcionassem corretamente. Diversas iterações de desenvolvimento e teste foram realizadas para ajustar e otimizar o sistema.

Na primeira iteração, integrou-se o sensor de som e implementou-se a amostragem dos dados. Nesta fase, foram configurados o ADC e o DMA através de três variáveis importantes: a taxa de amostragem, a largura do buffer DMA e a quantidade de buffers DMA. Assim, a ESP32, através do ADC interno, lê as tensões do sensor MAX9814 e utiliza o protocolo I2S para transferir esses dados para a memória. A taxa de amostragem escolhida foi de 40 KHz, conforme o teorema de Nyquist, permitindo a amostragem de sinais de até 20 KHz, que é o limite do sensor. Para as outras duas variáveis, escolheu-se 1024 e 2, respectivamente, resultando em um buffer pequeno. Enquanto um buffer estará sendo lido pela CPU, o outro está sendo preenchido pelo controlador DMA, resultando em uma taxa de transferência de 80 *KB/s*, conforme a equação 4.1, caracterizando uma análise de janela fixa.

$$TransferRate = \frac{BitsPerSample}{8} \cdot SampleRate$$
$$TransferRate = \frac{16bits}{8} \cdot 40KHz = 80KB/s \tag{4.1}$$

A Figura 11 mostra o resultado da impressão do buffer amostra por amostra, obtendo-se aproximadamente a forma de onda do sinal de áudio original. Isso permitiu concluir que o sensor foi integrado corretamente.

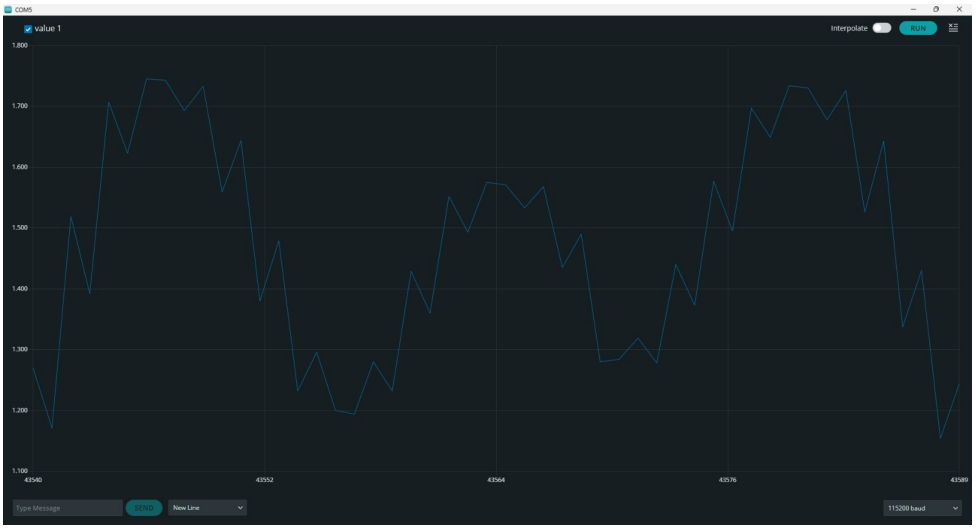

Figura 11: Forma de onda do sinal de áudio lido pelo sensor

Na segunda iteração, foi realizado o tratamento desses dados removendo a componente DC, centralizando as amostras em torno de 0 e aplicando esses dados a uma FFT com janela de Hann para obter a frequência com mais intensidade em um determinado período de tempo. Neste caso, a cada 25 *ms*, que é o período entre cada interrupção da CPU, conforme a equação 4.2.

$$T = \frac{BufferLength}{SampleRate}$$
$$T = \frac{1024}{40KHz} \approx 25ms \tag{4.2}$$

Essa metodologia permite que as notas sejam reconhecidas independentemente do instrumento musical utilizado, eliminando a necessidade de um tempo específico de reconhecimento para cada instrumento. A identificação baseia-se exclusivamente na frequência da nota, garantindo que o sistema funcione de maneira consistente e eficiente, independentemente da fonte sonora.

A Figura 12 mostra o resultado de aplicar a FFT a cada intervalo de tempo *T*. Utilizando um gerador de ondas, e comparando o valor calculado com o valor utilizado no gerador, verificou-se que a aplicação está calculando a frequência corretamente.

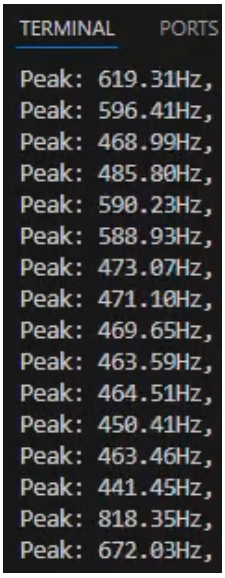

Figura 12: Saída do SoniCUBE com frequências

Na terceira iteração, foi realizado o mapeamento das frequências para as notas musicais utilizando a seguinte equação:

$$noteIndex = (12 \cdot \lfloor log_2(f/440) \rfloor + 69)\%12 \tag{4.3}$$

Essa equação é baseada no padrão MIDI e estabelece uma relação direta entre o número da nota MIDI e a frequência correspondente, independentemente da oitava em que a nota se encontra. Nessa equação, f representa a frequência a ser mapeada, enquanto *noteIndex* (o resto da divisão do índice da nota MIDI por 12) retorna um valor entre 0 e 11, correspondente às 12 notas musicais dentro de uma oitava. A Figura 13 ilustra a relação entre as frequências calculadas e suas respectivas notas, confirmando que o mapeamento está funcionando de forma precisa.

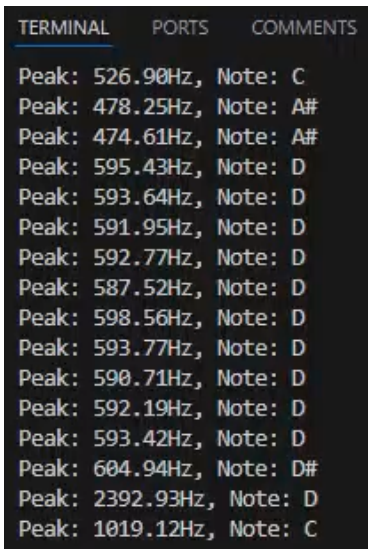

Figura 13: Saída do SoniCUBE com notas mapeadas

Na quarta e última iteração, implementou-se a interface Bluetooth, permitindo o uso do sistema como um joystick, funcionando de forma que ao sistema reconhecer uma nota válida o botão fica pressionado por 100 *ms*. O *noteIndex* foi mapeado para um botão do joystick, conforme mostrado na Tabela 2. Este mapeamento foi projetado para manter próximos os botões que são adjacentes nos modelos de joystick existentes, garantindo maior ergonomia e facilidade de uso para determinados instrumentos, como o piano. A Figura 14 demonstra esse mapeamento na prática, imprimindo o índice da nota e seu botão correspondente, possibilitando a verificação de que o mapeamento está correto.

Tabela 2: Mapeamento de notas para botões de joystick

noteIndex	0	1	2	3	4	5	6	7	8	9	10	11
Note	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
Joystick Button	Left	Down	Right	Up	LT	LB	RB	RT	X	A	B	Y

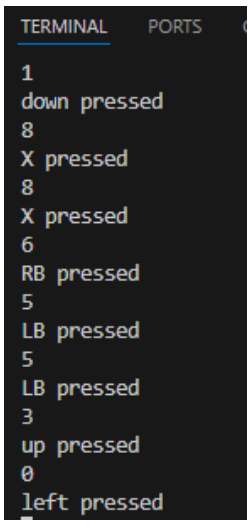

Figura 14: Saída do SoniCUBE com mapeamento do joystick

Durante a última iteração, identificou-se a necessidade de implementar um debounce no sistema. Como cada nota é lida a cada 25 *ms*, uma nota tocada por mais que esse período resulta em múltiplas leituras da mesma nota. Para resolver esse problema, o SoniCUBE foi configurado para reconhecer uma nota apenas após ela ser repetida 15 vezes consecutivas. Isso garantiu que apenas notas efetivamente tocadas fossem consideradas, evitando leituras falsas causadas por ruídos ou toques breves não intencionais ao custo de um aumento na latência.

5

RESULTADOS E ANÁLISE

A fase de testes e validação envolveu a verificação de todas as funcionalidades do SoniCUBE em diferentes cenários de uso. Primeiramente, utilizou-se um testador de joysticks online (Test, 2024) para verificar o funcionamento básico do sistema no celular, conforme ilustrado na Figura 15. Durante esse teste, cada nota musical foi tocada em um teclado, e observou-se se o botão correspondente na tela ficava laranja. Esse procedimento foi repetido para cada nota musical, permitindo a confirmação de que a conexão Bluetooth funcionava corretamente e que o mapeamento das notas para os botões do joystick estava preciso.

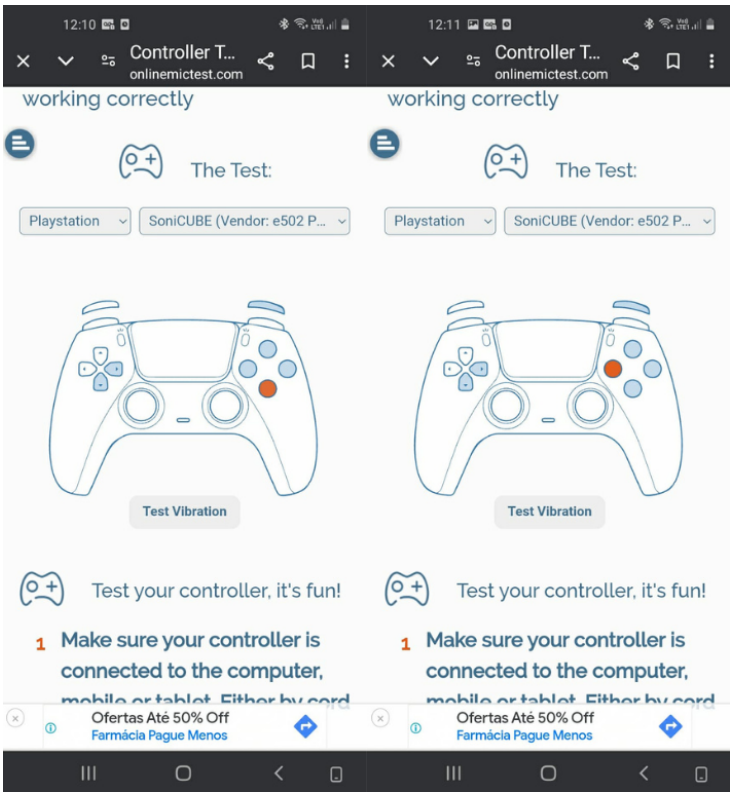

Figura 15: Teste básico do SoniCUBE

Para analisar o tempo de resposta do sistema, utilizou-se a função *millis()* da biblioteca Arduino, que permitiu calcular o intervalo de tempo desde o reconhecimento inicial da nota até o

envio da mesma via Bluetooth. Os resultados indicaram tempos de resposta variando entre 480 e 500 *ms*. Conforme ilustrado na equação 5.1, esse tempo de resposta é aproximadamente 500 *ms*.

$$delay = T \cdot debounceTimes + pressTime \tag{5.1}$$

$$delay = 25ms \cdot 15 + 100ms \approx 500ms$$

Estudos como o de (Long & Gutwin, 2018) investigam os efeitos de diferentes níveis de latência em jogos eletrônicos, sugerindo que a sensibilidade dos jogos à latência pode variar significativamente de acordo com o ritmo do jogo. Embora a relação precisa entre os requisitos temporais dos jogos e a quantidade de latência ainda não seja completamente compreendida, o estudo revela que, à medida que a latência aumenta, os jogadores tendem a adaptar suas técnicas de jogo, o que pode alterar a experiência original do jogo. Adicionalmente, (Claypool & Claypool, 2006) corrobora esses achados ao demonstrar que jogos menos dependentes do tempo de ação são menos sensíveis à latência, funcionando de maneira aceitável até 500 ou 1000 *ms* de atraso. Esses resultados indicam que, em determinados casos, é possível proporcionar uma experiência de jogo tolerável mesmo com uma latência de 500 *ms*.

Outros testes foram realizados em um computador com Windows. No Windows PC, outro passo é necessário antes da utilização, pois o Windows só tem compatibilidade nativa com os joysticks da Microsoft. Controles genéricos, como é o caso do SoniCUBE, precisam de um software adicional para torná-los compatíveis com o Windows. Para isso, foi utilizada a plataforma Steam.

Após a validação inicial do funcionamento, foram realizados testes reais utilizando diferentes instrumentos musicais, como flauta doce, teclado e violão, sendo os dois primeiros via app em um celular, como mostra a figura 16, em diferentes jogos dos mais simples aos mais atuais e complexos. Alguns dos jogos testados foram:

- **Tomb Raider:** Foi impossível jogá-lo devido à necessidade de utilizar os analógicos.
- **Terraria:** Permitiu realizar algumas ações, mas a jogabilidade foi muito limitada pela falta dos analógicos.
- **Stardew Valley:** Começou a mostrar uma boa utilização para o SoniCUBE, exceto por um botão que não estava disponível no nosso mapeamento. Contudo, através da própria Steam, poderia ser remapeado de outra nota menos utilizada.
- **Pokémon:** Este foi o jogo em que o SoniCUBE apresentou o melhor desempenho, sendo um jogo de turno, que não depende da velocidade das ações, e que necessita de poucos botões, o sistema se saiu muito bem.

Figura 16: Testes no computador com diferentes instrumentos

Essa validação mostrou que o SoniCUBE atende aos requisitos iniciais e funciona conforme o esperado em um ambiente real.

Os testes realizados com diferentes instrumentos demonstraram a eficácia do sistema em reconhecer notas musicais a partir da quinta oitava e mapear corretamente para os comandos do joystick. Jogos complexos que necessitam do uso de analógicos ou mais de doze botões não foram compatíveis com o SoniCUBE devido a limitação das 12 notas musicais, alguns requisitando uma configuração pessoal em cada jogo, mas jogos mais simples, como Stardew Valley, jogos de turno, como Pokémon, mostraram um resultado excelente. O sistema mostrou-se robusto na identificação de notas, mesmo em ambientes com níveis variados de ruído sonoro.

Realizando uma análise comparativa com o projeto Musical Gamepad, pode-se concluir que o SoniCUBE trouxe uma série de melhorias, como:

- **Acessibilidade:** O desempenho do SoniCUBE não depende do hardware de cada dispositivo, além de ser uma alternativa mais barata.
- **Maior proteção contra ruídos:** Por se tratar de um sistema embarcado, o SoniCUBE tem menos dispositivos que possam causar ruídos a aplicação.
- **Baixo Consumo de Energia:** A plataforma ESP32 é eficiente em termos de energia, ideal para aplicações móveis e alimentadas por bateria.
- **Portabilidade:** Devido ao seu tamanho pequeno e capacidade de operar independentemente, o SoniCUBE oferece excelente portabilidade, tornando o projeto utilizável em qualquer lugar.
- **Conectividade Bluetooth:** A capacidade de conectar-se diretamente a dispositivos móveis via Bluetooth e ser identificado como um joystick são grandes vantagens, permitindo um controle mais direto e intuitivo sem necessidade de cabos.

6

CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho apresentou o desenvolvimento do SoniCUBE, um sistema embarcado que permite o controle de jogos eletrônicos através de instrumentos musicais, mapeando notas musicais para os botões de um joystick. O objetivo principal de explorar a interseção entre música e tecnologia foi plenamente alcançado, visto que foi possível jogar utilizando notas musicais, proporcionando uma experiência de entretenimento e aprendizado musical.

O SoniCUBE atendeu aos requisitos iniciais e comparado ao projeto Musical Gamepad superou vários aspectos. Oferecendo maior portabilidade e eficiência energética devido ao uso do ESP32, além de uma conexão Bluetooth confiável que permitiu um controle mais intuitivo e direto e uma captura de dados mais eficiente e com menor latência, graças a implementação do DMA, melhorando significativamente a experiência do usuário.

Para trabalhos futuros, há várias direções promissoras a serem exploradas. Primeiramente, implementar um circuito de filtragem que permita a leitura de frequências abaixo da quinta oitava, ampliando a gama de instrumentos compatíveis. Investigar a possibilidade de adicionar acordes ou o reconhecimento de múltiplas notas simultâneas, além de diminuir a latência desse reconhecimento, o que aumentaria a quantidade de botões disponíveis e permitiria a utilização do SoniCUBE em jogos mais complexos. Criar jogos específicos para o ensino de teoria musical e prática instrumental, utilizando o SoniCUBE como interface principal. Otimizar a interface do usuário para uma experiência mais intuitiva e amigável. E trabalhar na redução do consumo de energia e na miniaturização dos componentes para tornar o dispositivo ainda mais portátil e eficiente.

Por fim, o SoniCUBE apresenta potencial para promover inovações no campo da gamificação e da educação musical. Os resultados obtidos até o momento indicam que o sistema pode ser uma base sólida para futuras pesquisas e desenvolvimentos.

REFERÊNCIAS

Claypool, M. & Claypool, K. (2006). Latency and player actions in online games. *Commun. ACM*, 49(11):40–45.

GOGAMERS (2024). Pgb 2024 retrata a popularidade e a crescente importância dos jogos digitais na vida do brasileiro. Disponível em: https://gogamers.gg/o-gamer-no-brasil/pgb-2024/. Último acesso: 12 de junho de 2024.

Legaki, N.-Z., Xi, N., Hamari, J., Karpouzis, K., & Assimakopoulos, V. (2020). The effect of challenge-based gamification on learning: An experiment in the context of statistics education. *International Journal of Human-Computer Studies*, 144:102496. Disponível em: https://www.sciencedirect.com/science/article/pii/S1071581920300987. Último acesso: 20 de junho de 2024.

Long, M. & Gutwin, C. (2018). Characterizing and modeling the effects of local latency on game performance and experience. In *Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*, 285–297.

Melo, R., Mendes, D., Érico Marcelo Amaral, & Júnior, J. D. (2023). Gamificação na reabilitação física de amputados: Uma abordagem baseada em jogo sério. In *Anais Estendidos do XXII Simpósio Brasileiro de Jogos e Entretenimento Digital*, 1292–1302. Disponível em: https://sol.sbc.org.br/index.php/sbgames_estendido/article/view/27921. Último acesso: 19 de junho de 2024.

of Encyclopaedia Britannica, T. E. (2019). equal temperament. Disponível em: https://www.britannica.com/art/equal-temperament. Último acesso: 9 de agosto de 2024.

of Encyclopaedia Britannica, T. E. (2024). octave. Disponível em: https://www.britannica.com/art/octave-music. Último acesso: 9 de agosto de 2024.

of Inpired Acoustics, T. E. (2024). Midi note numbers and center frequencies. Disponível em: https://inspiredacoustics.com/en/MIDI_note_numbers_and_center_frequencies. Último acesso: 9 de agosto de 2024.

Ossada, J. C., Martins, L. E. B., Ranieri, B. S., & Belgamo, A. (2012). Gerse: Guia de elicitação de requisitos para sistemas embarcados. In *Workshop em Engenharia de Requisitos*. Disponível em: https://api.semanticscholar.org/CorpusID:41891542. Último acesso: 25 de junho de 2024.

Silva, J. A., da Silva, R. P. L., Vieira, V. C., & Isaltino, S. R. C. A. (2022). A contribuição da música para o desenvolvimento e aprendizagem na educação infantil. *Vox Metropolitana*, (6):18–30. Disponível em: https://revistavox.metropolitana.edu.br. Último acesso: 19 de junho de 2024.

Test, O. M. (2024). Controller tester. Disponível em: https://www.onlinemictest.com/controller-tester/. Último acesso: 28 de junho de 2024.

Tripette, J., Murakami, H., Ryan, K. R., Ohta, Y., & Miyachi, M. (2017). The contribution of nintendo wii fit series in the field of health: a systematic review and meta-analysis. *PeerJ*, 5(3600). Disponível em: `https://peerj.com/articles/3600/`. Último acesso: 20 de junho de 2024.

White, E. (2011). *Making Embedded Systems: Design Patterns for Great Software*. O'Reilly Media, 1st edition.