

SINFONIA DE DRIVES: MÚSICA COM ARDUINO, FLOPPY DRIVES E HDD
SYMPHONY OF DRIVES: MUSIC WITH ARDUINO, FLOPPY DRIVES, AND HDD

Fernando Parra Cano¹
Maylon Pires Macedo²
Rodrigo Diniz³

RESUMO: Este trabalho apresenta o desenvolvimento de um dispositivo musical experimental por meio da reutilização criativa de componentes eletrônicos obsoletos, como unidades de disquete, discos rígidos e alto-falantes, controlados por um microcontrolador Arduino UNO. O uso do Arduino possibilita projetos interativos de baixo custo, unindo tecnologias antigas a recursos modernos, incentivando o reaproveitamento de componentes e reduzindo o impacto ambiental. A proposta surge da preocupação com o descarte inadequado de lixo eletrônico e da necessidade de soluções sustentáveis que integrem tecnologia, arte e educação. Inspirado na cultura *Do It Yourself* (DIY) e na acessibilidade do Arduino, o projeto busca explorar novas formas de expressão musical por meio da manipulação de dispositivos eletromecânicos que, embora obsoletos, ainda possuem potencial funcional e criativo. A metodologia incluiu a separação de recursos físicos e digitais, montagem detalhada do sistema em *hardcase* e integração de *software* e *hardware* para controle via protocolo MIDI. Foram utilizados Arduino IDE, Java, Moppy Control GUI e Logic Pro X, que converteram arquivos MIDI em sinais compreendidos pelos componentes reaproveitados. As unidades de disquete atuaram como geradores de sons graves, os alto-falantes reproduziram melodias e harmonias, porém com limitação de polifonia, situação que foi contornada com a separação inteligente das vozes musicais entre os alto-falantes utilizados, e o HDD foi adaptado para sons percussivos. Uma fita de LED digital completou a experiência com efeitos visuais sincronizados. Os resultados demonstraram a viabilidade técnica e estética do projeto, evidenciando que é possível transformar sucata eletrônica em instrumentos musicais funcionais e criativos, aplicáveis em performances ou como ferramenta pedagógica em oficinas de música, tecnologia e sustentabilidade.

Palavras-chave: Áudio; Criatividade; Música experimental; Reutilização.

ABSTRACT: This work presents the development of an experimental musical device through the creative reuse of obsolete electronic components, such as floppy disk drives, hard disks, and speakers, controlled by an Arduino UNO microcontroller. The use of Arduino enables low-cost interactive projects, combining old technologies with modern resources, encouraging the reuse of components and reducing environmental impact. The proposal arises from the concern with the improper disposal of electronic waste and the need for sustainable solutions that integrate technology, art and education. Inspired by the *Do It Yourself* (DIY) culture and the accessibility of Arduino, the project seeks to explore new forms of musical expression through the manipulation of electromechanical devices that, although obsolete, still have functional and creative potential. The methodology included the separation of physical and digital resources, detailed assembly of the system in *hardcase* and integration of *software* and *hardware* for control via MIDI protocol. Arduino IDE, Java, Moppy Control GUI and Logic Pro X were used, which converted MIDI files into signals understood by the reused components. The floppy disk units acted as bass sound generators, the speakers

Análise e Desenvolvimento de Sistemas - Fatec Itapetininga - e-mail: fernando.cano@fatec.sp.gov.br
¹

Prof Dr. - Fatec Itapetininga - e-mail: maylon.macedo@fatec.sp.gov.br²

Prof Esp. - Fatec Itapetininga - e-mail: rodrigo.diniz@fatec.sp.gov.br³

reproduced melodies and harmonies, but with limited polyphony, a situation that was overcome with the intelligent separation of musical voices between the speakers used, and the HDD was adapted for percussive sounds. A digital LED strip completed the experience with synchronized visual effects. The results demonstrated the technical and aesthetic feasibility of the project, showing that it is possible to transform electronic scrap into functional and creative musical instruments, applicable in performances or as a pedagogical tool in music, technology and sustainability workshops.

Keywords: Audio; Creativity; Experimental music; Reuse.

1 INTRODUÇÃO

O descarte inadequado de equipamentos eletrônicos tem se tornado um problema ambiental crescente. Em 2019, foram geradas cerca de 53,6 milhões de toneladas de lixo eletrônico no mundo, com previsão de aumento para 74,7 milhões até 2030 (Forti *et al.*, 2020). Muitos desses materiais, como unidades de *Floppy Disk Drives* (FDD), *Hard Drive Disk* (HDDs) e placas eletrônicas, tornam-se obsoletos rapidamente, apesar do potencial de reutilização.

A popularização do Arduino e o crescimento da cultura DIY têm viabilizado projetos interativos de baixo custo, inclusive na música experimental. A junção de tecnologias obsoletas com microcontroladores modernos permite novas formas de expressão artística, aliando inovação à prática do reaproveitamento. Iniciativas que reutilizam componentes antigos reduzem o impacto ambiental e desenvolvem habilidades em eletrônica, programação e design, com aplicações em contextos educativos e artísticos.

Entre essas alternativas, destaca-se o uso de unidades de FDD e HDDs como fontes sonoras. Quando controlados digitalmente, esses dispositivos produzem sons mecânicos organizáveis musicalmente. Com o uso de softwares como o Moppy e ambientes compatíveis com MIDI, é possível transformá-los em instrumentos experimentais.

Este trabalho propõe o desenvolvimento de um dispositivo musical utilizando componentes obsoletos como FDD e alto-falantes, controlados por um Arduino UNO. A proposta busca comprovar a viabilidade técnica e artística dessa abordagem, incentivando a reflexão sobre sustentabilidade, criatividade e inovação na música e na educação.

2 METODOLOGIA

A pesquisa abordada neste projeto, caracterizada como desenvolvimento experimental, objetiva a reutilização de *hardware* antigo de computador para a criação de um dispositivo musical controlado por Arduino.

Os materiais utilizados para o desenvolvimento deste projeto estão listados na Tabela 1.

Tabela 1 – Recursos utilizados no projeto

ID	PRODUTO	QUANTIDADE
01	Arduino UNO	1
02	Unidade de disquete	3
03	Alto-falante de 2,5" 3w 8 ohms	3
04	HDD Maxtor STM380815AS	1
05	Carregador USB 5V 1.5A	1
06	Fonte 5V 3A	1
07	Hardcase 60cm X 33cm X 12 cm	1
08	Kit com 120 <i>jumpers</i> para Arduino	1
09	Fita com adesivo autocolante 20mm X 3m	1
10	Caixa protetora em acrílico para Arduino UNO	1
11	Fita de led digital e controlador com sensor de som	1
12	Perfil de alumínio de sobrepor de 50 cm e canto de 90 graus para fita de led	1
13	Filtro de linha de energia com 5 tomadas e cabo de 1 metro	1
14	Kit espaguete termo retrátil	1
15	Cabo USB tipo A para tipo B	1
16	Microfone de cristal piezoelétrico	1
17	Amplificador 120w RMS X 2 canais TPA3116 D2	1

18	Fonte 12V 3A	1
19	Conector fêmea P4	2
20	Cabo P2 estéreo X borne 3 vias	1

Fonte: Elaboração pelos autores (2025).

Os *softwares* utilizados no desenvolvimento deste projeto estão listados na tabela 2.

Tabela 2 – Softwares utilizados no projeto

ID	Software	Desenvolvedor
01	Moppy Control GUI 2.1.0	Sam Archer
02	Java Runtime Environment 8	Oracle
03	Arduino IDE 2.3.6	Arduino
04	Logic Pro X 10.7.9	Apple

Fonte: Elaboração própria (2025).

A metodologia adotada neste trabalho está detalhada em etapas, a seguir:

2.1 SEPARAÇÃO DOS RECURSOS NECESSÁRIOS E MONTAGEM DO PROJETO

A primeira etapa da metodologia consiste na seleção do *hardware*, acessórios e demais recursos necessários, bem como a montagem do projeto.

2.1.1 Separação dos Recursos Necessários

Inicialmente foram definidos os recursos necessários, tendo sido escolhido primeiramente o *hardware* a ser utilizado como fonte sonora para reprodução das músicas, neste caso, três unidades de FDD, conforme a Figura 1, para simular instrumentos musicais mais graves, como baixo elétrico, três alto-falantes para simular outros instrumentos do arranjo musical e um HDD para simular instrumentos de percussão.

Figura 1 – Unidade de disquete aberta

Fonte: Elaborado pelos autores (2025).

Em seguida, foi separada uma fonte de energia de 5V e 3A para os *drives* de disquete, tendo em vista que o consumo de energia de uma unidade de disquete pode variar entre 0,3 e 0,7 amperes.

Logo após, foi definido um amplificador de áudio a ser utilizado para amplificar o som da bobina de voz do HDD, tendo sido definido o modelo TPA3116 D2. Como fonte de energia do amplificador, foi definida uma fonte de 12V e 3A, considerando que o dispositivo opera entre 12V à 26V, para conexão à fonte, utilizou-se um conector do tipo fêmea P4 e para conexão auxiliar de áudio foi necessário um cabo P2 com borne 3 vias.

Após isso foi definido um Arduino UNO como microcontrolador a ser utilizado, tendo em vista que além de sua praticidade, também possui a quantidade de conexões necessárias para o projeto.

Em seguida, foi definida uma *hardcase* para montagem e acomodação do *hardware*.

Posteriormente, foi separada uma fita de led digital rítmica para obter efeitos de luz sincronizados com a música. Para alimentação do controlador da fita de led, foi separado um carregador USB 5V e 1.5A, e, para obter uma luz difusa da fita de led, utilizou-se um perfil de alumínio de sobrepor.

Também foram separados outros recursos como, filtro de linha para conexão das fontes de energia, *kit* com *jumpers* de Arduino, para conectar o *hardware* às portas adequadas do Arduino UNO, fita com adesivo autocolante para aderir as peças e acessórios à *hardcase*, caixa protetora em acrílico para Arduino UNO, *kit* de

espaguete termo retrátil para isolar emendas efetuadas em fios elétricos, cabo USB tipo A para tipo B para conexão do Arduino UNO ao computador e microfone de cristal piezoelétrico para possibilitar amplificação do som dos dispositivos de *hardware* para melhor usabilidade em performances ao vivo e gravações.

2.1.2 Montagem do Projeto

Primeiramente foram fixados recortes de fita com adesivo autocolante na região inferior das unidades de FDD, dos alto-falantes, do filtro de linha, da caixa protetora em acrílico para Arduino UNO, do amplificador de som, do perfil de alumínio de sobrepor e do microfone de cristal piezoelétrico.

Logo após, foram soldados fios finos nos terminais dos alto-falantes, conforme a figura 2.

Figura 2 – Alto-falante



Fonte: Elaborado pelos autores (2025).

Em seguida, a fita led foi instalada no perfil de alumínio de sobrepor, tendo sido utilizado 50 cm de fita, em seguida, o filtro de linha foi fixado no canto direito da *hardcase*, o perfil de alumínio com a fita de led, tendo sido fixado no canto superior juntamente ao controlador da fita de led, e, a fita de led foi conectada ao controlador, e o controlador foi conectado ao carregador USB, conectado ao filtro de linha.

Após isso, os *drives* de disquete, os alto-falantes e o amplificador de som foram fixados e acomodados na *hardcase*.

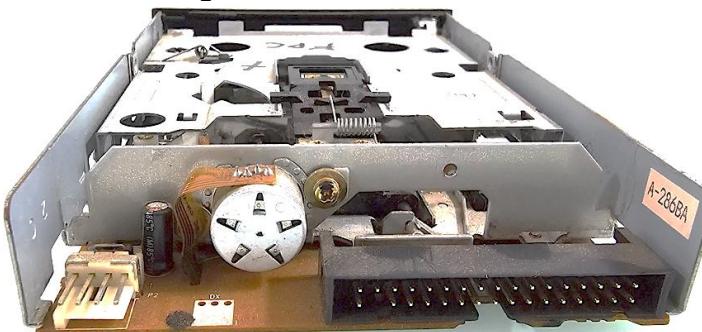
Na sequência, o Arduino foi instalado na caixa protetora e fixado na *hardcase*, ao lado esquerdo da primeira unidade de disquete.

Em seguida, foram conectados, em paralelo, três *jumpers* no terminal positivo de 5V de cada uma das três unidades de FDD, neste caso sendo o primeiro pino

dentre os quatro pinos de energia, e, três *jumpers* no terminal negativo de 5V de cada uma das três unidades de FDD, neste caso sendo o segundo pino dentre os quatro pinos de energia, após isso, foram conectados os cabos à fonte de 5V 3A, utilizando um conector fêmea P4, na devida polaridade. Para isolar os fios corretamente foi utilizado espaguete termo retrátil. Logo após, a fonte foi conectada ao filtro de linha e os cabos foram organizados na *hardcase*.

Após isso, em cada unidade de disquete foi conectado um *jumper* fêmea X fêmea nos pinos 11 e 12, fechando um circuito possibilitando a inicialização do *drive* quando receber energia. Na Figura 3, pode ser observado as conexões do FDD.

Figura 3 – Conexões do FDD



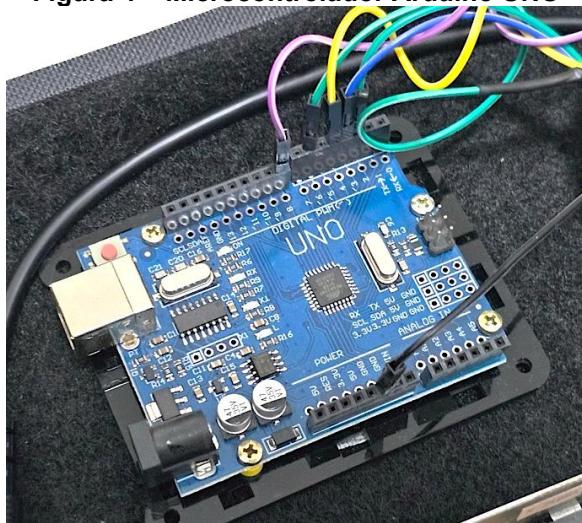
Fonte: Elaborado pelos autores (2025).

Posteriormente, para cada *drive* de disquete foi conectado um *jumper* fêmea no pino de direção do motor de passo, pino 18, os *jumpers* foram conectados em paralelo juntamente à um *jumper* de pontas macho X macho, os fios foram isolados utilizando espaguete termo retrátil e o *jumper* de ponta macho foi conectado à porta digital 3 do Arduino UNO.

Na sequência, para cada *drive* de disquete foi conectado um *jumper* fêmea no pino *step*, pino 20, os *jumpers* foram conectados em paralelo juntamente a um *jumper* de pontas macho X macho, os fios foram isolados utilizando espaguete termo retrátil e o *jumper* de ponta macho foi conectado à porta digital 2 do Arduino UNO.

Após isso, em uma das unidades de FDD foi conectado um *jumper* do tipo fêmea no pino 19 e foi conectado com os fios dos terminais negativos dos alto-falantes em paralelo juntamente a um *jumper* macho X macho, os fios foram isolados utilizando espaguete termo retrátil e o *jumper* de ponta macho foi conectado à porta GND, *ground*, do Arduino UNO, conforme a Figura 4.

Figura 4 – Microcontrolador Arduino UNO



Fonte: Elaborado pelos autores (2025).

Logo após, para cada um dos três alto-falantes, em cada um dos polos positivos foi conectado um *jumper* macho X macho à cada fio soldado anteriormente e os fios foram isolados com espuaguete termo retrátil, e, cada *jumper* de ponta macho foi conectada às portas pares digitais do Arduino, sendo a primeira conectada à porta 4, a segunda à porta 6 e a terceira à porta 8.

Em seguida, no conector de energia do amplificador de som, foi instalado um conector P4 fêmea conforme a polaridade indicada. Após isso, o cabo do conector auxiliar de áudio foi conectado ao borne de 3 vias do cabo P2 estéreo X borne 3 vias, seguindo a polaridade indicada. Depois disso, a fonte de 12V 3A foi conectada ao filtro de energia e ao cabo P4 do amplificador de som.

Na sequência, os parafusos de tipo torx do HDD foram retirados, e na sequência a tampa foi removida. Em seguida, foram soldados dois fios nos terminais do atuador do HDD. Em seguida, o HDD foi fixado na *hardcase* ao lado direito do terceiro *drive* de disquete e os fios dos terminais do atuador foram conectados à saída esquerda de alto-falante do amplificador de som, como pode ser visto na Figura 5.

Figura 5 – HDD fixado na hardcase



Fonte: Elaborado pelos autores (2025)

Por fim, foi fixado o microfone de cristal piezoelétrico na *hardcase*.

2.2 SOFTWARES UTILIZADOS

Inicialmente, foi efetuado o *download* e a instalação do *software* Arduino IDE 2.3.6. Após aberto o programa, foi instalada a biblioteca TimerOne, tendo sido selecionados os submenus *sketch*, incluir biblioteca e gerenciar bibliotecas.

Logo após, foi efetuado o *download* do arquivo compactado em formato zip contendo o *software* MoppyControlGUI-2.1.0 e também foi feito o *download* do arquivo zip que contém o código do Moppy para Arduino, Moppy2Microcontroller-2.1.1, no *website* GitHub.

Após isso, o arquivo Moppy2Microcontroller-2.1.1.zip foi descompactado e o arquivo Moppy2-Arduino.ino foi aberto no *software* Arduino IDE 2.3.6. Em seguida, o Arduino UNO foi conectado ao computador e no *software* Arduino IDE no menu ferramentas, foi selecionado o tipo de placa de Arduino, neste caso, Arduino UNO e foi selecionada a porta serial correspondente ao Arduino conectado. Posteriormente, o código do Moppy foi enviado ao Arduino por meio da opção ‘enviar usando programador’, como pode ser observado na figura 6.

Figura 6 – Código do Moppy2 para Arduino

```

1  /*****
2  */
3  /* | \_ | */
4  /* | \ / | */
5  /* | [V] | / \ [V] | \ [V] | */
6  /* | [ ] | [ ] | [ ] | [ ] | */
7  /* | [ ] | [ ] | [ ] | [ ] | */
8  /* | [ ] | [ ] | [ ] | [ ] | */
9  /* | [ ] | [ ] | [ ] | [ ] | */
10 /* | [ ] | [ ] | [ ] | [ ] | */
11 /* | [ ] | [ ] | [ ] | [ ] | */
12 /*
13  * This file left intentionally blank to appease the Arduino IDE.
14  *
15  * FOR GENERAL CONFIGURATION OPTIONS SEE MoppyConfig.h
16  *
17  * See src/main.cpp for the top level code. Functional
18  * code files can be found in the src folder in the sketch directory, but
19  * these should be opened using a text editor or different IDE as
20  * the Arduino IDE does weird things if you start trying to use
21 */

```

Fonte: Elaborado pelos autores (2025).

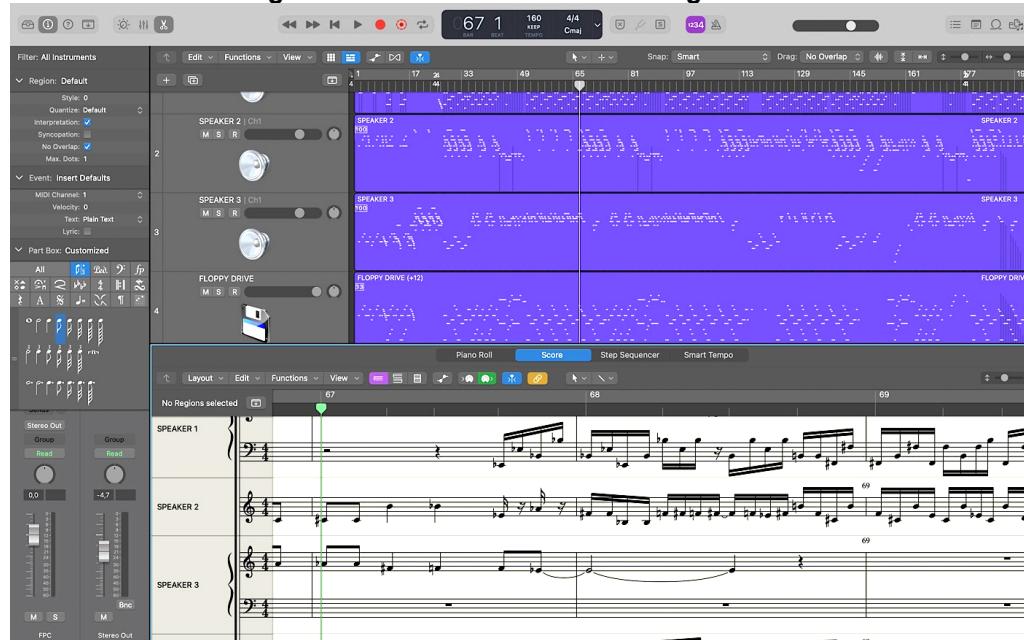
Em seguida, foi efetuado o *download* e instalação do *software Java Runtime Environment*, pré-requisito do MoppyControlGUI-2.1.0.

Na sequência, foi preparado e ajustado o arranjo musical de uma música no *software de produção musical Logic Pro X 10.7.9*. No *software*, foi aberto um novo projeto e importado o arquivo MIDI de uma música, arquivo que armazena informações sobre notas musicais, ritmos, e outros dados de instrumentos virtuais contidos no arranjo. Após isso, foram criadas quatro novas *tracks* de instrumento virtual, foram selecionados como instrumentos o *plugin ‘External Instrument’*. Em seguida, foi selecionada a opção ‘*Logic Pro Virtual Out*’ como destino de saída de MIDI para cada uma das quatro *tracks* no *plugin ‘External Instrument’*, também foram selecionados os canais MIDI para cada uma das quatro *tracks*, sendo o canal 1 para a *track* a ser utilizada para os *drives* de disquete e os canais 2, 3 e 4, para os alto-falantes, tendo em vista a sequência em que os estes dispositivos foram conectados ao Arduino UNO.

Na sequência, para ajustar o arranjo musical para ser reproduzido corretamente, primeiramente foi ajustado o baixo elétrico entre a região das notas musicais C0 e B2, 16.35 Hz - 246.941635 Hz, mantendo a mesma tonalidade musical,

apenas ajustando dentre as oitavas 0 e 2, caso necessário. Após isso, foram ajustados os instrumentos musicais harmônicos e melódicos para as *tracks* dos alto-falantes, tendo em vista que o Moppy envia as notas musicais apenas de maneira monofônica, não tendo suporte à polifonia, dessa forma, os acordes musicais de instrumentos harmônicos foram separados entre os alto-falantes, priorizando também os instrumentos musicais mais importantes do arranjo, considerando a possibilidade de combinações dentre os 3 alto-falantes, e, mantendo as notas musicais do arranjo entre a região das notas C2 e B3, utilizando transposição de oitava musical, caso necessário, conforme a Figura 7.

Figura 7 – Interface do software Logic Pro X



Fonte: Elaborado pelos autores (2025).

Em seguida, foi adicionada uma *track* de instrumento virtual para o HDD, com o plugin ‘Sforzando’ e selecionada uma bateria eletrônica 8 bits.

3 REFERENCIAL TEÓRICO

De acordo com Mcroberts (2015), Arduino é uma plataforma baseada em microcontroladores que facilita o uso de eletrônica em projetos diversos, sendo muito utilizada por iniciantes, educadores e profissionais. Conta com um *hardware* de código aberto que permite conectar sensores, motores, luzes, entre outros acessórios, e

possui um *software* (Arduino IDE) para programar seu funcionamento, ela permite a criação de sistemas interativos. O Arduino se destaca como uma ferramenta acessível e robusta para explorar a eletrônica e a programação.

Segundo Ismailov *et al.* (2022), os microcontroladores Arduino são ferramentas versáteis amplamente utilizadas devido à sua facilidade de uso e flexibilidade. Suas principais aplicações incluem a educação, onde auxiliam no aprendizado prático de programação e eletrônica; a prototipagem rápida de projetos eletrônicos; e a integração com sensores para coleta e processamento de dados ambientais. Também são muito usados em automação residencial, permitindo o controle remoto de luzes, eletrodomésticos e sistemas de segurança, além de projetos de robótica, onde comandam motores e servomecanismos. Artistas os utilizam em instalações interativas, unindo tecnologia e criatividade. Com o avanço da *Internet of Things* (IoT), o Arduino tem sido fundamental na conexão de dispositivos à Internet, e é amplamente empregado em pesquisas científicas para soluções customizadas de coleta e análise de dados.

Conforme Archer (2019), uma unidade de disquete, no inglês, *floppy disk drive*, é um dispositivo que lê e grava dados em disquetes, mídias magnéticas flexíveis com trilhas circulares de informação. Seu funcionamento depende de um motor de passo que move o cabeçote de leitura e gravação entre as trilhas para acessar os dados. Esses *drives* podem ser reutilizados como dispositivos musicais, já que a movimentação do cabeçote em diferentes frequências gera vibrações sonoras semelhantes às de um alto-falante. Com o uso de um Arduino e um aplicativo em Java, é possível controlar essas vibrações para produzir notas musicais reconhecíveis, demonstrando o potencial criativo de reaproveitar tecnologias antigas.

De acordo com Cook (2015), o MIDI é um padrão criado em 1983 para padronizar a comunicação entre instrumentos musicais eletrônicos e computadores, transmitindo dados musicais em vez de áudio. Sua simplicidade, versatilidade e fácil implementação o tornaram amplamente adotado na música, automação e robótica. Mesmo com novas tecnologias, o MIDI permanece essencial na produção musical. O Arduino, por sua vez, pode enviar e receber sinais MIDI, sendo uma ferramenta acessível e prática para projetos musicais interativos.

Segundo Archer (2019), o *Moppy Control GUI* é um *software* que permite transformar unidades de FDD e outros dispositivos em instrumentos musicais,

controlando seus componentes mecânicos de forma precisa. Ele utiliza o protocolo MIDI para receber informações musicais e traduzi-las em comandos comprehensíveis pelos *drives*, fazendo com que eles emitam sons ao movimentar suas cabeças de leitura. Essa integração permite que instrumentos MIDI tradicionais e *softwares* de produção musical controlem os *drives* de disquete.

De acordo com Hobson (2015), usar um HDD como alto-falante é uma maneira criativa de reaproveitar componentes internos, especialmente a bobina de voz e o braço atuador.

Conforme Blount (2001), dentro de um HDD, o braço atuador, responsável por mover a cabeça de leitura e gravação sobre os discos, é controlado por um mecanismo chamado bobina de voz, semelhante ao encontrado em alto-falantes. Quando um sinal elétrico passa por essa bobina, cria-se um campo magnético que move o braço, e esse movimento pode vibrar o ar ao redor, gerando ondas sonoras, funcionando de maneira parecida com o diafragma de um alto-falante.

4 RESULTADOS E DISCUSSÃO

A construção do dispositivo musical experimental, reutilizando unidades de FDD, alto-falantes e um HDD, resultou em um sistema funcional capaz de executar arranjos musicais por meio de sinais MIDI enviados do *software* Logic Pro X ao Arduino UNO, que, por sua vez, controla os dispositivos eletromecânicos conforme o protocolo definido pelo Moppy Control GUI.

Os disquetes demonstraram desempenho satisfatório na reprodução de notas musicais graves, simulando com eficácia timbres similares ao de um baixo eletrônico. Esse comportamento corrobora a afirmação de Archer (2019), segundo a qual as vibrações geradas pela movimentação dos cabeçotes podem ser exploradas como elementos sonoros. O controle dos motores de passo através do Arduino permitiu variações tonais distintas conforme a frequência aplicada, revelando-se eficaz para fins musicais.

Os alto-falantes de 2,5 polegadas utilizados foram capazes de emitir sons simples, funcionando como canais adicionais para representar instrumentos harmônicos e melódicos. Apesar de sua limitação física e de potência, sua integração com o Arduino validou a viabilidade de seu uso em arranjos que priorizem simplicidade e efeitos experimentais. A limitação de polifonia foi contornada com a separação

inteligente das vozes musicais entre os três dispositivos, conforme o modelo de organização proposto na metodologia.

O uso do HDD como componente de percussão, embora mais desafiador, demonstrou potencial expressivo. A amplificação do atuador do HDD com o TPA3116 D2 resultou em uma resposta sonora satisfatória, especialmente em ritmos mais marcados, reforçando os apontamentos de Hobson (2015) e Blount (2001) sobre o uso de bobinas de voz em contextos criativos como este projeto.

A sincronização com a fita de LED rítmica ampliou o aspecto performático do dispositivo musical, associando visualmente os impulsos musicais com estímulos luminosos, o que pode ser interpretado como um reforço sensorial que potencializa a experiência estética, uma abordagem alinhada às práticas interativas descritas por Ismailov *et al.* (2022).

Assim, os resultados obtidos indicam que a proposta metodológica foi coerente em sua essência, promovendo a reutilização criativa de componentes obsoletos e demonstrando o potencial do Arduino como plataforma de experimentação musical. A possibilidade de personalização, baixo custo e a integração com softwares profissionais de produção musical revelam um caminho promissor para artistas, educadores e pesquisadores interessados em explorar intersecções entre música, tecnologia e sustentabilidade.

Os resultados obtidos do projeto em funcionamento podem ser observados nos vídeos disponíveis em <https://drive.google.com/drive/folders/1JRWR14e7rd-N97QZEdEj-9J3SWIBxL9B?usp=sharing>

Na Figura 8 pode ser observada uma visão geral do projeto elaborado.

Figura 8 – Visão geral do projeto elaborado



Fonte: Elaborado pelos autores (2025).

5 CONSIDERAÇÕES FINAIS

O desenvolvimento do dispositivo musical com componentes eletrônicos obsoletos, como FDD, HDD e alto-falantes, mostrou-se uma solução viável para o reaproveitamento tecnológico, promovendo sustentabilidade, criatividade e inovação. Usando Arduino e softwares como o Moppy, foi possível transformar resíduos eletrônicos em fontes sonoras funcionais, ampliando a expressão musical e incentivando o consumo consciente de tecnologia. O projeto também revelou seu potencial educativo e artístico, ao estimular habilidades em eletrônica, programação e produção musical, oferecendo uma alternativa acessível à criação sonora contemporânea.

REFERÊNCIAS BIBLIOGRÁFICAS

ARCHER, Sam. **How to Create Floppy Drive Music.** 2019. Disponível em: <https://makezine.com/projects/how-to-create-floppy-drive-music/>. Acesso em: 19 abr. 2025.

BLOUNT, Walker C. Noise reduction using dampening in voice coil motors/Actuators of Hard Disk Drives. **IBM Storage Technology**,(Oct. 2001), p. 1-6, 2001. Disponível em: http://johnjohn1.free.fr/Informatique/DisqueDur/IBM_Hitachi/IBM_noise.pdf. Acesso em: 19 abr. 2025.

COOK, Mike. **Arduino music and audio projects**. Apress, 2015.

FORTI, Vanessa *et al.* The global e-waste monitor 2020. **United Nations University (UNU), International Telecommunication Union (ITU) & International Solid Waste Association (ISWA)**, Bonn/Geneva/Rotterdam, v. 120, 2020. Disponível em: https://np-mag.ru/wp-content/uploads/2020/08/The-Global-E-waste-Monitor-2020-Quantities-flows-and-the-circular-economy-potential_compressed.pdf. Acesso em: 19 abr. 2025.

HOBSON, James. **Hard Drive... Speakers?** 2015. Disponível em: <https://hackaday.com/2015/03/03/hard-drive-speakers/>. Acesso em: 19 abr. 2025.

ISMAILOV, Alisher Shakirovich *et al.* Study of arduino microcontroller board. **Science and Education**, v. 3, n. 3, p. 172-179, 2022. Disponível em: https://www.researchgate.net/profile/Alisher-Ismailov/publication/359502443_Study_of_arduino_microcontroller_board/links/62402fca8068956f3c50ea36/Studyoarduinomicrocontrollerboard.pdf. Acesso em: 19 abr. 2025.

MCROBERTS, Michael. **Arduino Básico-2ª edição: Tudo sobre o popular microcontrolador Arduino**. Novatec Editora, 2015. Disponível em: <https://s3.novatec.com.br/capitulos/capitulo-9788575224045.pdf>. Acesso em: 19 abr. 2025.