

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
Campus DIVINÓPOLIS
GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

Luiz Matanta de Andrade

PROJETO E CONSTRUÇÃO DE PROTÓTIPO DE UM PEDAL DE
EXPRESSÃO MIDI

Divinópolis
2019

Luiz Matanta de Andrade

PROJETO E CONSTRUÇÃO DE PROTÓTIPO DE UM PEDAL DE
EXPRESSÃO MIDI

Monografia de Trabalho de Conclusão de Curso
apresentado à comissão avaliadora do curso de
Graduação em Engenharia Mecatrônica como
parte dos requisitos exigidos para a obtenção do
título de Engenheiro Mecatrônico .
Áreas de integração: Mecânica, Eletrônica e Com-
putação.

Orientador: Prof. Dr. Christian Gonçalves
Herrera

Divinópolis
2019

Luiz Matanta de Andrade

PROJETO E CONSTRUÇÃO DE PROTÓTIPO DE UM PEDAL DE
EXPRESSÃO MIDI

Monografia de Trabalho de Conclusão de Curso
apresentado à comissão avaliadora do curso de
Graduação em Engenharia Mecatrônica como
parte dos requisitos exigidos para a obtenção do
título de Engenheiro Mecatrônico .
Áreas de integração: Mecânica, Eletrônica e Com-
putação.

Comissão Avaliadora:

Prof. Dr. Christian Gonçalves Herrera
CEFET/MG *Campus V*

Prof. Dr. João Carlos de Oliveira
CEFET/MG *Campus V*

Prof. Mestre Nestor Dias de Oliveira Volpini
CEFET/MG *Campus V*

Divinópolis
2019

Resumo

Este trabalho consiste no projeto de um pedal de expressão musical. Trata-se de um pedal similar a um acelerador onde o ângulo entre a plataforma pisada e o solo é medido através de um sensor, cujo sinal é enviado ao microprocessador, que gera mensagens para serem interpretadas por softwares instalados em um computador pessoal. Esse trabalho tem o intuito de mostrar o princípio de funcionamento, as características construtivas e as etapas desenvolvidas para o projeto e construção do pedal. Como atualmente no mercado é escassa a produção de pedais de expressão musical deste tipo, busca-se então projetar um sistema que seja capaz de atender vários músicos. Assim, esse projeto engloba três áreas de grande importância na Engenharia Mecatrônica: Mecânica, Eletrônica e Computação. A metodologia de projeto aplicada consiste em dividir o trabalho em três etapas: Projeto Comportamental, Projeto Lógico e Projeto Físico. Espera-se que ao final do projeto, tenha-se um pedal de expressão capaz de controlar parâmetro dentro de um software que use o padrão VST.

Palavras-chave: Pedal de expressão musical, protocolo MIDI.

Abstract

This work consists in the design and construction of a prototype of a musical expression pedal. This is an accelerator-like pedal where the angle between the stepped platform and the ground is measured through a sensor whose signals are sent to the microprocessor, that generates messages to be interpreted by the software installed on a personal computer. This work aims to show the working principle, the constructive characteristics and the steps to develop the pedal design and construction. This kind of musical expression pedal is rare in the market, so we seek designing a system that could provide several musicians. Thus, this project will encompass three areas of great importance in Mechatronic Engineering: Mechanics, Electronics and Computing. The applied project methodology consists of dividing the work into three stages: Behavioral Design, Logical Design and Physical Design. It is expected that at the end of the project, there will be an expression pedal capable of controlling parameters with a software using the VST standard.

Keywords: Musical expression pedal, MIDI protocol.

Sumário

Lista de Figuras	ix
Lista de Acrônimos e Notação	x
1 Introdução	1
1.1 Definição do Problema	2
1.2 Motivação	2
1.3 Objetivo Geral	2
1.4 Objetivos Específicos	3
1.5 Estado da Arte	3
1.6 Organização do Documento	4
2 Revisão da Literatura	7
3 Fundamentação Teórica	11
3.1 Sensores	11
3.1.1 Potenciômetro Linear e Rotativo	11
3.1.2 Encoder Linear e Rotativo	13
3.1.3 Sistema ótico (LDR)	15
3.2 Microprocessador	16
3.3 Ferramenta de programação do microprocessador	19
3.4 Ferramenta CAD	19
3.5 Protocolo de Comunicação MIDI	19
3.5.1 Protocolo de Comunicação USB	22
3.5.2 Softwares de Áudio	23
3.5.3 Firmware	24
3.5.4 MIDIOX	25
4 Metodologia	26
4.1 Projeto Comportamental	26
4.1.1 Especificações	27
4.1.2 Execução	27
4.1.3 Arquitetura do sistema	29
4.2 Projeto Lógico	35
4.2.1 Arquitetura da plataforma	36
4.2.2 Projeto de Software	38

4.3	Projeto Físico	42
4.3.1	Projeto de Hardware	42
4.4	Validação do Código	44
5	Resultados	48
6	Considerações Finais	50
6.1	Conclusões	50
7	Anexos	51
	Referências	55

Lista de Figuras

1.1	Korg EXP-2 Pedal de Expressão para Teclados e Pedaleiras	5
1.2	Pedal Controlador De Expressao M-audio Ex-p	5
1.3	Diagrama de blocos do projeto	6
2.1	Esquema de funcionamento de um órgão hidráulico de Vitruvius	8
2.2	Reconstituição moderna do órgão hidráulico de Vitruvius.	8
3.1	Potenciômetro com suas posições e simbologia	12
3.2	Potenciômetro ligado a uma saída Analógica do Kit	13
3.3	Exemplo de encoder linear e rotativo	13
3.4	Esquemático do sensor rotativo com fototransistor	14
3.5	Foto de encoder linear	14
3.6	Resistor LDR	15
3.7	Gráfico da resposta do LDR	16
3.8	Estrutura interna de um Microprocessador	17
3.9	Hardware utilizado para equipamento	20
3.10	Cabo de ligação das portas midi e possíveis ligações	20
3.11	Exemplo típico das mensagens Midi	21
3.12	Arquitetura VST	24
4.1	Fluxograma do pedal	28
4.2	Diagrama de blocos do sistema	29
4.3	Apresentação dos módulos referentes ao arquiteto	30
4.4	Apresentação dos módulos referentes ao utilizador	31
4.5	Apresentação dos módulos referentes ao utilizador	32
4.6	Controlador Midi	37
4.7	Dependências do Mecanismo	38
4.8	Lógica do sensor	40
4.9	Relação Mensagens MIDI com o ângulo	41
4.10	Relação Mensagens MIDI com o Volume	42
4.11	Ligação do circuito	43
4.12	Validação da Comunicação	44
4.13	Validação Código	45
4.14	Pedal de expressão visto de baixo, com o arduino instalado e o sensor	46
4.15	Pedal de expressão	46
4.16	Régua Dentada	47

4.17	Potenciômetro com adaptador para régua dentada	47
4.18	Pedal de expressão	47
5.1	Interface Guitar Rig	48
5.2	Parte Física do Pedal	49
7.1	Dimensões	51
7.2	Dimensões	52
7.3	Vista Lateral do Pedal	52
7.4	Vista Lateral com a régua dentada para mover o potenciômetro	53
7.5	Vista superior da parte externa do pedal	53
7.6	Vista Superior da parte interna do pedal	54

Lista de Acrônimos e Notação

MIDI	Musical Instrument Digital Interface
USB	Universal Serial Bus
LAN	Local Area Network
MMA	MIDI Manufacturers Association
JMSC	Japan MIDI Standards Committee
IMUG	International MIDI Users Group
IMA	International MIDI Association
ALU	Unidade Lógica e Aritmética
UC	Unidade de Controle
CI	Circuito Integrado
SPI	Interface Serial Periférica
VST	Virtual Studio Technology
VSTi	Virtual Studio Technology Instruments

Introdução

O uso de computadores para produzir música vem de muitos anos. Com o avanço da tecnologia dos computadores a música e o áudio têm sido ricamente favorecidos.

Em um contexto extenso, a música no computador pode estar ligada à gravação e edição do áudio em computadores, ao uso de sistemas informatizados para produção de música, a variados tipos de instrumentos musicais eletrônicos e a automação de equipamentos de áudio e instrumentos musicais que estão ligados diretamente a criação artística interligada ao computador.

A utilização de instrumentos musicais mecânicos como o órgão hidráulico, onde se utilizava água para regular a pressão do ar, tem a sua origem desde antes Cristo. A idealização de colocar máquinas para tocar é tão antiga quanto a música (MACLEAN, 1905).

Com o surgimento da interface MIDI (Musical Instrument Digital Interface na sua tradução literal Interface Digital de um Instrumento Musical), que é um protocolo específico para detalhar sequências sonoras polifônicas digitalizadas e para conectar instrumentos musicais eletrônicos a computadores digitais, todos os sintetizadores, samplers¹, baterias e outros instrumentos eletrônicos de todas as marcas começaram a se conectar com os computadores e entre si.

Além de instrumentos digitais, alguns dispositivos periféricos são utilizados como um controle para enviar mensagens MIDI. Esse acontecimento beneficiou vários gêneros musicais que utilizam programas com MIDI. Portanto, programar instrumentos usando sequenciadores, ou seja, tocando sons de samplers se tornou uma prática comum (HUBER, 2007).

O pedal de expressão MIDI proposto trata-se de um dispositivo que transmite mensagens de dados para o computador através da posição do deslocamento em que se encontra. Em cada posição do sensor, ele envia um dado diferente, que será interpretado dentro de

¹Sampler um equipamento que consegue armazenar eletronicamente sons (samples) em uma memória e reproduzi-los posteriormente, um a um ou de forma conjunta.

um software como uma função específica (HAMMER, 1986).

Uma das vantagens do computador pessoal ter um sistema MIDI é que nele é possível ter várias aplicações diferentes dependendo do software que é carregado. Alguns exemplos de software que trabalham com MIDI são: Pro Tools, Pure Data, Guitar Pro, Reason, Logic Pro, entre outros.

Esse trabalho apresenta um projeto e construção de um protótipo de pedal de expressão MIDI. O dispositivo MIDI apresentado neste trabalho é um dispositivo que não emite sons, ele apenas gera comandos para que outro dispositivo traduza suas mensagens e as transforme em áudio.

1.1 Definição do Problema

O mercado musical está em frequente expansão. Atualmente se mantém entre um dos mais rentáveis na área cultural. De acordo com a Associação Nacional da Indústria da Música, que levantou dados junto a Receita Federal no ano de 2018, o setor movimentou dois bilhões de reais e 42 por cento desse valor foi destinado aos acessórios músicas e baterias (PAUL, 2019).

A tecnologia tem sido uma das principais ferramentas para a exploração da capacidade musical em diversas vertentes, seja de maneira facilitadora ou exploratória. Em questão de facilidade e alternativas para combinações de novos sons, um dos grandes problemas é a dificuldade de encontrar, no mercado, pedais de expressão MIDI/USB que não são acoplados a pedaleiras. Uma solução nesse sentido se torna muito conveniente aos músicos e produtores musicais que utilizam o computador pessoal para compor, gravar e editar suas peças.

1.2 Motivação

Como a evolução da tecnologia afeta tudo ao nosso redor, a música não seria uma exceção à regra. À medida que o tempo passa, novas tecnologias vem sendo descobertas e aplicadas. A motivação desse trabalho é evidenciar como a Engenharia Mecatrônica possui incontáveis aplicações, sendo uma delas a música. A música é algo que esteve presente durante toda minha vida. Sendo assim, este projeto inclui duas áreas de grande interesse pessoal.

1.3 Objetivo Geral

Projetar e construir um protótipo de um pedal de expressão microprocessado com comunicação USB sobre o protocolo MIDI, que se conecte a todos os software que usam

padrão VST.

1.4 Objetivos Específicos

- Projetar e construir o sistema mecânico de um pedal de expressão, baseado nos pedais já existentes no mercado. Ele sofrerá modificações que vão se adequar as necessidades do projeto.
- Implementar um sistema que envie mensagens midi para o computador através da porta USB e fazer com que todo computador o reconheça como um instrumento midi.
- Implementar técnicas para medição de ângulo entre as duas plataformas, em que o sensor ficará encarregado de ler o ângulo. Em seguida será enviado a leitura correspondente para o microprocessador que enviará uma mensagem midi para aquela posição de ângulo ao software.
- Projetar o sistema com o sensor escolhido.
- Projetar o circuito eletrônico e sua programação utilizando um Arduino.
- Projetar um sistema em que o pedal de expressão funcione com exatidão nos comandos de controle dentro do software.

1.5 Estado da Arte

Na literatura pesquisada foram escassas as referências técnicas e acadêmicas sobre projetos de pedal de expressão MIDI. Sendo assim, foi realizada uma análise de mercado sobre pedais de expressão mais utilizados pelos consumidores. A pesquisa foi feita com produtos nacionais e internacionais.

Nas empresas pesquisadas não foi encontrada nenhuma que fabrique pedais de expressão MIDI como instrumento individual, como o projeto proposto. Todas elas são produtos em que os pedais que enviam dados MIDI são acoplados a pedaleiras, e os que são individuais apenas controlam a amplitude do sinal de áudio, normalmente vindo de uma guitarra ou teclado eletrônico, que são dispositivos eletrônicos utilizados pelos músicos para processar os sinais de áudio dos instrumentos.

As Figuras 1.1 e 1.2 mostram exemplos de pedais vendidos no mercado da marca Korg e da M-audio, que possuem as características citadas acima. Esses pedais possuem uma grande variação de preço compreendida entre os valores de trezentos reais a até mil reais.

O diagrama da Figura 1.3 representa todo o sistema para o qual se relaciona esse trabalho, sendo que será executado o projeto do pedal e sua maneira de conexão com o computador. Podemos ver que esse pedal contém um sensor que capta dinamicamente cada posição em que se encontra. Essa posição é enviada para o MP (microprocessador), onde é transformada em mensagens do tipo MIDI. Cada posição resultante do movimento realizado pelo pé, através do pedal, gera um ângulo entre o pedal e o solo, ainda a ser definido, que representará diferentes valores MIDI para o computador.

No computador, o pedal de expressão é conectado através de uma entrada USB (Universal Serial Bus), que é um tipo de tecnologia que permite a conexão de periféricos sem a necessidade de desligar o computador. Em seguida deve-se fazer uma tradução do reconhecimento no computador, para que este entenda como um dispositivo MIDI.

A parte em que envolve o instrumento musical não será executada no trabalho, mas mostra que se conectar um instrumento musical na entrada ADC (Conversor Analógico Digital) será reconhecida pelo software com padrão VST, o mesmo que reconhecerá o pedal de expressão.

Dentro desse software o pedal tem a capacidade de mudar alguns parâmetros de como será enviado o sinal de som para caixa, parâmetros como timbre, controle de volume, sustain² e outros recursos que o software proporciona. A seguir passará para uma saída DAC (Conversor Digital Analógico) que está ligada ao amplificador onde se obtém o resultado final do áudio gerado.

Deve ser lembrado que os efeitos de processamento sonoro, técnica de sintetização e envelopes são deixadas para os componentes receptores. Isto significa que uma mesma mensagem enviada para dois sintetizadores (instrumento) diferentes pode causar sons diferentes. A qualidade do som está ligado diretamente a interface de áudio, que pode ser chamada popularmente de placa de som, cabeçote ou processador.

1.6 Organização do Documento

Este trabalho foi organizado em 7 capítulos. No presente capítulo foram apresentados uma introdução, a definição do problema, a motivação do trabalho, o objetivo geral, os objetivos específicos, estado da arte e a organização do documento.

No segundo capítulo encontra-se a revisão da literatura.

No terceiro capítulo foi apresentado a fundamentação teórica.

No quarto capítulo encontra-se a metodologia utilizada no projeto.

No quinto capítulo foram apresentados os resultados obtidos no projeto.

No sexto capítulo se encontra as considerações finais.

²Sustentação da nota, tempo que ela dura.



Figura 1.1: Korg EXP-2 Pedal de Expressão para Teclados e Pedaleiras
(Disponível em: <http://www.korg.com.br>)



Figura 1.2: Pedal Controlador De Expressao M-audio Ex-p
(Disponível em: <https://www.m-audio.com>)

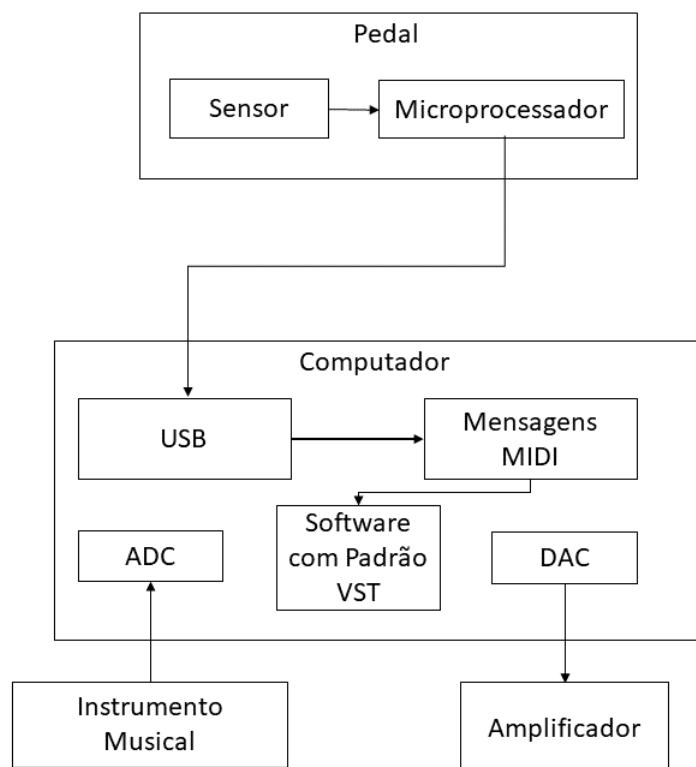


Figura 1.3: Diagrama de blocos do projeto
Fonte:(Matanta,2019)

Revisão da Literatura

O primeiro instrumento onde se encontrava um pedal de expressão era o órgão, que é um instrumento musical no qual o som é produzido pela passagem do ar comprimido através dos tubos de metal e madeira. Este é considerado um dos instrumentos mais antigos de toda música ocidental e o primeiro dos instrumentos de teclas (BENNETT, 1986).

Os órgãos são popularmente conhecidos por sua utilização em grandes igrejas e sua tradição na liturgia cristã (BENNETT, 1986). A história do órgão se inicia no século III antes de Cristo, com a criação do *hydraulos* ou órgão hidráulico pelo grego Ctesíbio de Alexandria. O invento do primeiro órgão veio a partir de uma flauta grega que é o pai dos órgãos. O *hydraulos* foi muito utilizado durante vários séculos em festas, teatros e nos circos, até surgir o órgão pneumático no século IX, movido por foles manuais (JENKINS, 2009).

De acordo com (JENKINS, 2009) a autorização do uso dos órgãos nas igrejas só foi concretizada em 1565, pelo Concílio de Milão e era um dos poucos instrumentos permitidos durante as celebrações religiosas. Tal realidade se estendeu por vários séculos, aspecto que levou o órgão a ter uma grande relevância e um papel de notória importância para a sociedade da época.

Os órgãos eletrônicos apareceram nos anos 70, como uma evolução dos sintetizadores (BENNETT, 1986). As Figuras 2.1 e 2.2 mostram o primeiro órgão criado, que possui bombas de ar para auxiliar nos seus mecanismos hidráulicos, onde para fazer o som era necessário pressionar o pedal para bombear a água gerando pressão, essa movimentação de ar era capaz de fazer com que o instrumento gerasse som.

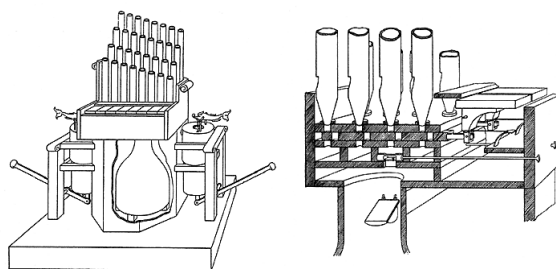


Figura 2.1: Esquema de funcionamento de um órgão hidráulico de Vitruvius (VITRUV, ZEHN BUBER ARCHITEKTUR, DARMSTADT, 1964) Disponível em www.argolis.de/Museen



Figura 2.2: Reconstituição moderna do órgão hidráulico de Vitruvius. Disponível em www.argolis.de/Museen

Definida a origem dos primeiros pedais de expressão abordaremos um pouco da história do MIDI.

Midi, é o protocolo que especifica como os instrumentos musicais podem ser interconectados e controlados remotamente.

O protocolo MIDI, que formaliza tanto a interface do hardware quanto uma linguagem de comunicação de mensagens com significados musicais, iniciou seu desenvolvimento na década de 1980 em resposta ao aumento da sofisticação e correspondente complexidade dos instrumentos eletrônicos, especialmente os sintetizadores (HAMMER, 1986).

Segundo (HAMMER, 1986) o comércio dos sintetizadores eletrônicos já existia antes do desenvolvimento do protocolo MIDI. Os primeiros só eram acessíveis ao grande público através de exposições especiais e da utilização esporádica em trilhas sonoras de filmes. Esses primeiros dispositivos eram analógicos e, antes do advento do transistor, eram enormes, caros e de difícil operação. Robert Moog construiu sintetizadores analógicos consideravelmente menores e mais baratos, direcionados a um grande mercado em potencial. No final da década de 1960 os sintetizadores Moog eram cada vez mais utilizados na música popular.

No entanto, os primeiros sintetizadores eletrônicos não foram pensados para serem

interconectados. Naturalmente, a interface de controle utilizada nesses sintetizadores eram também analógicos. O sistema envolvia duas portas de controles. Uma para controlar a frequência (altura) e duração das notas musicais. Outra no sentido de prover uma referência de tempo. A primeira porta consistia de dois sinais: um controlador de variação de tensão (V), proporcional a nota juntamente com um dispositivo disparador de pulso. A segunda porta consistia de um sincronizador de pulso externo.

No final da década de 1970, iniciou-se o desenvolvimento dos instrumentos eletrônicos analógicos e digitais. Com o barateamento no custo dos microprocessadores e uma produção em massa de circuitos integrados, os fabricantes de sintetizadores começaram a incorporar circuitos digitais em seus instrumentos. Muitos componentes eram menores, mais estáveis e mais baratos quando construídos a partir de circuitos digitais. Os timbres análogos, que exigiam um emaranhado de cordas físicas, puderam ser substituídos por painéis de controles menores e softwares especiais. Já no final desta década muitos sintetizadores eram inteiramente digitais.

Porém, no início da década de 1980, os engenheiros da Sequential Circuits, Roland Corporation e Oberheim Electronics começaram a integração de instrumentos de qualquer fabricante. Este padrão se tornou o MIDI 1.0, formalmente publicado em agosto de 1983.

O desenvolvimento do padrão MIDI não se deu diretamente por nenhum acidente, a utilização de microprocessadores como controladores de sintetizadores estava no início, os fabricantes se deram conta de que esta modificação do analógico para o digital oferecia uma chance de aprender com as experiências do passado. A indústria do computador já havia solucionado vários problemas relacionados com o endereçamento de mensagens na interconexão de computadores de rede.

Os engenheiros que iniciaram o trabalho em MIDI perceberam que poderiam adotar a estrutura do protocolo LAN (Local Area Network), que especificava tanto as exigências das placas (hardware) para a conectividade, quanto o formato das mensagens trocadas por essas placas. Essa abordagem que criou uma camada de abstração entre o trabalho interno dos instrumentos digitais e a linguagem utilizada na comunicação com o mundo externo, permanecendo com uma característica central do protocolo MIDI e de todas as suas melhorias e extensões, de acordo com HAMMER (1986).

O padrão MIDI é um produto de fabricantes comerciais e o controle sobre esse padrão é mantido por duas organizações: MIDI Manufacturers Association (MMA) e sua contraparte japonesa, Japan MIDI Standards Committee (JMISC). Qualquer intenção de mudança deve ser aprovada pelas duas organizações. Existem outros grupos voltados para os clientes que iniciou como IMUG- International MIDI Users Group e, posteriormente transformou-se em IMA International MIDI Association (HAMMER, 1986).

Quando o padrão MIDI foi criado, não foram definidas somente as características

elétricas da transmissão de informações entre equipamentos, como também foram estabelecidos os códigos digitais para diversas mensagens de comando a serem transferidas de um instrumento para o outro. Como, por exemplo, a execução de notas.¹

Dentre essas mensagens, há uma categoria que engloba todos os tipos de controles, como ajuste de volume e de pan², posição do pedal de sustain, ajuste de tempo do portamento, de intensidade de efeitos, etc. Essas mensagens são chamadas de control change mensagens ou mensagens de alteração de controle.¹

Os comandos de control change são divididos em dois subgrupos: controles contínuos (cujo valor pode variar gradualmente) e controles liga/desliga (que atuam como chaves de dois estados). Todas as mensagens de control change contêm identificação de canal, para que possam ser dirigidas ao instrumento desejado, que está configurado para receber no mesmo canal em que a mensagem está sendo transmitida.¹

Quando o músico move algum dos dispositivos de controle de seu instrumento (pedal de volume, alavanca de modulation, etc), são transmitidas mensagens correspondentes de control change, que contêm três códigos, compondo o seguinte formato: control change, número do controle e valor do controle.¹

O pedal de expressão é um dispositivo de ação contínua (gradual) e sua atuação específica (o parâmetro que altera) pode ser programada no instrumento. É um pedal similar a um acelerador, e é usado para dosar a intensidade de atuação. O movimento do pedal gera mensagens MIDI de control change, que transmitem a sua posição física, identificada por valores de 0 (pedal totalmente para trás) a 127 (pedal totalmente para a frente).¹

O pedal de expressão MIDI não tem função específica, e ao ser recebido via MIDI pode ser direcionado para atuar em um dentre diversos parâmetros. Ele é usado como comando de Expression (expressão), que é uma forma de acentuação volume principal (HAMMER, 1986).

O MIDI tem uma grande importância no meio tecnológico, onde mudou a maneira de fazer música no mundo. Ele está presente em todos os estúdios e praticamente em todas as casas de quem produz música com tecnologia.

¹ <https://www.midi.org/>

² Abreviado do termo Panoramic no inglês, este é semelhante ao controle de Balance encontrado em muitos sistemas de som domésticos

Fundamentação Teórica

Para o desenvolvimento de um pedal de expressão MIDI é preciso entender algumas funções separadamente, para conseguir alcançar o objetivo de ter um protótipo que seja capaz de manipular, analogicamente, parâmetros dentro de um software de computador gerando várias funções em termos musicais. Os principais aspectos dessa arquitetura devem ser: o sensor utilizado, o microprocessador, a ferramenta de programação do microprocessador, que no caso em questão é o Arduino¹ UNO, a ferramenta CAD para desenho, o protocolo de comunicação MIDI, o protocolo de comunicação USB e softwares de áudio.

3.1 Sensores

Para esse projeto foi necessário o uso de sensores onde se pode fazer a leitura exata em qual posição, em ângulo, se encontra o pedal. Nessa parte do capítulo se encontram alguns possíveis sensores a serem utilizados no projeto, que possuem características diferentes, mas o mesmo objetivo de expressar o ângulo medido.

Todos os sensores vistos aqui, são sensores analógicos que respondem através de sinais analógicos. Ou seja, sinais que mesmo limitados em uma certa faixa, podem variar entre inúmeros valores de tensão intermediários.(WENDLING, 2010)

Todos os sensores citados atendem o projeto, mas somente na parte de execução haverá uma análise mais refinada, onde o objetivo será apresentar qual mais se adequa ao projeto.

3.1.1 Potenciômetro Linear e Rotativo

A Figura 3.1 mostra um potenciômetro que quando se roda o cursor para um dos lados ele aumenta sua resistência e para o lado inverso diminui. Temos também os 3 terminais

¹Arduino é uma plataforma de prototipagem eletrônica de hardware livre e de placa única, projetada com um microcontrolador Atmel AVR

em que as extremidades recebem 0V de um lado e 5V do outro e o terminal central ligado a alguma entrada analógica.

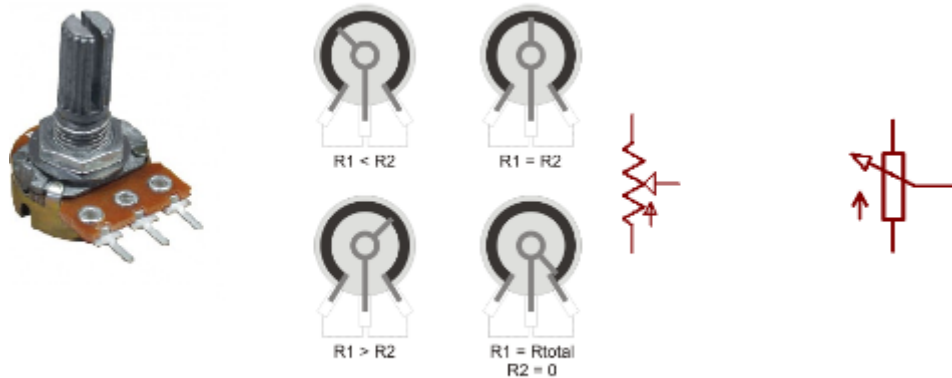


Figura 3.1: Potenciômetro com suas posições e simbologia
 Fonte: LUÍS, F. P., Funcionamento e Utilização de Sensores, 2006

Embora o potenciômetro tenha maior aplicabilidade em ajustar parâmetros (como por exemplo; volume de caixas amplificadas) podemos utilizá-lo como sensor de posição. No contexto de ser utilizado como sensor de posição, encontram-se como exemplos robôs e equipamentos industriais, para informar a angulação exata de um braço mecânico.

No caso desse projeto, será utilizado para definir o deslocamento da parte superior do pedal. Portanto, através desse componente é possível que a mudança da variável mecânica como um ângulo seja transformada em uma mudança de característica elétrica.

O potenciômetro é definido como um divisor de tensão variável. Ele é composto por uma faixa de material resistivo (geralmente grafite) ligada entre dois terminais externos. Neste material desliza um cursor, ligado diretamente ao terminal central do potenciômetro. Esse cursor pode ser movimentado através de um eixo rotativo ou um pino de plástico ou metal, quando alternamos a posição do cursor, alteramos a resistência entre o terminal central e os dois terminais externos do potenciômetro (LUÍS, 2006).

Segundo a referência pesquisada (LUÍS, 2006) para utilizar um potenciômetro como um sensor de posição é necessário ligar um dos terminais da extremidade com uma tensão positiva e outro com uma tensão negativa. A tensão do terminal central variará entre esses dois valores de tensão. A leitura da posição do potenciômetro pode ser feita através de uma porta analógica, pois o mesmo já é um divisor de tensão e as entradas já são bufferizadas. Os terminais terão suas extremidades alimentadas com 5V e 0V que se encontram no potenciômetro. Lembrando que o terminal deve ser ligado somente na entrada analógica, no caso desse projeto, na entrada analógica do Arduino, como na Figura 3.2.

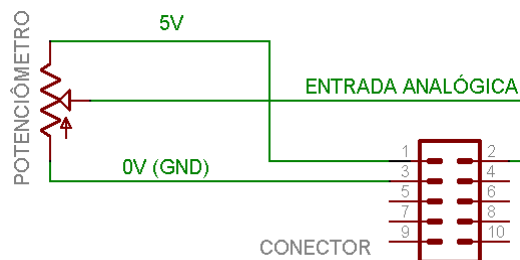


Figura 3.2: Potenciômetro ligado a uma saída Analógica do Kit
 Fonte: LUÍS, F. P., Funcionamento e Utilização de Sensores, 2006

3.1.2 Encoder Linear e Rotativo

Os encoders são sensores que são utilizados para determinar: posição, sentido de giro de um eixo de motor, velocidade, entre outras funcionalidades. Ele gera informações precisas em suas aplicações. A Figura 3.3 define o entendimento simples de um encoder.

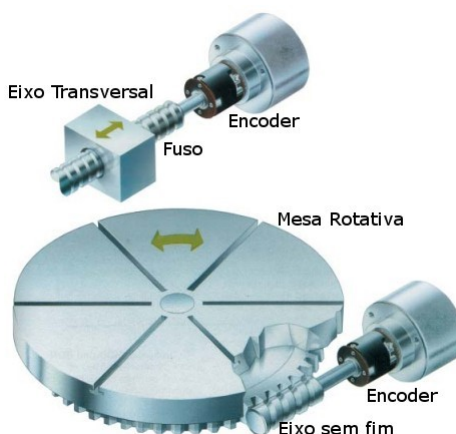


Figura 3.3: Exemplo de encoder linear e rotativo
 Fonte: LUÍS, F. P., Funcionamento e Utilização de Sensores, 2006

A Figura 3.3 mostra o encoder, que vem acoplado com um eixo, em que na ponta tem-se um eixo sem fim. Esse eixo sem fim tem o objetivo de movimentar para esquerda ou para direita a mesa rotativa, indicando a posição exata. O eixo transversal tem a mesma característica do eixo sem fim, porém a medida que o cursor movimentar o eixo se move para frente ou para trás no fuso. Esses dois tipos de encoders na figura se enquadram no termo rotativo, pois têm que girar para mudar sua posição.

Os encoders rotativos são usados para medir o movimento rotacional de um eixo. A Figura 3.4 mostra os componentes fundamentais de um encoder rotativo, que se utiliza de um diodo emissor de luz (LED), um disco que pode ser construído por vários tipos de materiais como vidro, metal ou plástico e um detector de luminosidade que é colocado do lado oposto do disco.

A Figura 3.4 mostra os componentes necessários no encoder rotativo que seria um diodo

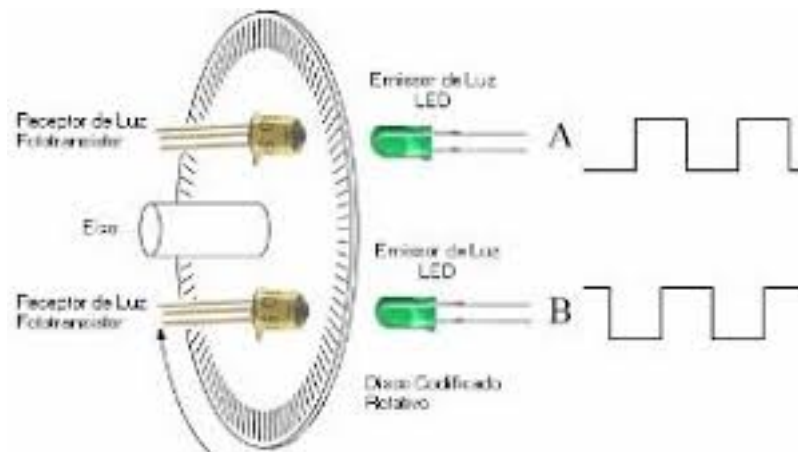


Figura 3.4: Esquemático do sensor rotativo com fototransistor

Fonte: <https://ballast.com.br/encoders-uso-e-aplicacoes/>

emissor de luz (LED), um disco e um detector de luz (fototransistor) no lado contrário ao disco. O disco se encontra em um eixo rotatório e possui várias perfurações regulares que são repetidas continuamente formando segmentos opacos e transparentes codificados no disco (STEMMER; UFSC, 2016).

De acordo com (LUÍS, 2006) os encoders do tipo linear medem a posição de um objeto ao longo de um comprimento e os rotativos identificam a posição através do movimento de rotação do eixo. Os encoders rotativos são usados para medir o movimento rotacional de um eixo.

O encoder linear, consiste em uma escala (fita codificada) e um sensor que lê os espaçamentos entre as escalas da fita. Tem uma resolução que é medida em pulsos por distância. Exemplo: pode-se dizer que uma régua que apresenta 80 pulsos por milímetros, possui uma resolução de 80 pulsos de contagem das marcas pelo sensor a cada 1 milímetro linear na fita do encoder.



Figura 3.5: Foto de encoder linear

Fonte: (<https://www.exportersindia.com>)

A Figura 3.5 mostra um encoder linear onde ela percorre esse cursor dando a posição de onde ela se encontra. É como se fosse uma régua e a posição em que ela está parada acima é um valor diferente dentro dessa régua.

3.1.3 Sistema ótico (LDR)

Esse sensor LDR (do inglês Light Dependent Resistor), ou resistor depende de luz. O LDR é um sensor cuja a resistência varia conforme a incidência de luz sobre o seu corpo, conforme exemplificado na Figura 3.6.



Figura 3.6: Resistor LDR
Fonte: (<http://4.bp.blogspot.com>)

O LDR não possui polaridade, é possível ligar seus terminais de variadas formas. Em sua composição temos sulfeto de Cádmio (CdS), um material semicondutor, que é disposto num traçado ondulado na superfície do componente. Essa composição tem a capacidade de reduzir sua resistência a uma passagem de corrente elétrica quando a luminosidade sobre ele aumenta, em um ambiente escuro, sua resistência será alta. Quando aumenta gradativamente a intensidade da luz que incide sobre ele, podemos verificar que a resistência cairá. Sabendo que esses valores são influenciados pelo uso de componentes utilizados, a quantidade de luz no ambiente e o próprio multímetro. A Figura 3.7 mostra a resposta do LDR (LUÍS, 2006).

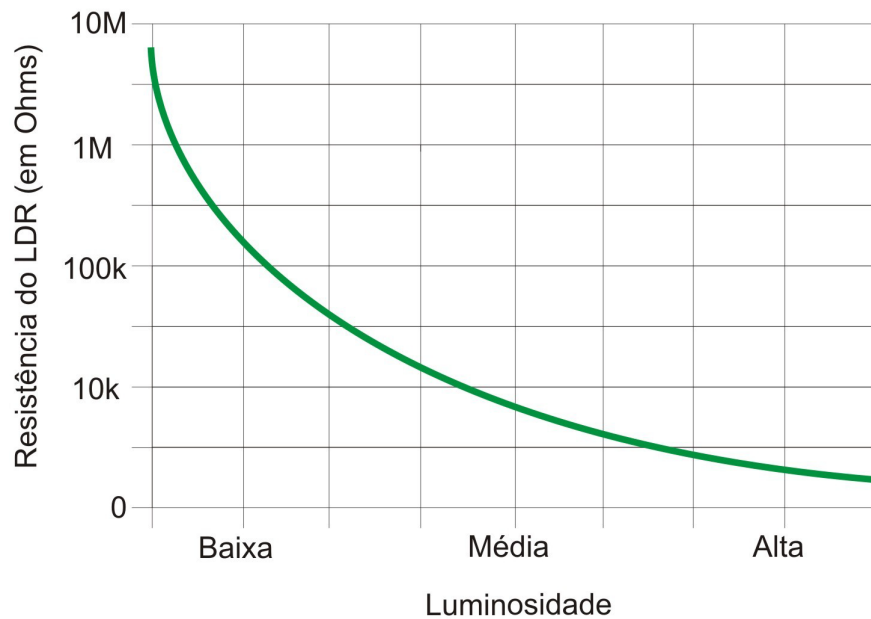


Figura 3.7: Gráfico da resposta do LDR
Fonte: LUÍS, F. P., Funcionamento e Utilização de Sensores, 2006

Para fazer a medida da luminosidade do ambiente através do LDR, é necessário fazer com que a variação da resistência do componente seja convertida em uma variação de tensão. Essa tensão é monitorada através do microcontrolador (Arduino), através de suas entradas analógicas. Para fazer a linearização, nesse caso, seria feita um código no Arduino onde mapearia só os pontos linearizados.

3.2 Microprocessador

O microprocessador é um dispositivo lógico programável em um único chip de silício. Ele age sob o controle de um programa armazenado em memória, executando operações lógicas, aritméticas, tomada de decisão, além de entrada e saída, permitindo a comunicação com outros dispositivos periféricos (SKLAR, 1988). No dia a dia, temos inúmeras aplicações de microprocessadores como em relógios digitais, calculadoras, automóveis, micro-ondas, máquinas de lavar entre outros.

Apesar dos diversos tipos de fabricantes e famílias de microprocessadores, pode-se identificar alguns aspectos comuns na sua arquitetura.

Tipicamente lê separadamente cada instrução de um programa guardado na sua memória, obtém os seus operandos quando for necessário, manipula os dados de acordo com especificado no código e lê dados de dispositivo de entrada ou envia dados para dispositivos de saída.

Um microprocessador é a parte principal de um microcomputador e sua função principal é executar instruções. Ele possui duas unidades básicas que são a Unidade Lógica

Aritmética (ULA), responsável pelas operações lógicas e aritméticas, e a Unidade de Controle (UC) que serve para decodificação e execução das instruções, fornecendo sinas de temporização necessárias para todas as partes do processador além de registradores para armazenamento da informação binária, que são instruções, endereços e dados. A Figura 3.8 podemos ver mais detalhadamente o processo(TANENBAUM, 2016).

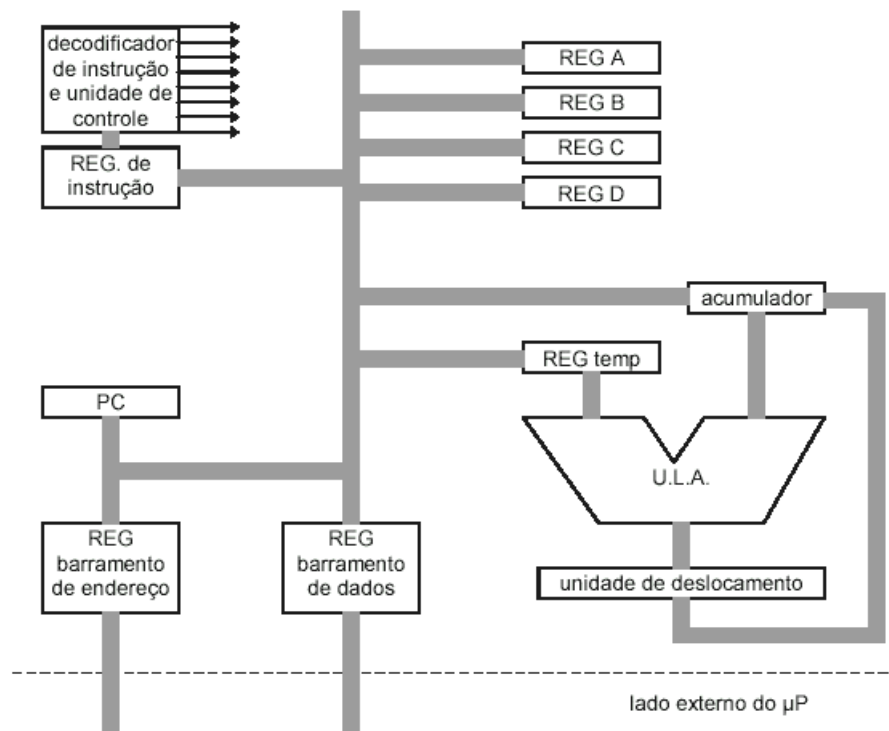


Figura 3.8: Estrutura interna de um Microprocessador
 Fonte: <http://iris.sel.eesc.usp.br/sel433a/Micros.pdf> , pág 8

A Figura 3.8 mostra a parte do microprocessador onde se realiza todas as operações lógicas e aritméticas onde cada bloco tem um significado e importância.

Registradores corresponde a uma memória local rápida do microprocessador, destinada ao armazenamento de dados e instruções. O registrador pode ser de propósito geral, para as operações de movimentação de dados e operações lógicas e aritméticas, ou especiais que são funções específicas para determinados fins(TANENBAUM, 2016).

Dentre esses propósitos especiais temos o Acumulador que é o principal registrador dentro de um processador que participa das operações de entrada e saída de dados. O Registrador de Flags que armazenam os indicadores de estado do processador, que seria um bit cada estado, onde serve para detectar ocorrência de estouro em uma operação ou a ocorrência de um estado nulo. Contador do Programa é um registrador que armazena o endereço da memória do início da próxima instrução a ser executada. E por último dentro dos propósitos especiais o Ponteiro da Pilha que armazena o endereço da última posição ocupada da pilha, ou seja, é a parte para armazenamento de dados temporário Unidade

Lógica e Aritmética (ALU), implementa as operações lógicas e aritméticas dependendo do microprocessador, em geral, o resultado de uma operação é armazenado no acumulador.

Unidade de Controle (UC), todas as funções do microprocessador são controladas pela UC, ela é retirada cada instrução de memória, interpretando-a fornecendo os sinais de controle necessários à sua execução.

Circuitos de temporização (Gerador de Clock), implementam o funcionamento síncrono do processador indicando onde cada etapa da execução deve ocorrer.

Controle e Decodificação (Memória de Microprogramas), memória apenas de leitura onde lê as atividades internas para serem realizadas a cada instrução.

Decodificador de Instrução, recebe as instruções armazenadas na memória e gera códigos do Microprograma que realizara a tarefa definida.

Para a escolha do microprocessador desse projeto teve que se levar em conta alguns parâmetros técnicos como, velocidade de processamento em tempo real, consumo, custo do projeto, confiança no fornecedor e experiência anterior ao qual se chegou no microcontroladores ATMEL ATMEGA328, que é o componente principal da placa do Arduino Uno utilizado nesse trabalho.

O dispositivo de 8 bits da família AVR com arquitetura RISC avançada e com encapsulamento DIP28. Ele conta com 32kb de Flash, 2 KB de RAM e 1 KB de EEPROM. Pode operar em até 20MHz, entretanto a placa de Arduino opera em 16MHz. Esse microcontrolador pode operar com tensões bem baixas, de até 1,8 V, mas nessa tensão apenas opera 4MHz.

Possui dois modos de consumo super baixos, o Power-down Mode² e o Power-save Mode³, para que o sistema possa poupar energia em situações de espera. Possui, como periféricos uma USART que funciona a até 250kbps, uma SPI, que vai a até 5MHz, e uma I2C que pode operar até 400kHz.

Conta com um comparador analógico interno ao CI e diversos timers, além de 6 PWMs. A corrente máxima por pino é de 40mA, mas a soma da corrente de todo o CI não pode ultrapassar 200mA. Ele possui um oscilador interno de 32kHz que pode ser utilizado, por exemplo, em situações de baixo consumo.⁴

²É um modo de economia de energia no dispositivo, onde se reduz a tensão de uso

³modo de economia de energia no dispositivo

⁴<https://store.arduino.cc/usa/arduino-uno-rev3>, acessado em 09 de junho de 2019

3.3 Ferramenta de programação do microprocessador

Como já foi descrito anteriormente o microprocessador utilizado está acoplado no Arduino que é uma plataforma eletrônica open-source⁵ baseada em softwares e hardwares de fácil implementação. Possui uma extensa quantidade de bibliotecas prontas para diversas aplicações o que facilita a criação de novos projetos. Para sua codificação o Arduino atua em um ambiente desenvolvimento integrado que se baseia no processamento do projeto, que tem suporte para linguagens de programação C, C++ e Java (ARDUINO, 2015).

3.4 Ferramenta CAD

Foi utilizado um software do tipo CAD desenvolvimento, especificamente o Solidworks, que é uma ferramenta que se utiliza nos campos da arquitetura, design de interiores, engenharia mecânica, engenharia geográfica e em vários outros ramos EARLE (1992).

A escolha específica pelo Solidworks se deu por algumas vantagens que ele proporciona, como compatibilidade, onde o programa é universal e tem adaptável a outros softwares. Pela sua alta performance nas ferramentas gráficas, pode desenhar, modelar e sua qualidade gráfica que gera em seus desenhos 2D e 3D, ao qual se adequará a esse projeto.

3.5 Protocolo de Comunicação MIDI

Podemos pensar na tecnologia MIDI como uma partitura para os controladores MIDI e computadores, ou seja, é através desta tecnologia que as mensagens são enviadas de forma a reproduzir os sons configurados nos botões ou potenciômetros (KARPF, 2003) (KOGAN, 2008).

Atualmente, todos os equipamentos musicais eletrônicos utilizam o protocolo MIDI. Este protocolo foi aceito pelos músicos e compositores devido ao seu hardware, Figura 3.9, de baixo custo e também a capacidade de comunicação como já dito.

Através dos cabos MIDI e das ligações DIN de 5 pinos, Figura 3.9, é possível fazer a comunicação com o software do protocolo, com uma velocidade de transmissão de 31250 bits por segundo (bps) (HUBER, 2007).

Os controladores MIDI possuem uma porta de entrada e saída, midi in e midi out, USB e a porta Thru, o que permite fazer a ligação em cadeia entre vários equipamentos midi, Figura 3.10.

⁵Open source é um termo em inglês que significa código aberto. Isso diz respeito ao código-fonte de um software, que pode ser adaptado para diferentes fins.

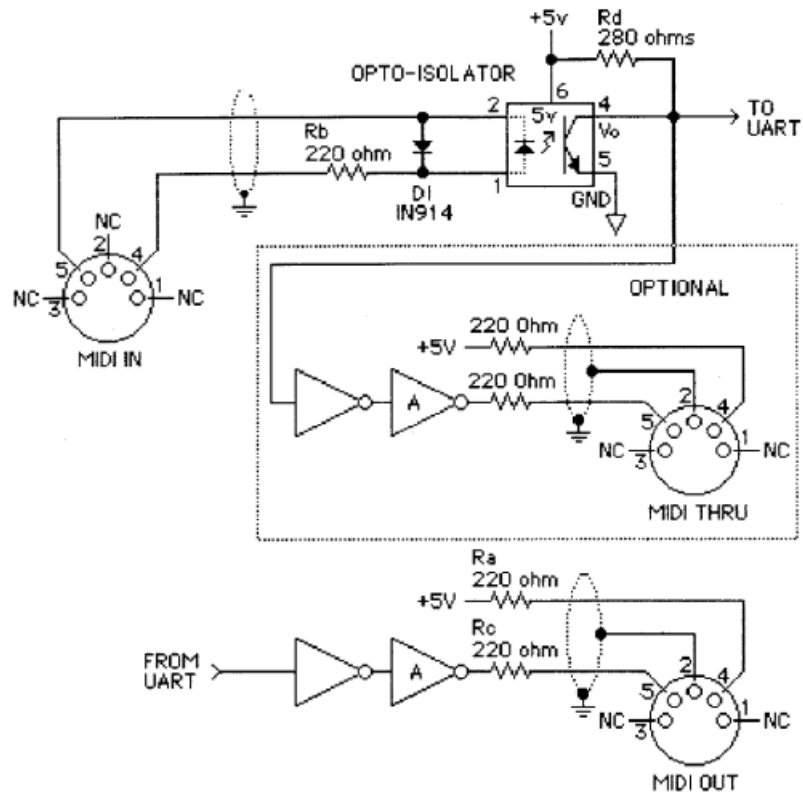


Figura 3.9: Hardware utilizado para equipamento Fonte MIDI. (MMA, 1995)

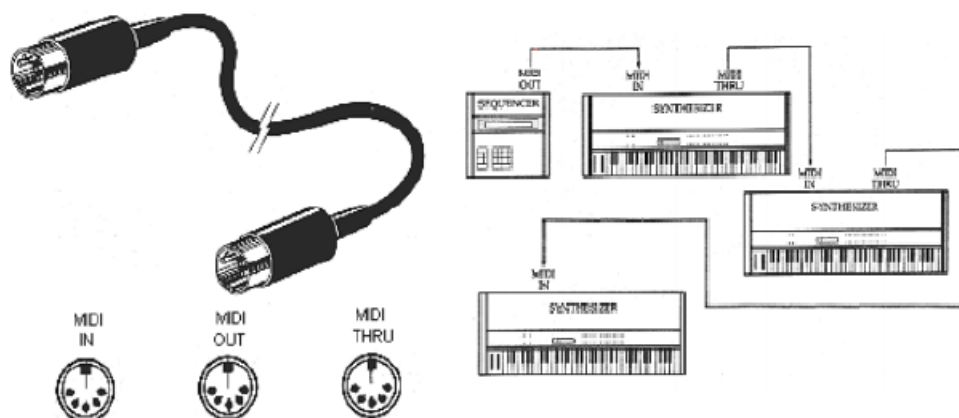


Figura 3.10: Cabo de ligação das portas midi e possíveis ligações Fonte:(KOGAN,2008)

A mensagem transmitida tem sempre o seu tempo de processamento, esta tem que ser recebida, processada e ser enviada para o módulo de som. Então nesse equipamento temos 2 tipos de mensagens, ativar nota e desativar nota (KOGAN, 2008).

A grande vantagem dos equipamentos que contém a tecnologia MIDI é que conseguem simular qualquer som de qualquer instrumento musical, e para o fazer basta apenas um controlador MIDI, sintetizador ou sequenciador (KARPF, 2003). A Figura 3.11 representa os tipos de mensagens enviadas e recebidas ao ser pressionado ou acionado um botão de pressão ou um potenciômetro.

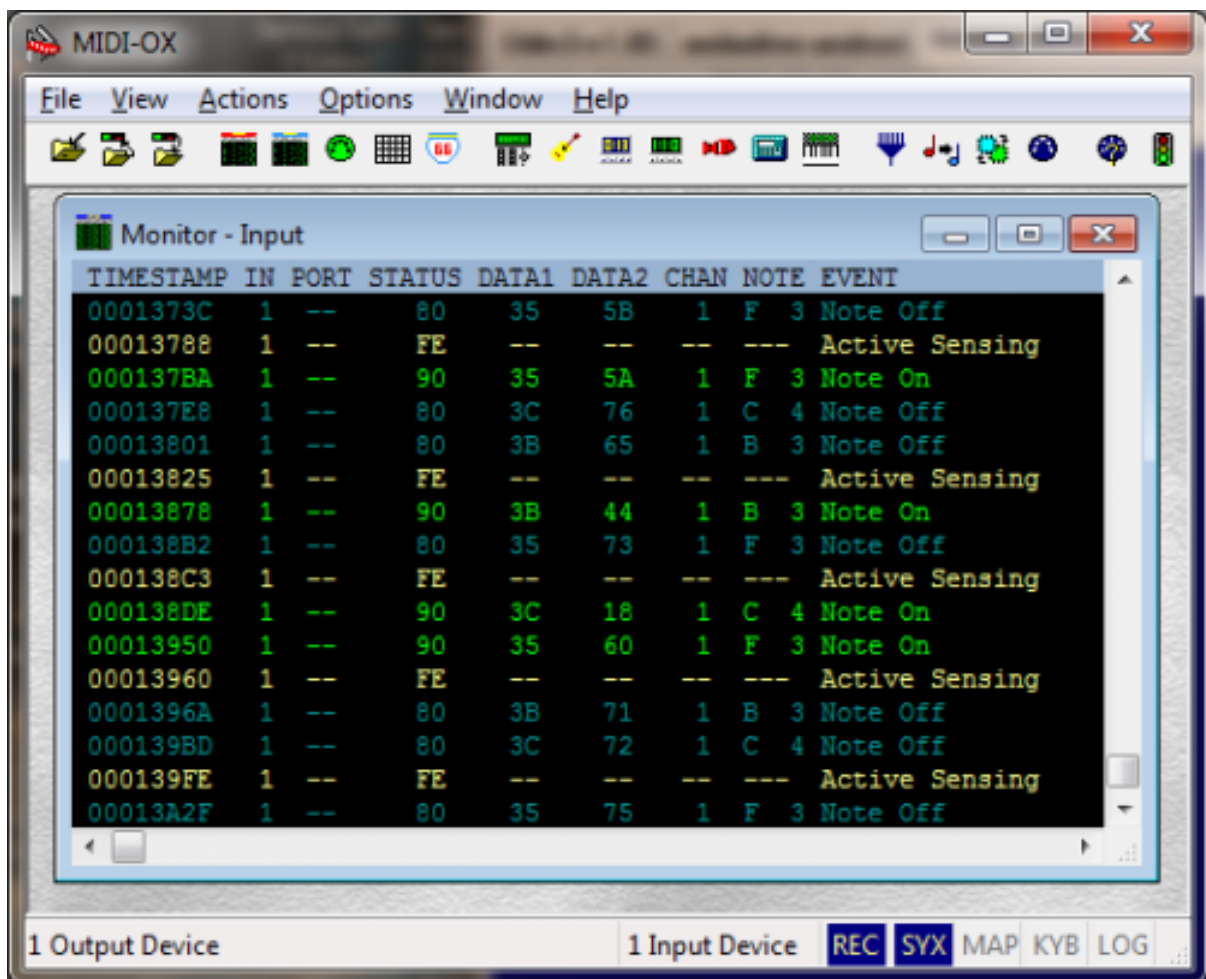


Figura 3.11: Exemplo típico das mensagens Midi

Fonte:(Arvydas, 11 de Julho de 2013. Disponível em: <arvydas.co.uk>)

A Figura 3.11 mostra mensagens que dependem do canal MIDI. São as mensagens que, quando enviadas, afetam apenas o canal MIDI selecionado.

NOTE ON é o comando para tocar uma determinada nota, com uma determinada velocidade, onde velocidade no caso específico significa intensidade, é um evento como podemos ver na tabela.

NOTE OFF comando para desligar a nota que está sendo tocada. Também há dentro

do comando a velocidade de desligamento da nota, mas poucos teclados a usam, e são mais raros os sequencers que deixam que manipulemos esses dados. Muito comum, porém, é usar o comando Note on com velocidade zero para desligar a nota, e isso serve para podermos economizar alguns bytes, num processo chamado running status. E também podemos ver que é um evento.

Na coluna do Note podemos ver que mostra a nota musical a ser selecionada, como na primeira linha, que está escrita a letra C que representa a nota dó na linguagem musical.

Na coluna Chan que está abreviatura de Program Change é o som do instrumento musical a ser selecionado na hora que fosse tocar o acorde, no exemplo acima está CHAN 1, ou seja usou apenas um instrumento musical, se tivesse CHAN 2 em algumas dessas linhas quer dizer que usou outro instrumento.

A coluna do TIMESTAMP ela lista o momento em que cada mensagem ocorreu. Port seria em qual porta MIDI está, no caso do PC em qual porta do computador ela se encontra no caso da tabela 3.11.

A coluna do IN que está conectado na entrada 1 e está recebendo os dados. O Status na tabela seria se está sendo pressionado ou não. Podemos ver que 80 significa não está pressionado e 90 pressionado.

Os controladores MIDI através da ligação USB, estes permitem evitar uma placa de som, com a entrada MIDI, que tem um valor elevado. Os controladores por ligação USB precisam sempre de um programa no computador que faça a conversão digital para analógico. Onde aqui mostra um dos objetivos desse trabalho que é evitar uma placa de som que diminuiria o valor total do produto. O protocolo de comunicação MIDI possui bibliotecas prontas facilitando a implementação do projeto proposto.

3.5.1 Protocolo de Comunicação USB

USB é um protocolo de comunicação a qual suporta transferência de dados de computadores e periféricos.

Uma definição comercial para USB é um protocolo de comunicação de periféricos, desenvolvidos por grandes indústrias de telecomunicações e PCs que proporciona plug and play⁶ a alta velocidade, baixo custo e fácil uso.

O USB reage como um bus Master/Slave onde o Master é o USB Host que toma conhecimento da inserção e do desligamento dos devices⁷, onde se inicia o processo de enumeração e comanda todas as transações subsequentes. É responsabilidade dele coletar os status e as estatísticas de cada periférico. Os periféricos são chamados de slaves⁸ do

⁶A tecnologia ligar e usar (em inglês: Plug and Play ou ainda, PnP)

⁷É uma peça de equipamento de hardware de computador usada para fornecer dados e sinais de controle a um sistema de processamento de informações, como um computador ou dispositivo de informação

⁸é um modelo de comunicação em que um dispositivo ou processo possui controle unidirecional sobre

barramento, sendo que são de dois tipos; funcionais, como teclado, mouse, joysticks. E o outro slave é o Hub que funciona para conectar outros dispositivos.⁹

Uma das suas características que as especificações USB reconhecem até 1727 dispositivos ao mesmo tempo. O cabo USB consiste em apenas 4 fios, 2 deles D+ e D- são usados na transmissão de dados através de variações de sinais. O D+ é utilizado para transferência de dados a 12Mbps e o D- para transferência de dados a 1,5Mbps. Os outros dois fios são para a transmissão de energia e aterramento. Visto que as duas extremidades do cabo são diferentes e não permutáveis, isso permite a transmissão bidirecional e evita confusões nas conexões do cabo.¹⁰

3.5.2 Softwares de Áudio

Os softwares de áudio também possuem um padrão que é VST (Virtual Studio Technologic em inglês) desenvolvida pela Steinberg, conhecida empresa desenvolvedora de softwares para produção musical. É uma espécie de protocolo que permite o livre desenvolvimento de novos instrumentos virtuais na forma de plug-ins, que podem ser utilizados em qualquer software de áudio que possua suporte a VST. Para entender como funciona todo o processo deve-se entender o que são Estúdios Virtuais (LANGFORD, 2013).

Segundo (LANGFORD, 2013) Estúdios Virtuais são softwares para produção musical, que simulam no computador o ambiente de estúdio real, com todos os instrumentos necessários para a condensação sonora, edição, gravação de áudio e sua reprodução. No mercado existem várias marcas voltadas a diversas finalidades, desde de versões amadoras até as profissionais para estúdios. Exemplos: Reaper, Reason, Sónar, Guitar Rig, Protools, Orion entre muitos outros.

Cada Estúdio Virtual citado possui um setup de plug-ins único. O plug-in é cada um dos instrumentos musicais existente no software. O piano, os sintetizadores, módulos de efeitos, sequenciador de batidas entre outros, são todos considerados plug-ins. Esses plug-ins só funcionam dentro dos Estúdios Virtuais, mas geralmente são programas autônomos contendo arquivo próprio (LANGFORD, 2013).

Todo Estúdio Virtual com suporte a VST possui uma pasta em seu diretório, geralmente nomeada de VST. Toda vez que se inicia o programa, ele verifica na pasta todos os plug-ins. Os plug-ins que estiverem nelas são reconhecidas pelo programa. Esse é objetivo desse trabalho criar um controle onde seja reconhecido com plug-in e tenha a capacidade de controlar qualquer parâmetro dentro dos programas de Estúdio Virtual.

Na Figura 3.12 encontra-se uma das maneiras que se podem trabalhar Com o software com padrão VST. Mostra que através do VSTi, pedal do projeto, pode se controlar os

um ou mais dispositivos.

⁹ <<https://www.usb.org/>>. Acesso em: 03 de jun. 2019.

¹⁰<https://www.usb.org/>

efeitos que existem dentro do software. A origem do audio dentro do software pode vim de um arquivo wave ou de um instrumento musical.

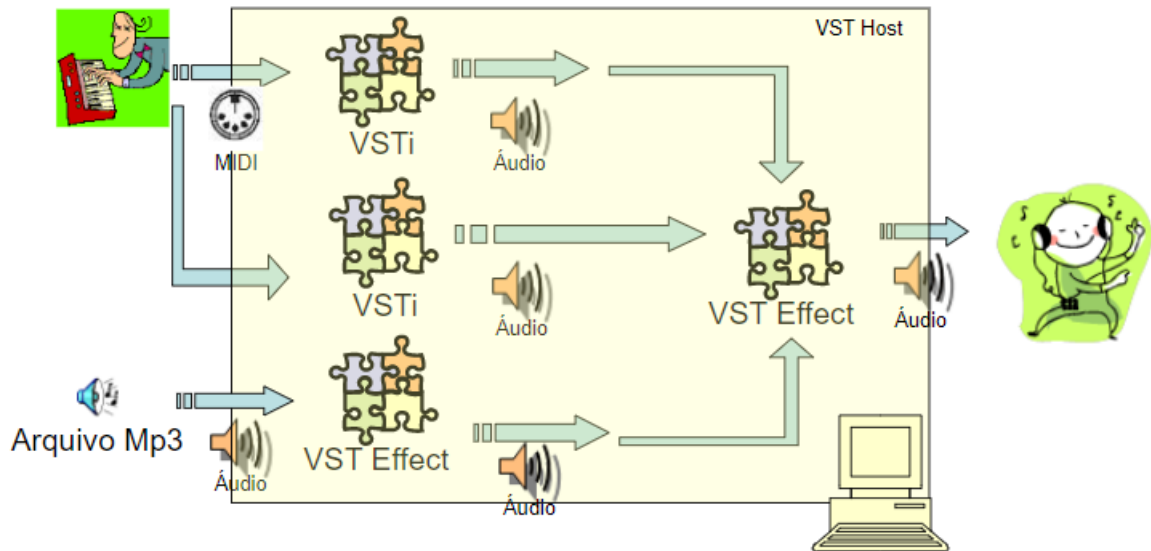


Figura 3.12: Arquitetura VST
Fonte: Hugo Santana, Virtual Studio Technology

3.5.3 Firmware

O Firmware é um conjunto de programas que vêm de fábrica armazenados na Memória Rom. Cada um desses programas presentes no chip possuem uma função específica. O Firmware armazena todas as informações sobre inicialização e rotinas para possibilitar o correto funcionamento de seu equipamento.

Uma segunda definição para Firmware: trata-se de um tipo de software específico que está associado a um dispositivo de hardware. Consiste em um ou mais programas de computador (instruções operacionais) instalados diretamente em um componente de hardware (microchip), Ou seja, um Firmware é gravado permanentemente em um circuito de memória durante o processo de fabricação do componente.

Neste projeto usou-se o HIDUINO que é um projeto open source disponível no GitHub sob licença MIT (Massachusetts Institute of Technnoly). Que consiste em um Firmware capaz de transformar um dispositivo Arduino UNO em um dispositivo MIDI USB nativo. Por padrão o Arduino é reconhecido pelo sistema operacional como um dispositivo de comunicação serial, sendo necessário um software intermediário como o Hairless MIDI<->Serial Bridge para transformar as mensagens recebidas pela porta serial em mensagens MIDI. Com o HIDUINO elimina-se a necessidade de softwares intermediários, pois o

controlador será reconhecido pelo sistema operacional como um dispositivo de interface humana (HID, do inglês Human Interface Device).

3.5.4 MIDIOX

o MIDIOX é derivado de um software de testes criado pela Microsoft em 1991, chamado MIDI Monitor, que foi aprimorado por Jamie O’Connel e Jerry Jorgenrud. Quando foi lançado o Windows 95, os dois programadores re-escreveram todo o código em 32 bits, com muitas funções adicionais, e batizaram a nova versão de MIDIOX. O MIDIOX é um software de grande utilidade para quem quer analisar a operação MIDI de seus equipamentos, e diagnosticar problemas. Ele pode mostrar todas as mensagens que o equipamento transmite, e também enviar as mensagens criadas pelo usuário, para testar ou configurar seus equipamentos. As telas principais do MIDIOX mostram as mensagens que estão sendo recebidas (Monitor-Input) e transmitidas (Monitor-Output) pelo software. Existem várias outras janelas e quadros que também podem ser visualizados, para acompanhar detalhes da operação MIDI.

Metodologia

O desenvolvimento deste trabalho foi dividido em três partes:

- Projeto Comportamental
- Projeto Lógico
- Projeto Físico

Cada parte possui sua especificação, execução e validação. Este relatório de Trabalho de Conclusão de Curso apresenta todas as parte do Projeto citadas acima, que serão detalhadas em cada subcapítulo.

No Projeto Comportamental, primeiramente é feita a ideia e avaliado as necessidades que o projeto demanda, algumas ideias são especificações físicas para de atender as necessidades do usuário. Na execução serão feitos todos os passos necessários para atender a parte de especificação, e analisar os possíveis erros. Por fim, a validação é o momento que une o que foi especificado à execução onde, se for compatível, validará esta parte do projeto.

O Projeto Lógico é definido como o planejamento de todos os detalhes para a construção do pedal. Ele envolve a parte de comunicação do protótipo ao dispositivo, a escolha do seu processador e sua programação para o envio de dados do sensor para o computador.

No Projeto Físico parte-se para o desenvolvimento da estrutura mecânica, com todo seu dimensionamento, a escolha do material e a placa de circuito impresso; onde envolverá conectores, alimentação e a ligação direta com o sensor.

4.1 Projeto Comportamental

Esta é a seção do projeto que define o comportamento em que o projeto se enquadra.

4.1.1 Especificações

Esta subseção define as condições mínimas esperadas do pedal, para que faça o mesmo funcionar adequadamente. Os critérios são:

- O pedal deve medir dinamicamente o ângulo que o pedal faz com o solo de acordo com o movimento que o músico faz para pisar
- O pedal deve comunicar-se com o PC e informar o ângulo ao sistema operacional
- O pedal deve apresentar sensibilidade tal que permita ao músico se expressar artisticamente
- O pedal deve ser robusto e durável, ser resistente a quedas e a fluídos

4.1.2 Execução

A execução mostra como o usuário vê o pedal de expressão para sua utilização. O fluxograma a seguir da Figura 4.1 exemplifica detalhadamente.

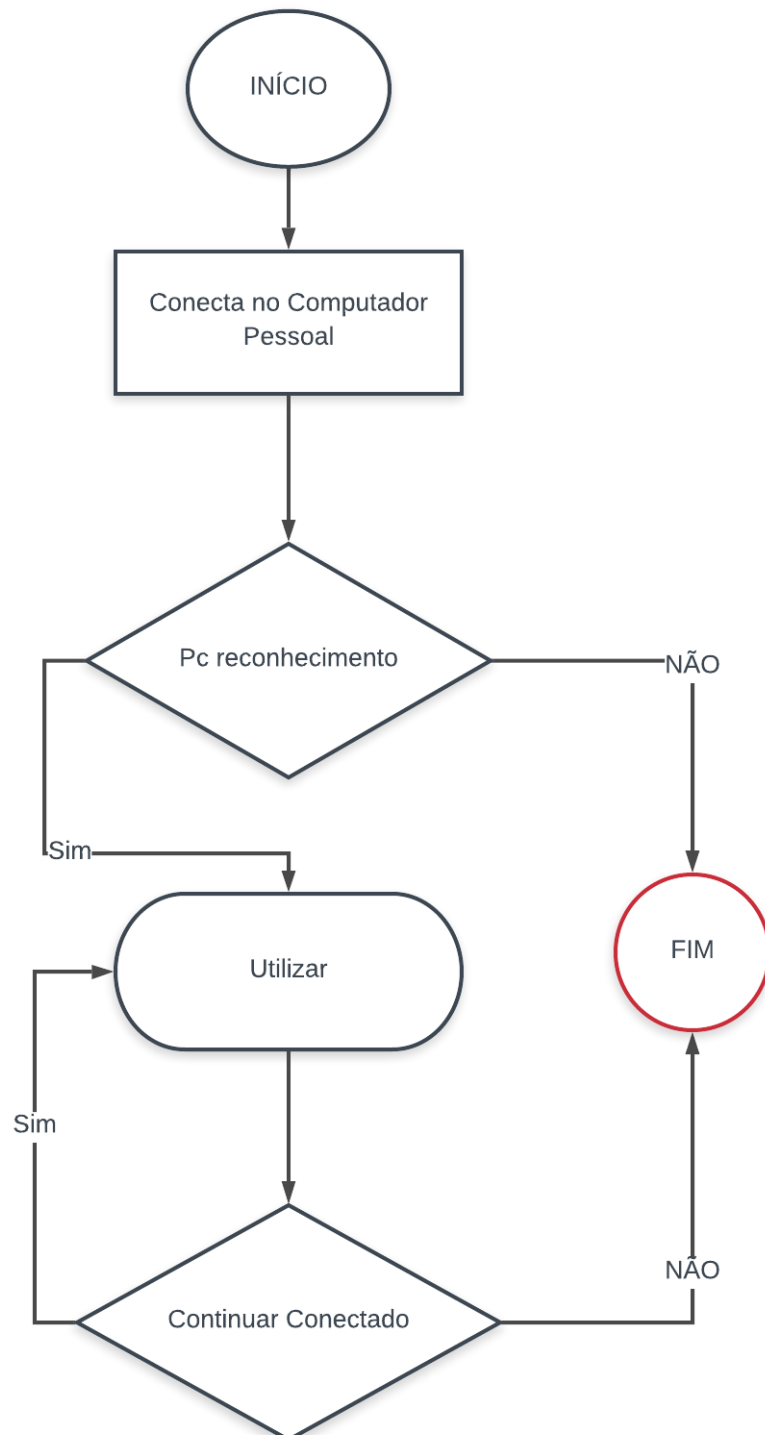


Figura 4.1: Fluxograma do pedal
Fonte: (Arquivo pessoal do autor)

Ao adquirir o pedal de expressão o usuário conecta sua entrada USB no computador, em seguida temos dois estados; onde o computador reconhece ou não. Se não reconhecer será o fim do processo, ou seja, nada poderá ser feito. Caso reconheça, o usuário pode utilizar o pedal de expressão. Em seguida quando estiver em uso terá duas opções; não optar por continuar conectado e finalizar o processo ou se continuar o processo, ele volta em utilizar, podendo fazer esse ciclo infinitamente até não querer estar conectado.



Figura 4.2: Diagrama de blocos do sistema
Fonte: (Arquivo pessoal do autor)

A figura 4.2 mostra um diagrama de blocos do pedal de expressão, em que se tem o sensor, nele será gerado uma medida para a qual o microprocessador interpretará os dados em sua entrada. Em seguida, traduzirá os dados que serão enviados para o computador pessoal de forma que ele interprete dentro do software escolhido.

O projeto tem apenas duas alternativas: ligado (on) ou desligado (off). Ou seja, o pedal não possui um estado intermediário. Portanto, neste capítulo é possível observar as etapas do projeto comportamental, que é de extrema importância para saber as qualidades do pedal de expressão e prever possíveis problemas antes de avançar para as próximas etapas do projeto lógico e físico.

4.1.3 Arquitetura do sistema

Na fase inicial do projeto foram definidas algumas funcionalidades que o sistema deverá ter para cumprir os objetivos propostos. O sistema foi dividido em duas partes de usuários onde; em que uma é o arquiteto e outro é o utilizador. O arquiteto será o projetista do sistema, este poderá configurar a máquina conforme desejar, podendo alterar a programação interna do pedal mudando algumas funcionalidades como a de mapeamento de ângulo do sensor, para enviar as mensagens MIDI. A Figura 4.3 mostra uma tabela que identifica os módulos utilizados pelo arquiteto, assim como suas funções.

Controlador MIDI	
Arquiteto	
Módulo	Função
Configurar	O arquiteto pode configurar todo o sistema, ou seja, pode alterar os sons configurados.
Implementar	O arquiteto pode fazer alterações ao equipamento, implementando novos módulos ao controlador midi.
Arquivar	Permite arquivar qualquer tipo de ficheiro para a memória interna do controlador midi.

Figura 4.3: Apresentação dos módulos referentes ao arquiteto
Fonte: Autoria Própria

O utilizador irá apenas tirar partido das funcionalidades que se encontram no controlador de som, assim como; modelagem de sons, gravação e configuração. Tudo será feito através de um VST.

A Figura 4.4 mostra exatamente as possibilidades que o utilizador terá com o pedal de expressão midi.

Módulo	Função		
Modelar Sons	Este módulo será dividido em três partes em que teremos volume, velocidade.	Volume	Nesta secção é possível controlar o volume do programa que se está a utilizar através do potenciómetro que está implementado no controlador midi
		Velocidade	Nesta secção é possível controlar a velocidade (BPM) das músicas que estão a ser reproduzidas pelo programa.
Gravação	Faz a gravação de som através de uma entrada de um microfone para um ficheiro wave		
Auto	Tem a possibilidade da criação de sons através de um conversor onde se conecta ao computador juntamente com um instrumento		
Configurar	Neste módulo é possível fazer o mapeamento dos potenciómetros.		
Arquivar	Permite arquivar qualquer tipo de música na memória interna do computador a qual está em uso.		

Figura 4.4: Apresentação dos módulos referentes ao utilizador
 Fonte: Autoria Própria

A Figura 4.5 apresentada os casos de uso desse projeto.

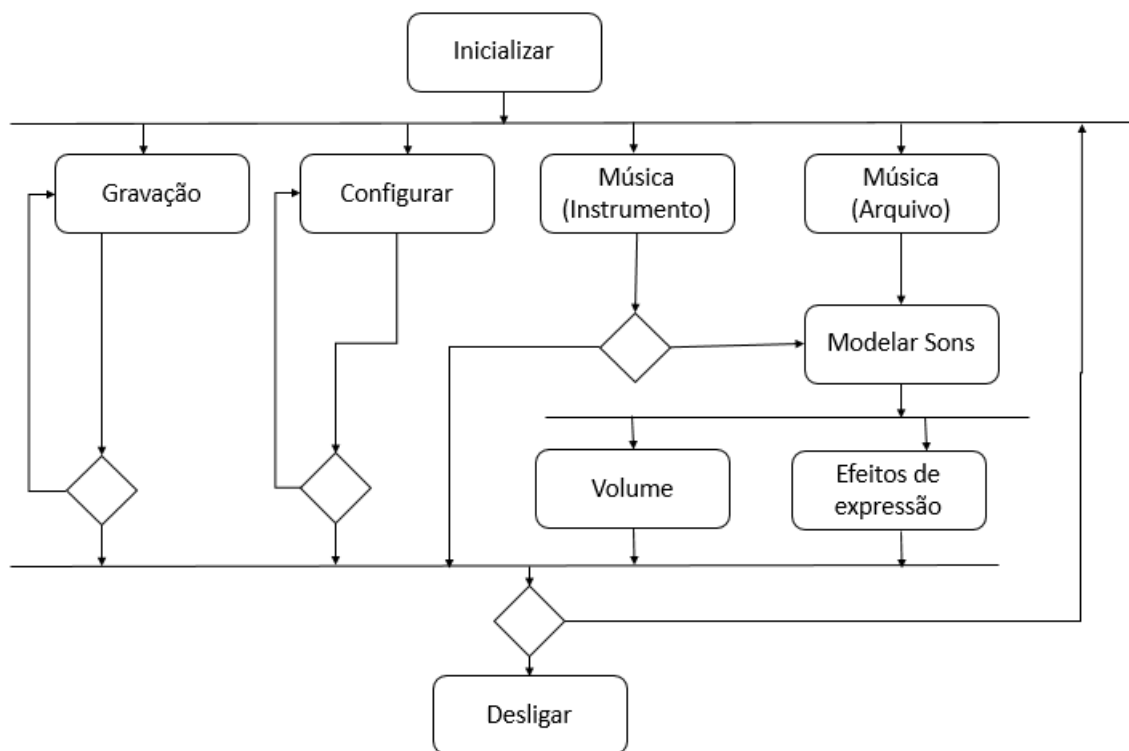


Figura 4.5: Apresentação dos módulos referentes ao utilizador

Fonte: Autoria Própria

Os casos de usos implementados são os seguintes:

Caso 1:

- Nome: Implementar;
- Descrição: Este caso permite ao arquiteto fazer a configuração do sistema;
- Atores envolvidos: Arquiteto;
- Pré-condições: Conhecimento do código implementado e todo sistema da placa do pedal
- Fluxo: O Arquiteto pode fazer a alteração dos efeitos sonoros dentro do programa VST.

Caso 2:

- Nome: Implementar;
- Descrição: Este caso permite ao arquiteto fazer a implementação de um sistema novo no controlador midi;
- Atores envolvidos: Arquiteto;

- Pré-condições: Conhecimento do código e da implementação do hardware;
- Fluxo: O arquiteto pode fazer alterações no equipamento, implementando a varredura do sensor no controlador MIDI.

Caso 3:

- Nome: Inicializar;
- Descrição: Este caso permite ao utilizador ligar o controlador midi;
- Atores envolvidos: Arquiteto;
- Pré-condições: O equipamento deve que estar ligado por USB a um computador que tenha instalado um programa padrão VST;
- Fluxo: Permite o início da utilização de todo o equipamento e funcionalidade.

Caso 4:

- Nome: Música (Instrumental);
- Descrição: Este caso permite ao utilizador fazer suas próprias composições com algum instrumento a sua escolha;
- Atores envolvidos: Utilizador;
- Pré-condições: Ter algum instrumento musical que deve ser ligado a uma placa de som;
- Fluxo: Através do pedal é possível alterar qualquer parâmetro dentro do programa escolhido VST. Podemos citar como exemplo; volume, velocidade ou qualquer função que o programa VST do usuário tiver.

Caso 5:

- Nome: Música (Arquivo);
- Descrição: Este caso permite ao utilizador fazer alterações nas músicas com formato wave;
- Atores envolvidos: Utilizador;
- Pré-condições: Ter músicas no seu disco rígido;
- Fluxo: Através do pedal é possível alterar qualquer parâmetro dentro do programa escolhido VST. Podemos citar como exemplo; volume, velocidade ou qualquer função que o programa VST do usuário tiver.

Caso 6:

- Nome: Modelar Sons;
- Descrição: Este caso permite ao utilizador fazer alterações as músicas que vão ser produzidas;
- Atores envolvidos: Utilizador;
- Pré-condições: ter um equipamento inicializado e conhecimento dos efeitos que estão configurados no controlador;
- Fluxo: Através do pedal é possível alterar qualquer parâmetro dentro do programa VST escolhido. Podemos citar como exemplo; volume, velocidade ou qualquer função que o programa VST do usuário tiver.

Caso 7:

- Nome: Volume;
- Descrição: Este caso permite a alteração do volume de reprodução;
- Atores envolvidos: Utilizador;
- Pré-condições: ter um equipamento inicializado e conhecimento dos efeitos sonoros que os vários tipos de volume efetuam, low, mid e high;
- Fluxo: O utilizador pode fazer alterações ao volume assim como alterações aos low's, mid e high.

Caso 8:

- Nome: Efeitos de expressão;
- Descrição: Este caso permite o controle de efeitos que o programa VST possui, a quantidade de efeitos utilizados e controlados são de acordo com cada programa com padrão VST utilizado;
- Atores envolvidos: Utilizador;
- Pré-condições: ter um equipamento inicializado e conhecimento dos efeitos sonoros que o pedal de expressão permiti controlar.

Caso 9:

- Nome: Gravação;

- Descrição: Este caso permite ao utilizador fazer a gravação através de um microfone dentro dos programas VST;
- Atores envolvidos: Utilizador;
- Pré-condições: Ter um equipamento inicializado e estar dentro do programa escolhido, com o padrão VST. O computador necessitará de um microfone
- Fluxo: Através do microfone é possível fazer a gravação de qualquer som. Após o som gravado é possível fazer alterações com o uso de pedal de expressão midi.

Caso 10:

- Nome: Configurar;
- Descrição: Este caso permite ao utilizador configurar o sensor do controlador MIDI;
- Atores envolvidos: Utilizador;
- Pré-condições: Ter um equipamento inicializado, conhecimento do programa e o modo de mapeamento.
- Fluxo: O utilizador pode configurar o sensor através do mapeamento dos programas com padrão VST.

4.2 Projeto Lógico

Será descrito a lógica do sistema proposto e a maneira como as informações serão construídas e disponibilizadas aos interessados.

Definirá a tecnologia de base para o projeto físico onde serão relatadas as configurações necessárias para hardware e software. O Projeto Lógico é independente da implementação ou tecnologia que será executada para produzir o projeto.

Para o projeto lógico se definiu as seguintes especificações:

- O dispositivo embarcado deve ser alimentado com uma tensão de 5V, fornecida pela entrada USB do computador.
- O dispositivo embarcado deve ter o menor custo possível.
- O dispositivo tem que ser reconhecido automaticamente como um instrumento MIDI em qualquer computador

Para realizar esses procedimentos é necessário a escolha do componentes a serem utilizados.

4.2.1 Arquitetura da plataforma

A plataforma foi construída através do Arduino Uno, para implementar um controlador de som. O funcionamento em tempo real e de rápida resposta do microcontrolador ATmega328P, é de grande importância para o funcionamento do pedal de expressão midi, pois assim permite que não exista nenhum delay durante o seu tempo de funcionamento.

A Figura 4.6 apresenta a arquitetura da plataforma e a Figura 4.7 as dependências do mecanismo, onde está incluído todo o processo necessário para o funcionamento do controlador midi, hardware (sensor) e o Arduino.

Como mostra a figura, a plataforma para além do micro controlador ATmega328P dispõe dos seguintes módulos:

- Comunicação por USB, que garante a comunicação entre o computador e a plataforma e também a alimentação de energia do sistema;
- Comunicação do tipo UART¹, que através deste módulo irá ser feita a ligação com o software VST permitindo a comunicação rápida entre os dispositivos a serem utilizados;
- Hardware (sensor), para dar início aos eventos pré-definidos ao movimento do potenciômetro.

¹UART (Universal Asynchronous Receiver/Transmitter) é o componente chave do subsistema de comunicação serial de um computador. O UART pega bytes de dados e transmite os bits individuais de forma sequencial. No destino, um segundo UART reúne os bits em bytes completos.

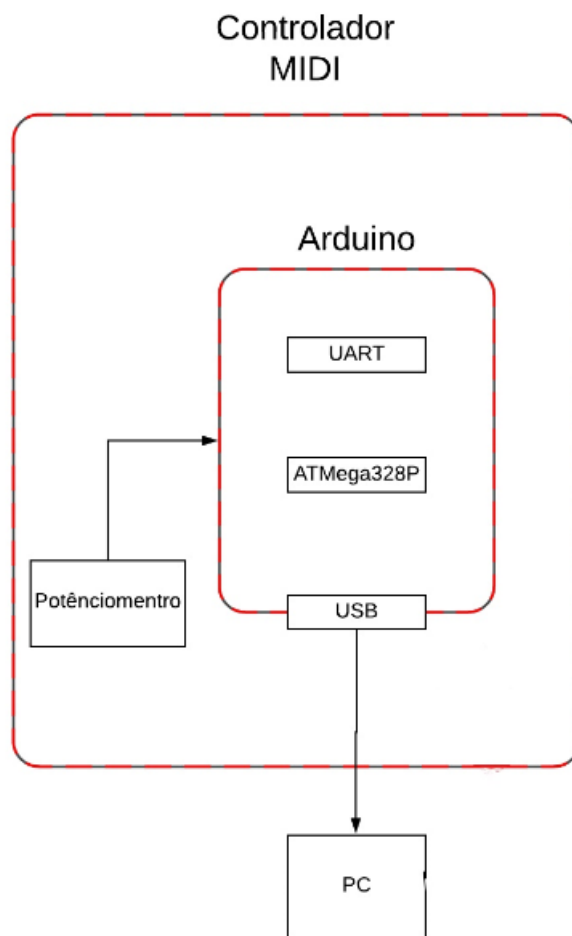


Figura 4.6: Controlador Midi
Fonte: Autoria Própria

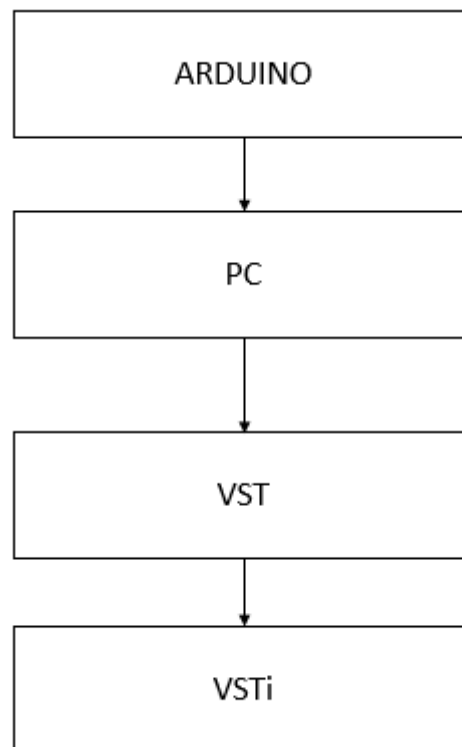


Figura 4.7: Dependências do Mecanismo
Fonte: Autoria Própria

A Figura 4.7 mostra as dependências do mecanismo, ou seja;

- É no Arduino que está implementada toda a parte de hardware do controlador midi.
- O PC é a base da comunicação entre o Arduino e os programas que são necessários para o funcionamento do controlador.
- O VST é uma interface que integra sintetizadores e efeitos de áudio com editores e dispositivos de gravação de sons digitais.
- O VSTi é o formato em que se enquadra o pedal de expressão para funcionar corretamente dentro do software VST.

4.2.2 Projeto de Software

O Protocolo de comunicação

O protocolo utilizado para a comunicação do dispositivo com o software compatível é o MIDI, já foram realizado o estudo das especificações técnicas deste protocolo, disponíveis no manual.

Leitura das entradas

Para as leituras de entrada foi necessário escolher o sensor mais viável entre os três comentados no capítulo de Fundamentação Teórica. O sensor escolhido foi o potenciômetro, o qual já foi especificado suas características. A leitura da entrada do potenciômetro foi feita utilizando o método de varredura. Um loop infinito verifica constantemente o estado de cada entrada e, quando há alteração no estado (por exemplo: o potenciômetro foi movido para qualquer direção), é gerada e enviada a mensagem MIDI previamente atribuída a esta entrada.

Entradas analógicas: potenciômetro

O potenciômetro foi ligado a entrada analógica do Arduino, Figura 4.16, em seguida, de acordo com programa feito (anexo), a leitura pode ser realizada no método de varredura. A posição do potenciômetro resulta em um valor numérico que varia entre 0 e 1023, uma vez que o conversor analógico/digital do Arduino trabalha com uma resolução de 10 bits, variando entre 0 e 127, portanto foi necessário realizar uma conversão entre essas escalas, mostrada na Figura 4.8.

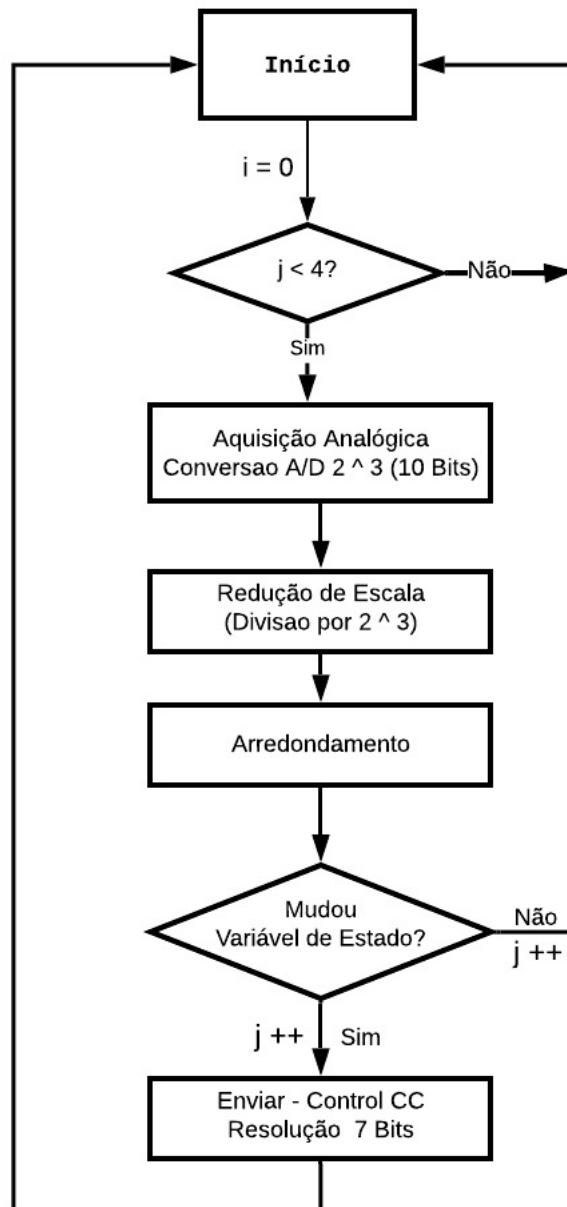


Figura 4.8: Lógica do sensor
Fonte: (Arquivo pessoal do autor)

As Figuras 4.9 e 4.10 encontramos a calibração do sensor. Verificou-se a validação através de testes realizados pelo autor. A validação foi feita com ajuda do MIDIOX que manda mensagens em hexadecimal a cada variação de ângulo.

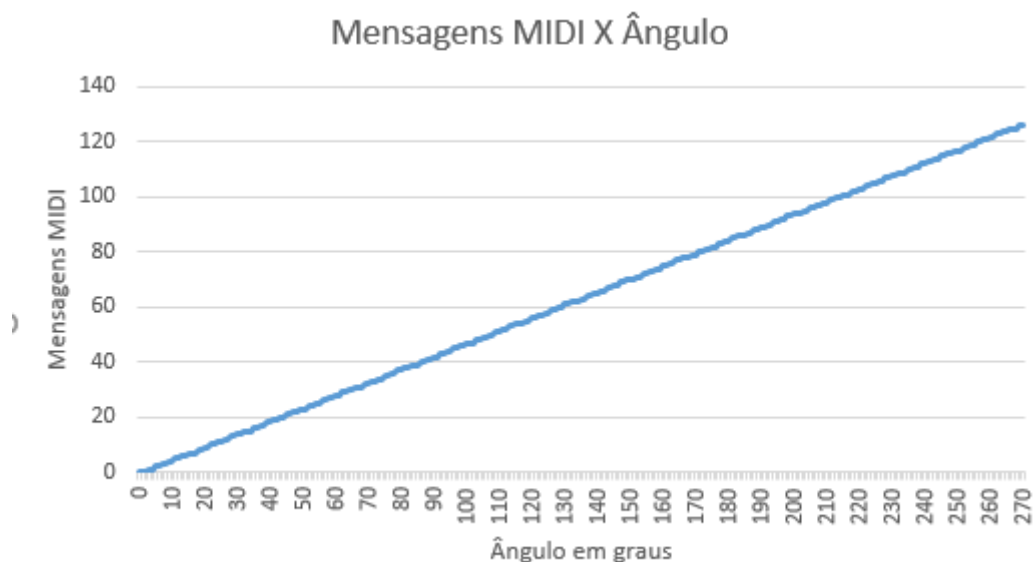


Figura 4.9: Relação Mensagens MIDI com o ângulo
Fonte: (Arquivo pessoal do autor)

A Figura 4.9 é possível observar que no eixo x encontramos a variação de ângulo em graus que vai de 0 até 270 graus, que equivale ao grau de rotação do potenciômetro. No eixo y temos o número de mensagem, que pelo gráfico indica 127 mensagens. Relacionando os 2 eixos temos a relação de mensagem MIDI a cada posição de grau, lembrando que as mensagens MIDI nos softwares trabalham apenas com números inteiros. Pela divisão de mensagens por ângulo encontramos que a cada 2,125 graus se manda uma mensagem MIDI, mas as mensagens MIDI que são enviadas em hexadecimal só trabalham com valores inteiros, portanto a cada 2 graus se envia uma mensagem MIDI. Se, por exemplo, variar o cursor a 3 graus, a mensagem enviada será a mesma que em 2 graus. No caso do projeto o pedal varia uma angulação de 48 graus.

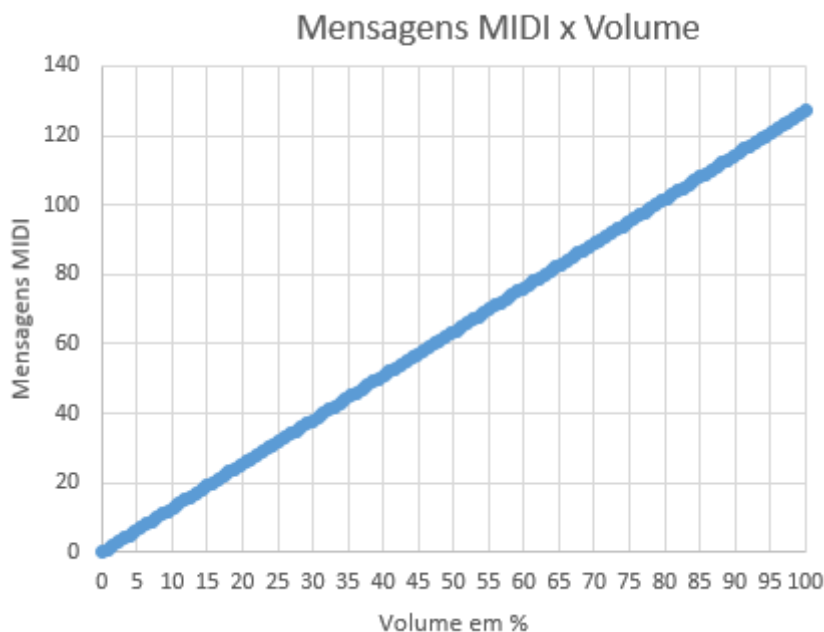


Figura 4.10: Relação Mensagens MIDI com o Volume
Fonte: (Arquivo pessoal do autor)

A Figura 4.10 mostra a relação do envio das mensagens e o quanto esse envio altera no volume. As mensagens MIDI, elas só trabalham com números inteiros. A consequência direta no projeto seria que, em alguns pontos teremos zona morta, que é quando se tem pontos de movimento no potenciômetro que não envia mensagens, já que as mensagens são enviadas somente quando temos um número inteiro.

4.3 Projeto Físico

Nesta seção será feita a implementação física do projeto, a mesma contempla desde a parte de confecção do programa até a parte mecânica. Possui um controle de testes e qualidade.

4.3.1 Projeto de Hardware

Definição dos componentes

A primeira etapa para o desenvolvimento deste trabalho consistiu em definir os componentes eletrônicos a serem utilizados, onde o mais importante é o Arduino. Um dos principais motivos seria a possibilidade de transformá-lo em um dispositivo MIDI Class Compliant, ou seja, após programado quando ligado a um porta de USB, o computador o reconhecerá como dispositivo MIDI, sem necessidade de instalação de drivers ou softwares intermediários para isso.

Componentes eletrônicos utilizados:

- 1 Arduino UNO
- 1 potenciômetro giratório linear de 1M ohm
- Fios para a conexão dos componentes

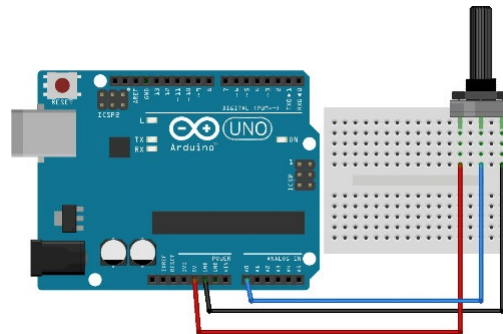


Figura 4.11: Ligação do circuito
Fonte: (Arquivo pessoal do autor)

Na Figura 4.11 mostra como será ligado o sensor no arduino, que a ligação se baseia em que o fio vermelho está conectado no Vcc 5V, o fio preto conectado no GND e o fio azul conectado na porta analógica A0.

Validando a comunicação com o computador

Na Figura 4.12 mostra a validação feita utilizando o Hiduino, no qual mudou o reconhecimento da placa Arduino para um instrumento MIDI.



Figura 4.12: Validação da Comunicação
Fonte: (Arquivo pessoal do autor)

4.4 Validação do Código

Para validar o comportamento do dispositivo e analisar se as saídas estão de acordo com o planejado foi usado o software MIDI OX a Figura 4.13, que funciona como um sniffer, analisando e exibindo na tela toda comunicação trocada entre dispositivos MIDI e o computador. Quando se obtém o conjunto de informação abaixo, isso significa que o dispositivo midi em questão está se comunicando com o computador de alguma forma. Se o dispositivo midi não estiver comunicando com o computador o MIDI OX não irá reconhecer e não acionará a opção de adicionar um instrumento MIDI ou responderá na sua DAW (estação de trabalho e áudio digital).

TIMESTAMP	IN	PORT	STATUS	DATA1	DATA2	CHAN	NOTE	EVENT
00004494	1	--	B0	01	6F	1	---	CC: Modulation
0000449A	1	--	B0	01	6E	1	---	CC: Modulation
0000449D	1	--	B0	01	6F	1	---	CC: Modulation
000044A6	1	--	B0	01	70	1	---	CC: Modulation
000044A9	1	--	B0	01	6F	1	---	CC: Modulation
000044A9	1	--	B0	01	70	1	---	CC: Modulation
000044AA	1	--	B0	01	6E	1	---	CC: Modulation
000044AB	1	--	B0	01	6F	1	---	CC: Modulation
000044B3	1	--	B0	01	70	1	---	CC: Modulation
000044B4	1	--	B0	01	6F	1	---	CC: Modulation
000044B5	1	--	B0	01	70	1	---	CC: Modulation
000044B8	1	--	B0	01	71	1	---	CC: Modulation
000044B9	1	--	B0	01	70	1	---	CC: Modulation
000044C7	1	--	B0	01	71	1	---	CC: Modulation
000044D8	1	--	B0	01	72	1	---	CC: Modulation
000044DE	1	--	B0	01	71	1	---	CC: Modulation
000044E4	1	--	B0	01	72	1	---	CC: Modulation
000044F7	1	--	B0	01	73	1	---	CC: Modulation
0000450A	1	--	B0	01	74	1	---	CC: Modulation
0000451B	1	--	B0	01	75	1	---	CC: Modulation
00004520	1	--	B0	01	73	1	---	CC: Modulation
00004521	1	--	B0	01	74	1	---	CC: Modulation
00004527	1	--	B0	01	75	1	---	CC: Modulation
00004527	1	--	B0	01	74	1	---	CC: Modulation
00004528	1	--	B0	01	75	1	---	CC: Modulation
0000453C	1	--	B0	01	76	1	---	CC: Modulation
0000455D	1	--	B0	01	77	1	---	CC: Modulation
0000456F	1	--	B0	01	78	1	---	CC: Modulation
00004571	1	--	B0	01	77	1	---	CC: Modulation
0000457E	1	--	B0	01	78	1	---	CC: Modulation
00004590	1	--	B0	01	79	1	---	CC: Modulation
00004593	1	--	B0	01	78	1	---	CC: Modulation
0000459E	1	--	B0	01	79	1	---	CC: Modulation
000045B2	1	--	B0	01	7A	1	---	CC: Modulation
000045B6	1	--	B0	01	79	1	---	CC: Modulation
000045BD	1	--	B0	01	7A	1	---	CC: Modulation
000045DA	1	--	B0	01	7B	1	---	CC: Modulation
000045F3	1	--	B0	01	7C	1	---	CC: Modulation

Figura 4.13: Validação Código
 Fonte: (Arquivo pessoal do autor)

Desenho e construção

Após a definição dos componentes eletrônicos, foi feito o desenho mecânico da sua parte estrutural. Por se tratar de um dispositivo que poderá ser utilizado por um músico durante sua apresentações ao vivo, é importante que o pedal seja resistente ao peso do usuário, resistente a fluídos e a possíveis impactos que podem ocorrer durante o transporte. O material utilizado na sua construção foi o alumínio fundido que se encaixa perfeitamente nos atributos citados acima, além de proporcionar uma melhor estética ao pedal de expressão. As peças foram desenhadas no software SolidWorks. O projeto mecânico foi uma adaptação dos milhares de pedais de expressão que já existem no mercado. Um único detalhe que demandou maior atenção foi projetar e construir um pedal onde seja possível colocar o Arduino dentro de sua estrutura física.

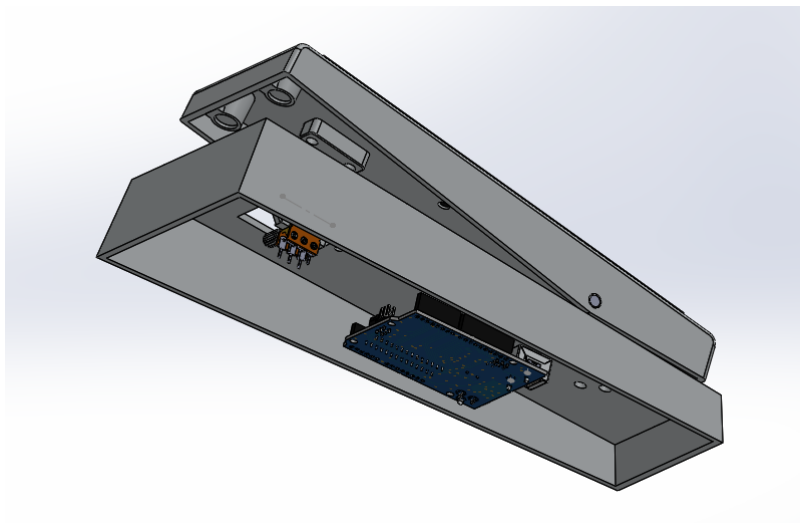


Figura 4.14: Pedal de expressão visto de baixo, com o arduino instalado e o sensor
Fonte: (Arquivo pessoal do autor)

Na Figura 4.14 é possível identificar como está a posição dos componentes dentro do Pedal de expressão midi. Pode-se observar a posição do arduino na região central do pedal, mais acima vemos onde o sensor (potênciometro) está localizado.

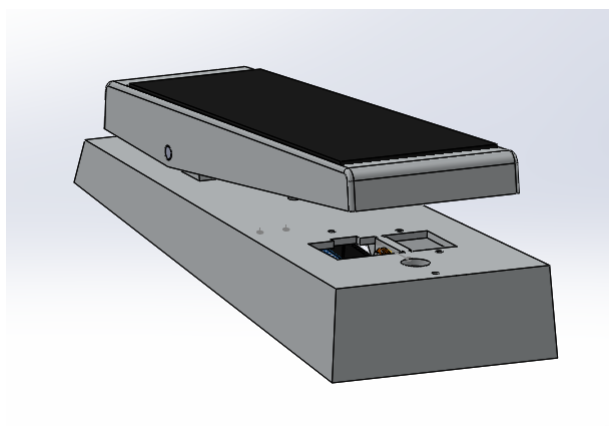


Figura 4.15: Pedal de expressão
Fonte: (Arquivo pessoal do autor)

Na Figura 4.15 temos o pedal em sua frontal o qual, ao se movimentar, a superfície superior vai mover o sensor para frente ou para trás, enviando mensagens midi. Essa superfície superior, que se movimentará com o pé, terá uma régua dentada (Figura 4.16) fixada que de acordo com o movimento que terá seu sensor (potenciômetro, Figura 4.17), no sentido de girar pra direita ou para esquerda, mandando mensagens midi de acordo com sua posição.



Figura 4.16: Régua Dentada
Fonte: (Arquivo pessoal do autor)



Figura 4.17: Potenciômetro com adaptador para régua dentada
Fonte: (Arquivo pessoal do autor)

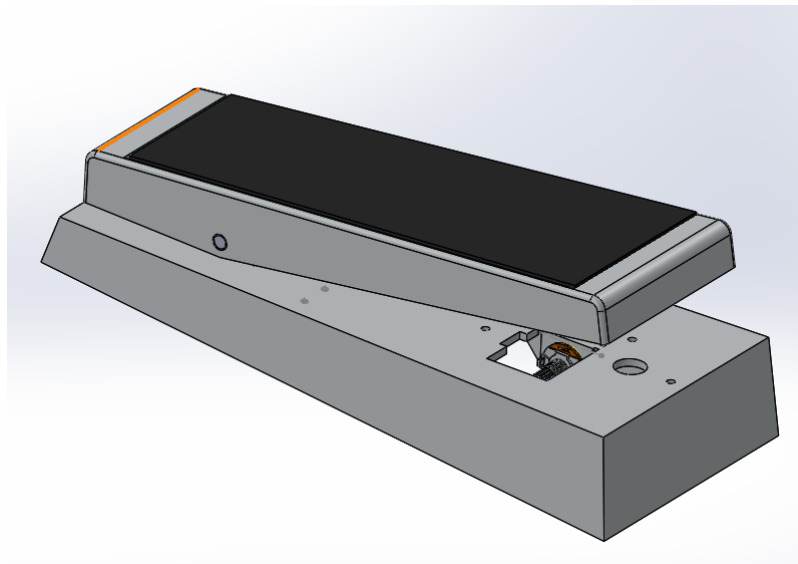


Figura 4.18: Pedal de expressão
Fonte: (Arquivo pessoal do autor)

Na Figura 4.18 encontra-se o pedal em sua vista lateral onde podemos ver que se forma um ângulo da plataforma, com a base no qual essa mudança de ângulo transformará em mensagens midi.

Capítulo 5

Resultados

Um dos resultados foi o reconhecimento no software Guitar Rig, esse software trabalha com modelagem de amplificadores e efeitos desenvolvidos pela empresa Native Instruments. O software usa modelagem do amplificador para permitir o processamento de sinal digital em tempo real. O Guitar Rig é um sistema modular, que oferece recurso para vários amplificadores, pedais de efeitos e hardware montado em rack. O software simula vários dispositivos, como pré-amplificadores, gabinetes, pedais e microfones. O sistema permite a personalização dos parâmetros do módulo, seja através da manipulação da interface gráfica ou, como no caso desse projeto, que é um controlador midi.



Figura 5.1: Interface Guitar Rig
Fonte: (Arquivo pessoal do autor)

A Figura 5.1 mostra o Pedal de Expressão MIDI, já conectado ao computador. Após selecionar o instrumento midi nas opções, pode-se ver que o programa reconhece o pedal de

expressão. É possível observar na parte New Control, no qual pode escolher o parâmetro que deseja controlar. No caso do Guitar Rig não existem limites de parâmetros para serem controlados, ou seja, é possível colocar infinitos parâmetros para serem controlados apenas um pedal de expressão midi.

Através da Figura 5.1, que foi realizado o teste pelo autor, foi escolhido controlar o Volume do rack que esta localizado no cabeçote TapeDeck. A movimentação das plataformas que estão ligadas diretamente ao sensor do pedal controlam o volume. Para fins de teste, foram mapeadas as funções mais utilizadas para mixagens ao vivo: volume, efeitos, equalizador e filtros. Todos eles apresentaram uma resposta satisfatória de acordo com a sensibilidade do pedal. A escolha das funções as quais o pedal controla fica a critério do utilizador.

A construção mecânica do pedal foi feita através de moldes. Usou-se de matéria prima o alumínio fundido que foi dividido em dois moldes, que no final se unem para transforma em uma única peça. Ao juntar essas duas partes temos o resultado da Figura 5.2, onde a régua dentada não faz parte desse processo.

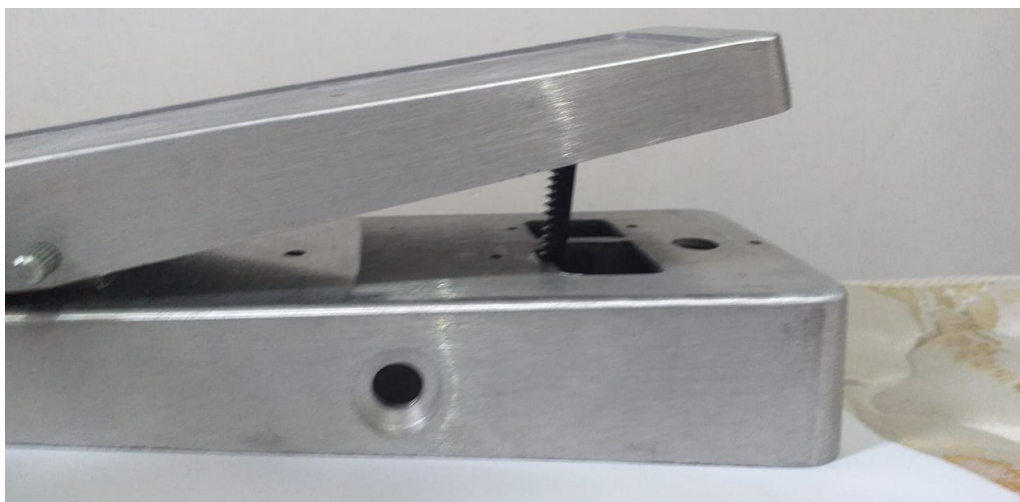


Figura 5.2: Parte Física do Pedal
Fonte: (Arquivo pessoal do autor)

O tempo de resposta do pedal foi satisfatório já que a resposta foi de 0,1 segundos, ou seja um atraso irrelevante para um controlador que será usado ao vivo. O resultado obtido foi bastante satisfatório, durante os testes as ações realizadas no controlador refletiram instantaneamente nas funções mapeadas no software, com uma latência imperceptível.

Considerações Finais

Neste capítulo são apresentadas as conclusões do Trabalho de Conclusão de Curso.

6.1 Conclusões

Após pesquisar vários tipos de materiais sobre os assuntos relacionados ao projeto, e então criar um dispositivo midi, concluiu-se ser possível construir um pedal de expressão midi, compatível com diferentes softwares e sistemas operacionais e que apresenta um baixo custo se comprado as opções disponíveis atualmente no mercado. Com a mapeação das funções de um software em controles físicos será possível aumentar as possibilidades de utilização do mesmo.

O trabalho é finalizado de forma que todos os objetivos específicos definidos inicialmente foram atingidos, uma vez que foram utilizados conceitos de eletrônica digital, como softwares e programação, alguns estudos de conceitos mecânicos, o comportamento do microcontrolador e a programação no Arduino, para que funcione como um dispositivo MIDI USB independente.

Anexos

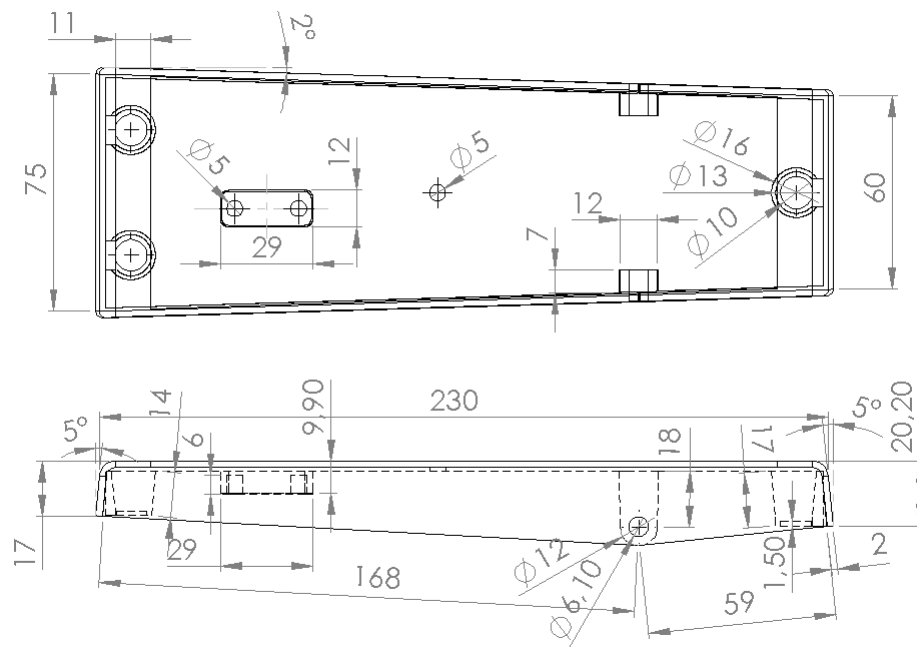


Figura 7.1: Dimensões
Fonte: (Arquivo pessoal do autor)

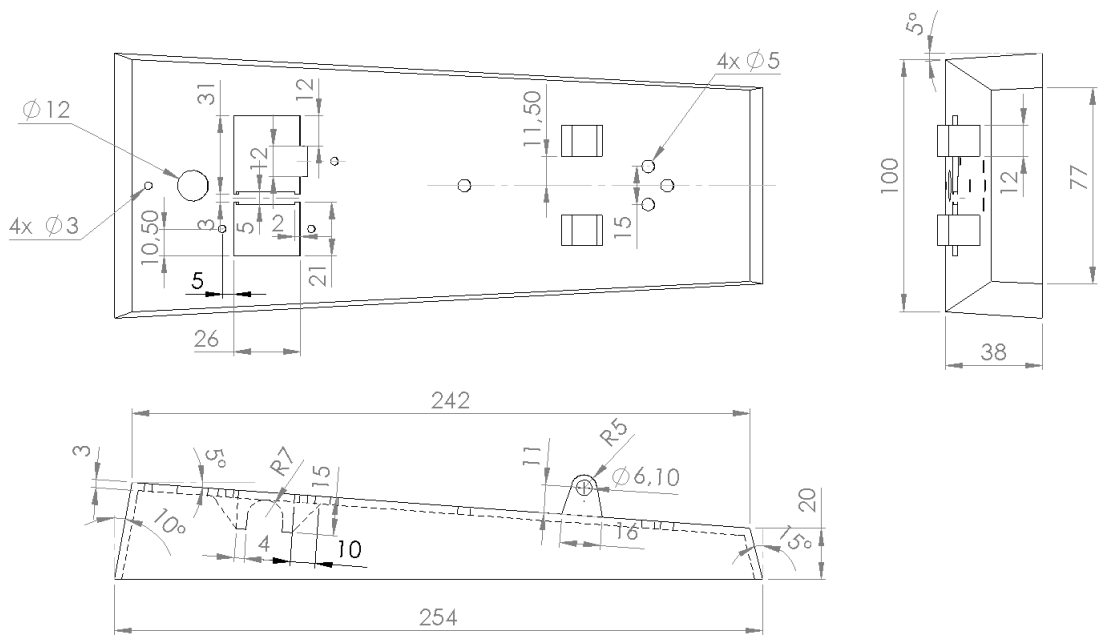


Figura 7.2: Dimensões
 Fonte: (Arquivo pessoal do autor)



Figura 7.3: Vista Lateral do Pedal
 Fonte: (Arquivo pessoal do autor)



Figura 7.4: Vista Lateral com a régua dentada para mover o potenciômetro
Fonte: (Arquivo pessoal do autor)



Figura 7.5: Vista superior da parte externa do pedal
Fonte: (Arquivo pessoal do autor)



Figura 7.6: Vista Superior da parte interna do pedal
Fonte: (Arquivo pessoal do autor)

Referências

- ARDUINO, S. A. Arduino. *Arduino LLC*, 2015.
- BENNETT, R. *Uma breve história da música*. : s.n., 1986.
- EARLE, J. H. *Graphics for Engineers: autocad release 11*. : Addison-Wesley, 1992.
- HAMMER, E. *MUSICAL APPLICATIONS OF MICROPROCESSORS-CHAMBERLIN, H.* : MIT PRESS 55 HAYWARD ST JOURNALS DEPT, CAMBRIDGE, MA 02142, 1986.
- HUBER, D. M. *The MIDI manual: a practical guide to midi in the project studio*. : Taylor & Francis, 2007.
- JENKINS, L. *Manual ilustrado dos instrumentos musicais*. : Irmãos Vitale, 2009.
- KARPF, B. W. *Sound and Code: the effects of digital technology on music-making*. 2003. Tese (Doutorado em Ciência da Computação) — Georgetown University.
- KOGAN, R. *Brief History of Electronic and Computer Musical Instruments*. : Citeseer, 2008.
- LANGFORD, S. *Digital audio editing: correcting and enhancing audio in pro tools, logic pro, cubase, and studio one*. : Routledge, 2013.
- MACLEAN, C. The Principle of the Hydraulic Organ. *Sammelbände der Internationalen Musikgesellschaft*, v.6, n.H. 2, p.183–236, 1905.
- PAUL, F. *Mercado de instrumentos musicais volta a crescer no Brasil*. : Addison-Wesley, 2019.
- TANENBAUM, A. S. *Structured computer organization*. : Pearson Education India, 2016.
- WENDLING, M. Sensores. *Universidade Estadual Paulista. São Paulo*, v.2010, p.20, 2010.