

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE  
MINAS GERAIS - CAMPUS BETIM  
BACHARELADO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Vinicius Alefe de Aquino

**PROJETO, CONSTRUÇÃO E ANÁLISE DE UM SIMULADOR  
AUTOMOBILÍSTICO**

Betim

2022

VINICIUS ALEFE DE AQUINO

**PROJETO, CONSTRUÇÃO E ANÁLISE DE UM SIMULADOR  
AUTOMOBILÍSTICO**

Trabalho de Conclusão de Curso apresentado à banca examinadora do curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – *Campus Betim*, como parte dos requisitos para obtenção do título de Bacharel em Engenharia de Controle e Automação.

**Orientador:** Dr. Leandro Freitas de Abreu

Betim

2022

## FICHA CATALOGRÁFICA

A657p Aquino, Vinicius Alefe de  
Projeto, Contrução e Análise de um Simulador Automobilístico /  
Vinicius Alefe de Aquino. – 2022.  
68 f.: il.

Trabalho de conclusão de curso (Bacharelado em  
Engenharia de Controle e Automação) - Instituto Federal de Educação,  
Ciência e Tecnologia de Minas Gerais, Câmpus Betim, 2022.

Orientação: prof. Dr. Leandro Freitas de Abreu

1. Simulador automobilístico. 2. Force feedback. 3. Instrumentação.  
4. Eletrônica. 5. Microcontroladores. I. Aquino, Vinicius Alefe de. II.  
Título.

Vinicius Alefe de Aquino

**PROJETO, CONSTRUÇÃO E ANÁLISE DE UM SIMULADOR  
AUTOMOBILÍSTICO**

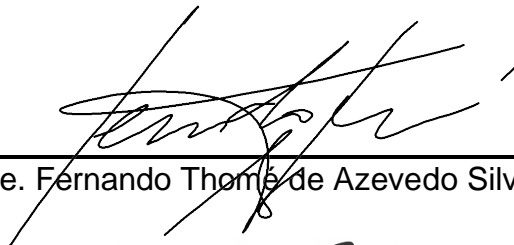
Trabalho de Conclusão de Curso apresentado à banca examinadora do curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – *Campus Betim*, como parte dos requisitos para obtenção do título de Bacharel em Engenharia de Controle e Automação.

Aprovado em 12 /12 /2022 pela banca examinadora:



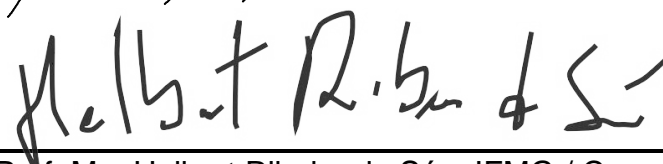
---

Prof. Dr. Leandro Freitas de Abreu (Orientador) – IFMG / Campus Betim



---

Prof. Me. Fernando Thomé de Azevedo Silva – IFMG / Campus Betim



---

Prof. Me. Helbert Ribeiro de Sá – IFMG / Campus Betim

## RESUMO

O presente trabalho tem como objetivo o projeto, desenvolvimento e construção de um simulador automobilístico – componente de hardware composto por volante e pedais capaz de simular, através da utilização de softwares específicos, a condução de um veículo automotor – que possa ser utilizado junto a softwares de simulação, possuindo um sistema ativo de *force feedback*. Foi construída uma base estrutural para o volante e outra base para os pedais, essas, responsáveis pela interface física de utilização do usuário e pelo suporte a implementação dos elementos elétricos e eletrônicos utilizados no projeto. O sistema eletroeletrônico foi constituído por elementos sensores de instrumentação, por elementos atuadores e de acionamentos elétricos, sendo todos elementos comandados por uma central eletrônica microprocessada. Na construção final do projeto, para a leitura posicional do volante, foi utilizado um encoder incremental e para a leitura de variação dos pedais foram utilizados sensores de efeito hall lineares. Na composição do sistema de atuação foi utilizado um motor de corrente contínua bem como um circuito de acionamento em ponte completa. Foram utilizados também técnicas de processamento de sinais e algoritmos de correção posicional para o tratamento de variáveis lidas dos sensores utilizados. Ao fim do projeto, o simulador construído pôde ser aplicado a softwares de simulação apresentado boa comunicação. Os resultados obtidos foram considerados satisfatórios tendo em vista também a aplicação bem sucedida dos elementos de instrumentação e atuadores do projeto, atendendo os objetivos gerais e específicos do projeto.

**Palavras-chave:** Simulador Automobilístico; Force Feedback; Microcontrolador; Eletrônica; Instrumentação.

## ABSTRACT

The objective of this work is the design, development and construction of an automobile simulator – a hardware component composed of a steering wheel and pedals capable of simulating, through the use of specific software, the driving of a motor vehicle – which can be used together with software simulation, having an active force-feedback system. A structural base was built for the steering wheel and another base for the pedals, which are responsible for the physical user interface and for supporting the implementation of the electrical and electronic elements used in the project. The electro-electronic system consisted of instrumentation sensing elements, actuator elements and electrical drives, all elements being controlled by a microprocessed electronic central. In the final construction of the project, for the positional reading of the steering wheel, an incremental encoder was used and, for the reading of the variation of the pedals, linear hall effect sensors were used. In the composition of the actuation system, a direct current motor was used as well as a full bridge drive circuit. Signal processing techniques and positional correction algorithms were also used for the treatment of variables read from the sensors used. At the end of the project, the built simulator could be applied to simulation software with good communication performance. The results obtained were rated good considering the successful application of the instrumentation elements and actuators of the project, reaching the general and specific objectives of the project.

**Key-Words:** Automobile Simulator; Force-Feedback; Microcontroller; Electronics; Instrumentation.

## LISTA DE ILUSTRAÇÕES

FIGURA 1 – MICROCONTROLADOR PIC18F4550 E SEUS PINOS DE ENTRADA E SAÍDA.....	13
FIGURA 2 – ATMEGA32U4 COM OS PINOS DE COMUNICAÇÃO USB DESTACADOS.....	13
FIGURA 3 – ILUSTRAÇÃO DA OCORRÊNCIA DO EFEITO HALL. ....	15
FIGURA 4 – CIRCUITO INTEGRADO (ESQUERDA) E DIAGRAMA DE BLOCOS DO SENSOR .....	16
FIGURA 5 – GRÁFICO DE TENSÃO X DENSIDADE DE FLUXO MAGNÉTICO DO SENSOR .....	16
FIGURA 6 – ESQUEMA BÁSICO DE CONSTRUÇÃO DE UM ENCODER ÓPTICO. ....	17
FIGURA 7 – ILUSTRAÇÃO DA ESTRUTURA BÁSICA DE UM ENCODER ÓPTICO INCREMENTAL... ..	18
FIGURA 8 – FORMAS DE ONDA EM RELAÇÃO AO ÂNGULO EMITIDAS POR UM ENCODER .....	18
FIGURA 9 – IMAGEM ILUSTRATIVA DE UM ENCODER INCREMENTAL.....	18
FIGURA 10 – DIAGRAMA SIMPLIFICADO DE UMA MÁQUINA CC. ....	19
FIGURA 11 – DIAGRAMA ESQUEMÁTICO DE UMA PONTE H. ....	20
FIGURA 12 – EXEMPLO DE MODULAÇÃO PWM DE UM SINAL SENOIDAL RETIFICADO .....	22
FIGURA 13 – SISTEMA DE FORÇA FEEDBACK POR CORREIA E POLIA ELEMENTAR.....	23
FIGURA 14 – DISPOSIÇÃO DOS PRINCIPAIS ELEMENTOS FÍSICOS DO SIMULADOR.....	26
FIGURA 15 – CIRCUITO ELÉTRICO GERAL DO SIMULADOR. ....	30
FIGURA 16 – CENTRAL ELETRÔNICA DO SIMULADOR.....	31
FIGURA 17 – MOTOR UTILIZADO NA VERSÃO FINAL DO PROJETO. ....	33
FIGURA 18 – MÓDULO IBT-2 E SEU POSICIONAMENTO NA BASE DO PROTÓTIPO. ....	34
FIGURA 19 – ESQUEMA ELÉTRICO DO SISTEMA DE COMANDO. ....	35
FIGURA 20 – ESQUEMA ELÉTRICO DO CIRCUITO DE LEITURA DO ENCODER.....	36
FIGURA 21 – ESQUEMA ELÉTRICO INTERNO DO ENCODER. ....	37
FIGURA 22 – ESQUEMA ELÉTRICO DO CIRCUITO DOS SENSORES AH49E.....	38
FIGURA 23 – ESQUEMA ELÉTRICO DAS CHAVES DAS BORBOLETAS DE CÂMBIO.....	39
FIGURA 24 – FLUXOGRAMA DE ETAPAS DO FIRMWARE DESENVOLVIDO. ....	40
FIGURA 25 – ILUSTRAÇÃO DO REGISTRADOR ADCSRA DO ATMEGA32U4. ....	42
FIGURA 26 – FLUXOGRAMA DAS OPERAÇÕES REALIZADAS PELAS INTERRUPÇÕES .....	43
FIGURA 27 – GRÁFICO DE LEITURA DO PEDAL DA EMBREAGEM COM MOVIMENTOS .....	45
FIGURA 28 – GRÁFICO DE MÓDULO (ACIMA) E FASE (ABAIXO) DO DIAGRAMA DE BODE .....	47
FIGURA 29 – FLUXOGRAMA DO MODO DE CONFIGURAÇÃO.....	49
FIGURA 30 – BASE ESTRUTURAL COM COMPONENTES MECÂNICOS E ELÉTRICOS .....	50

FIGURA 31 – ELEMENTOS DO EIXO PRINCIPAL.....	51
FIGURA 32 – ELEMENTOS DO SISTEMA DE TRANSMISSÃO.....	51
FIGURA 33 – VOLANTE DO SIMULADOR E CÂMBIO MONTADO.....	52
FIGURA 34 – BASE DO VOLANTE DO SIMULADOR. ....	53
FIGURA 35 – BASE DE PEDAIS.....	53
FIGURA 36 – CONJUNTO COMPLETO DO SIMULADOR. ....	54
FIGURA 37 – GRÁFICO DE LEITURA DO PEDAL DE EMBREAGEM E FILTRAGEM DO SINAL .....	55
FIGURA 38 – GRÁFICO DE LEITURA DO PEDAL E FILTRAGEM DO SINAL.....	56
FIGURA 39 – FFT DOS SINAIS DO PEDAL FILTRADO E NÃO FILTRADO.....	56
FIGURA 40 – DIGRAMAS DE BLOCOS DE MALHA ABERTA E FECHADA .....	58
FIGURA 41 – GRÁFICO DE ENTRADA DO SISTEMA DE ATUAÇÃO PARA UM ENSAIO .....	58
FIGURA 42 – RESPOSTA DO VOLANTE PARA ENTRADA SENOIDAL .....	59
FIGURA 43 – ATUAÇÃO DO SISTEMA EM MALHA FECHADA PARA UMA REFERÊNCIA EM 0.....	60
FIGURA 44 – RESPOSTA DO VOLANTE EM MALHA FECHADA. ....	60

## LISTA DE TABELAS

TABELA 1 – VALORES OBTIDOS EM CALIBRAÇÃO PARA A CONVERSÃO A/D DOS VALORES DE POSIÇÃO DOS PEDAIS DO SIMULADOR.....	55
--	----

## LISTA DE ABREVIATURAS E SIGLAS

AD – Analógico-digital

ADC – Analog-Digital Converter (Conversor analógico-digital)

EEPROM – Electrically-Erasable Programmable Read-Only Memory (Memória somente de leitura programável apagável eletricamente).

FIR – Finite Impulse Response (Resposta ao impulso finita)

HID – Human Interface Device (Dispositivo de interface humana)

IFMG – Instituto Federal de Minas Gerais

IGBT – Insulated Gate Bipolar Transistor (Transistor bipolar de porta isolada)

IIR – Infinite Impulse Response (Resposta ao impulso infinita)

LED – Light-Emitting Diode (Diodo emissor de luz)

MOSFET – Metal Oxide Semiconductor Field Effect Transistor (Transistor de efeito de campo de semicondutor óxido-metálico).

PWM – Pulse-Width Modulation (Modulação por largura de pulso)

FFT – Fast Fourier Transform (Transformação Rápida de Fourier)

## SUMÁRIO

1. Introdução .....	9
1.2. Objetivos .....	10
1.2.1 Objetivos gerais .....	10
1.2.2. Objetivos específicos .....	10
1.3. Justificativa .....	10
2. Referencial teórico .....	12
2.1. Microcontroladores e sistemas embarcados .....	12
2.2. Programação e HID .....	14
2.3. Sensores .....	14
2.3.1. Sensor de efeito Hall .....	14
2.3.2. Encoder Incremental .....	16
2.4. Atuador e Acionamentos .....	19
2.4.1. Motores CC .....	19
2.4.2. Circuito inversor em ponte completa .....	20
2.4.3. Modulação por largura de pulso .....	21
2.5. Sistemas de Force Feedback .....	22
2.6. Processamento digital de sinais .....	23
2.6.1. Filtros FIR .....	24
2.6.2. Filtros IIR .....	25
3. Metodologia .....	26
3.1. Hardware .....	26
3.1.1. Montagem Mecânica .....	27
3.1.1.1. Base estrutural .....	27
3.1.1.2. Eixo principal .....	27
3.1.1.3. Transmissão do <i>Force Feedback</i> .....	28

3.1.1.4. Volante .....	28
3.1.1.5. Base de pedais .....	29
3.1.2. Montagem Elétrica e Eletrônica.....	29
3.1.2.1. Circuito Geral .....	30
3.1.2.1. Microcontrolador e Central Eletrônica.....	31
3.1.2.2. Motor e Alimentação .....	32
3.1.2.3. Circuito de Acionamento .....	34
3.1.2.4. Encoder e Circuito Auxiliar .....	35
3.1.2.5. Sensores Hall.....	37
3.1.2.6. Controles do Volante .....	38
3.2. Software.....	39
3.2.1. Requisitos e Estrutura Geral de Programação .....	39
3.2.2. Interface e Comunicação USB .....	41
3.2.3. Aquisição e Parametrização de Dados dos Sensores .....	41
3.2.4. Tratamento de Variáveis e Processamento de Sinais .....	45
3.2.5. Atuação.....	48
3.2.6. Funções auxiliares .....	49
4. Resultados .....	50
4.1. Montagem física do simulador.....	50
4.1.1. Base Estrutural.....	50
4.1.2. Eixo Principal .....	50
4.1.3. Sistema de Transmissão .....	51
4.1.4. Volante e Câmbio.....	52
4.1.5. Panorama geral do simulador .....	52
4.2. Sensores .....	54
4.2.1. Leitura dos pedais e calibração .....	54
4.2.2. Encoder.....	57

4.3. Atuadores.....	57
4.4. Comunicação e aplicação do simulador .....	61
5. Conclusão .....	62
REFERÊNCIAS .....	63

## 1. INTRODUÇÃO

Com o avanço da tecnologia e popularização dos computadores e softwares de processamento gráfico, tornou-se cada vez mais comum a utilização desses recursos tecnológicos para a simulação de sistemas e processos. Esse fenômeno aplica-se a diversas áreas, tais como: circuitos elétricos, processos industriais, sistemas mecânicos e hidropneumáticos, entre outros. Dessa forma, a aplicação de simuladores beneficia muitas áreas no contexto global, especialmente as que demandam altos investimentos.

No contexto automobilístico, simuladores podem ser mais especificamente interpretados por estruturas físicas capazes de, através da comunicação com um computador, simular a condução de veículos automotores permitindo que um usuário comande um veículo virtual e receba informações sensoriais realísticas relacionadas à condução do veículo.

De maneira geral, simuladores automobilísticos são constituídos de um hardware composto por uma base estrutural com um volante e pedais, itens básicos para a condução de um veículo virtual. Atualmente, existem diversos fabricantes de simuladores automobilísticos – Logitech, Thrustmaster e Fanatec são alguns exemplos.

Os volantes e pedais presentes nos simuladores são caracterizados por realizarem uma comunicação direta com computador permitindo que receba os comandos executados pelo usuário e, ao mesmo tempo, retorne para o usuário o *force feedback* da simulação, que é a força reativa do veículo virtual que pode ser sentida pelo usuário no eixo de direção do volante.

Em se tratando da aplicação, esses simuladores são tipicamente utilizados junto a softwares de simulação que são os responsáveis por criar um ambiente virtual de simulação. Nesse contexto, o software utilizado é responsável não apenas por receber os comandos executados pelo usuário, mas também aplicá-los computando todo o processo físico simulado dentro do ambiente de simulação e retornar para o hardware (parte física do simulador) os sinais de *force feedback* computados pela simulação. Existem hoje muitos softwares de simulação disponíveis ao público no

mercado, como por exemplo: iRacing, Automobilista 2, Assetto Corsa, R-Factor 2, entre outros.

## **1.2. Objetivos**

### **1.2.1 Objetivos gerais**

Este trabalho visa o desenvolvimento de um simulador automobilístico para utilização em softwares de simulação. Objetiva-se projetar, construir e implementar, a partir de uma base mecânica elementar, os circuitos eletrônicos de sensores e atuadores, bem como a programação dos sistemas microprocessados utilizados.

### **1.2.2. Objetivos específicos**

- Construir uma estrutura mecânica elementar para o hardware;
- Elaborar o projeto eletrônico de leitura angular do volante;
- Desenvolver o projeto eletrônico de leitura dos pedais;
- Fazer ou implementar um circuito de potência para o comando do atuador;
- Unir todos os circuitos projetados junto a uma central de processamento;
- Programar o sistema microprocessado para leitura de sensores e comando do atuador;
- Programar o sistema microprocessado para a comunicação com o computador;
- Avaliar o desempenho dinâmico do sistema de atuação e instrumentação;
- Avaliar o desempenho geral do projeto em softwares de simulação.

## **1.3. Justificativa**

A grande vantagem dos simuladores automobilísticos vem da possibilidade de que sejam simulados processos que no mundo real podem ser difíceis e custosos de se realizar. Dessa forma, o desenvolvimento de protótipos que possibilitem o

desenvolvimento do setor automobilístico ao amenizar os custos de desenvolvimento, bem como dificuldades técnicas, é bem-vindo.

Pela ótica mercadológica, o desenvolvimento desse tipo de estudo pode ter grande proveito tendo em vista sua contribuição para a possível introdução de novos produtos no mercado, bem como a possibilidade de introdução de novos conhecimentos e técnicas de engenharia.

O desenvolvimento de protótipos de simuladores traz também benefícios que abrangem diversas áreas da Engenharia de Controle e Automação por permitir que sejam desenvolvidas e aplicadas técnicas de circuitos elétricos e eletrônicos, processamento de sinais, comunicação digital, entre outros conteúdos presentes no curso.

## **2. REFERENCIAL TEÓRICO**

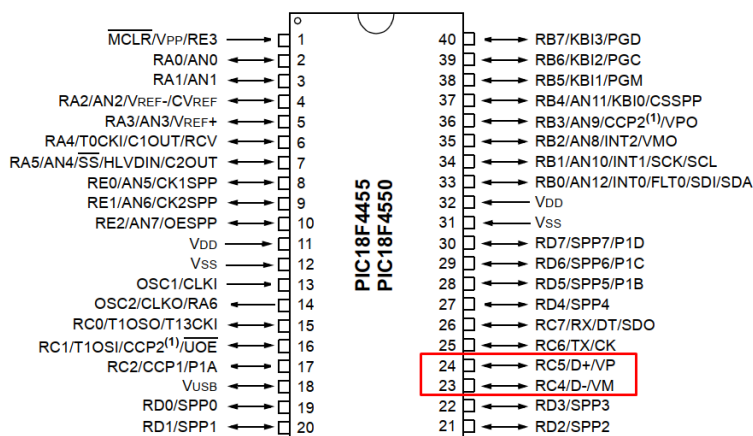
O presente trabalho consiste no desenvolvimento de um protótipo e, por isso, se vale da utilização de técnicas de engenharia bem como elementos de instrumentação específicos. Portanto, alguns desses elementos serão apresentados do ponto de vista teórico permitindo uma melhor compreensão de sua aplicação no projeto.

### **2.1. Microcontroladores e sistemas embarcados**

Um dos elementos centrais no desenvolvimento do projeto – principal componente da central eletrônica – é o microcontrolador. Microcontroladores são componentes da eletrônica digital amplamente utilizados para a automação e controle nos mais variados contextos (pessoal, industrial, residencial, entre outros). Gimenez (2002, p. 4) define os microcontroladores: “dispositivo semicondutor em forma de CI, que integra todas as partes básicas de um microcomputador.” Sendo uma espécie de microcomputador, um microcontrolador tem a capacidade de, através de um firmware (programa que contém instruções lógicas) desenvolvido por um projetista, comandar o seu próprio hardware como interfaces de entrada, saída e comunicação de forma a realizar funções específicas dentro de um projeto.

Tendo em vista uma aplicação em simuladores, uma forma de um equipamento de interface humana se comunicar com o computador diretamente e com um software de simulação é através da utilização de microcontroladores que possuam uma interface de comunicação usb interna. Atualmente, há uma série de microcontroladores disponíveis no mercado que possuem essa característica. Um exemplo disso é o microcontrolador da Microchip PIC18F4550 que possui um canal de comunicação USB disponível nos pinos 23 e 24 do CI (circuito integrado), respectivamente.

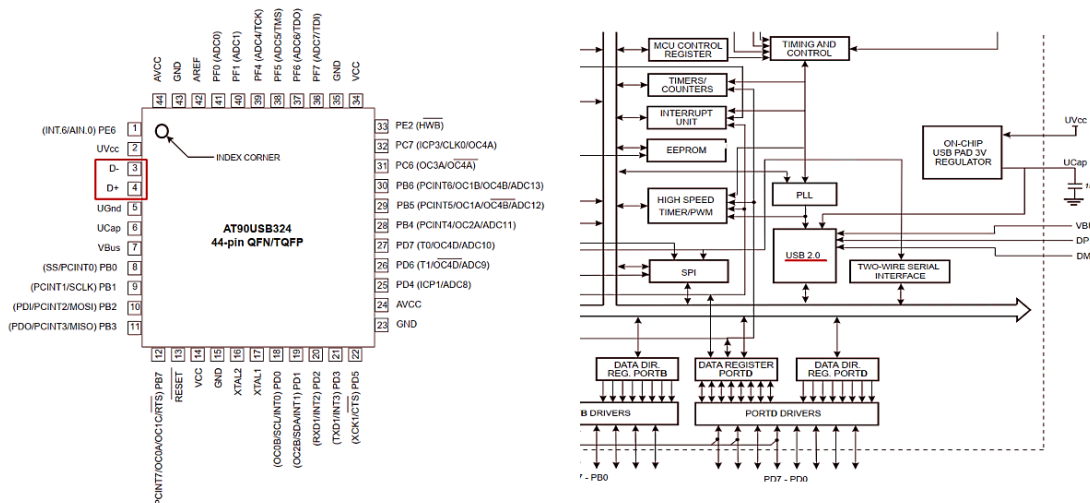
Figura 1 – Microcontrolador PIC18F4455 e seus pinos de entrada e saída.



Fonte: adaptado do datasheet. (MICROCHIP, Technology Inc.)

Com os sistemas embarcados em alta, um exemplo mais popular de microcontrolador capaz de realizar uma comunicação direta via interface USB 2.0 é o Atmega32u4 da Atmel. Esse microcontrolador está presente nos populares sistemas embarcados Arduino Leonardo e Arduino Micro que, diferentemente de grande parte dos embarcados da série Arduino, não utilizam chips intermediários de conversão USB-UART para a comunicação com o microcontrolador, mas possuem a interface USB 2.0 diretamente disposta no Atmega32u4.

Figura 2 – Atmega32u4 com os pinos de comunicação USB destacados em vermelho (à esquerda) e parte do diagrama de blocos do microcontrolador com sua interface USB sublinhada em vermelho (à direita).



Fonte: adaptado do datasheet. (ATMEL, Corporation).

## **2.2. Programação e HID**

Um importante aspecto a ser trabalhado junto ao microcontrolador é a estrutura de programação do mesmo. A programação de um microcontrolador como o Atmega32u4 pode ser feita para que o mesmo funcione e seja reconhecido pelo sistema operacional do computador como um simulador automobilístico. Essa programação pode ser realizada por padrão na linguagem C++, entretanto, a comunicação usb deve ser programada para que o computador reconheça o dispositivo conectado a interface USB como um dispositivo de classe HID (Human Interface Device – dispositivo de interface humana).

A classe HID é uma descrição atribuída a determinados dispositivos que operam através do protocolo USB que os identifica como dispositivos de entrada de informações instantâneas para o uso prático em tempo real de aplicações e tarefas. Mouses, teclados e joysticks são alguns exemplos de periféricos classificados como HID.

Existem muitas bibliotecas de programação disponíveis e de fácil acesso que facilitam a implementação de dispositivos HID em microcontroladores. Por já implementarem classes e objetos parametrizados com as corretas descrições dos HID reports (mensagens estruturadas que contém as informações organizadas segundo o protocolo HID), as bibliotecas tornam o trabalho de programação reduzido à atribuição e aquisição de valores entre as variáveis do firmware.

## **2.3. Sensores**

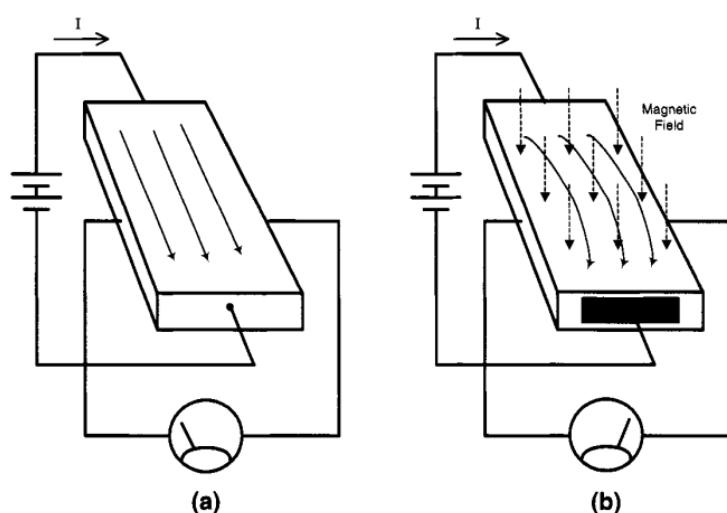
Durante o desenvolvimento do projeto foram utilizados dois principais elementos de instrumentação para a aquisição de grandezas físicas: sensores de efeito hall e encoder incremental.

### **2.3.1. Sensor de efeito Hall**

Sensores de efeito hall são transdutores capazes de converter a intensidade de um campo magnético em um ponto de medição em tensão elétrica a partir do efeito hall.

O efeito hall é um fenômeno descoberto por Edwin Herbert Hall em 1879 e é caracterizado pela ocorrência de uma diferença de potencial elétrico em um condutor de corrente elétrica quando o mesmo é submetido a um campo magnético. A tensão elétrica gerada é particularmente definida por ser perpendicular à direção da corrente elétrica, bem como ao campo magnético ao qual o condutor está submetido. Esse fenômeno ocorre devido ao deslocamento dos portadores de carga em movimento devido à interação com o campo magnético. A Figura 3 ilustra a ocorrência do efeito hall em um condutor.

Figura 3 – Ilustração da ocorrência do efeito hall. (a) condutor sem campo magnético aplicado e (b) condutor sujeito a um campo magnético e uma tensão hall medida pelo voltímetro.



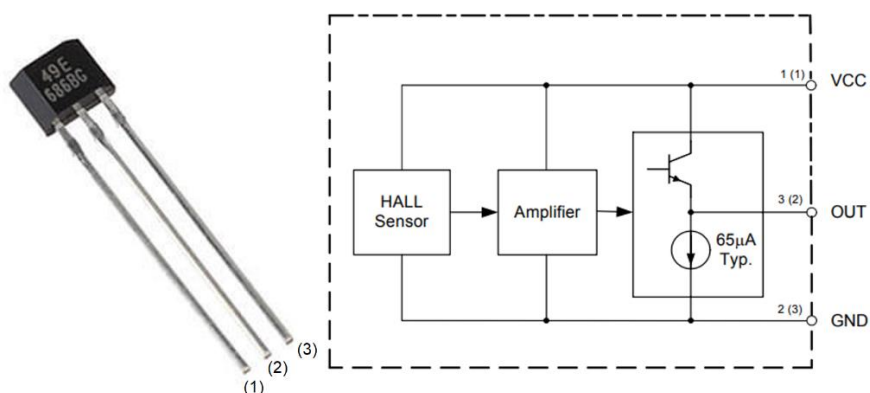
Fonte: Ramsden, 2006. (p.1)

Tendo em vista seu princípio funcionamento, os sensores de efeito hall podem ser baseados em diferentes tipos de topologia podendo ser analógicos/lineares ou digitais.

Sensores hall analógicos/lineares são comumente implementados através de semicondutores, permitindo assim a implementação dos mesmos em circuitos integrados tipicamente utilizados em circuitos eletrônicos. Um exemplo desse tipo de sensor é o AH49E – circuito integrado que implementa um sensor hall permitindo o condicionamento linear do sinal de saída de acordo com a intensidade e polaridade do campo magnético aplicado. A Figura 4 mostra estrutura interna em diagrama de blocos do AH49E que implementa não apenas o transdutor de efeito hall, mas também um circuito amplificador e condicionador do sinal de saída, permitindo aplicações

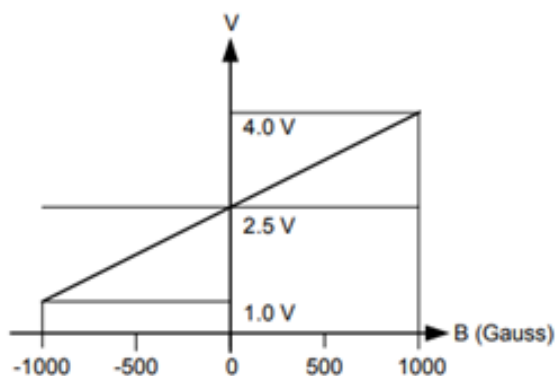
diretas do componente através dos seus pinos de alimentação e saída, dispensando a utilização de circuitos auxiliares.

Figura 4 – Circuito integrado (esquerda) e diagrama de blocos do sensor AH49E (direita).



Fonte: adaptado do datasheet. (BCD Semiconductor, 2008).

Figura 5 – Gráfico de tensão x densidade de fluxo magnético do sensor AH49E.



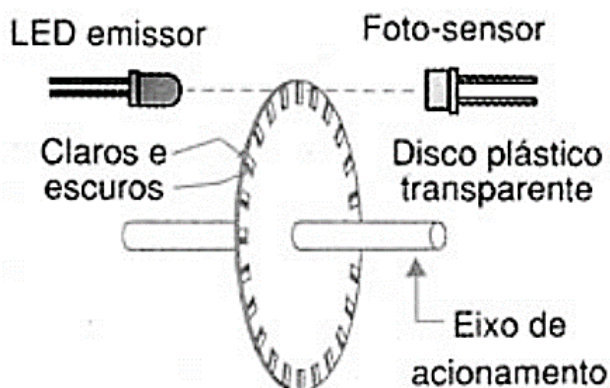
Fonte: adaptado do datasheet. (BCD Semiconductor, 2008).

### 2.3.2. Encoder Incremental

O segundo tipo de sensor utilizado no presente trabalho é o encoder rotativo incremental. Esse instrumento tem a função de informar indiretamente a variação angular em seu próprio eixo através de emissão de pulsos elétricos que correspondem (em quantidade) ao módulo da variação do ângulo de rotação axial.

Basicamente, um encoder elementar rotativo pode ser construído conforme mostra a Figura 6.

Figura 6 – Esquema básico de construção de um encoder óptico.



Fonte: [www.newtoncbraga.com.br](http://www.newtoncbraga.com.br)

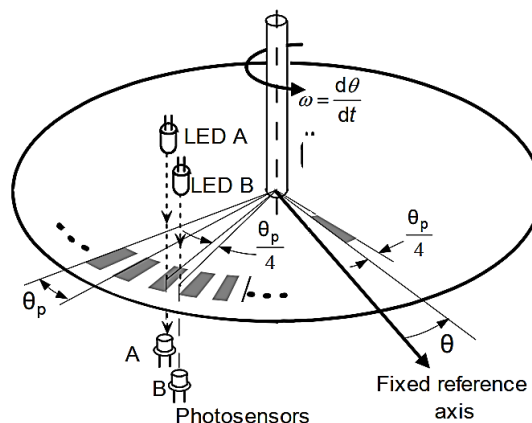
Baseado nessa estrutura ilustrada, no momento em que o eixo de acionamento do encoder óptico é rotacionado, pulsos são gerados no foto-sensor de acordo com a ocorrência do bloqueio da luz do led emissor feita pelo disco plástico.

Dessa forma, com o eixo do encoder rotativo acoplado a um eixo em que se deseja medir a velocidade de rotação, pode-se medir, através da frequência dos pulsos, o módulo da velocidade angular aproximada do eixo.

Na necessidade de medir-se a velocidade angular não apenas em módulo, mas também em sentido, utiliza-se um tipo de encoder rotativo conhecido como encoder incremental. A estrutura de um encoder incremental básico é bem parecida com a estrutura de um encoder relativo rotacional simples, porém, há a adição de mais um par LED e foto-sensor que opera, em relação aos claros e escuros destacados na imagem, com um desfasamento de  $90^\circ$  em relação a um ciclo completo de pulso.

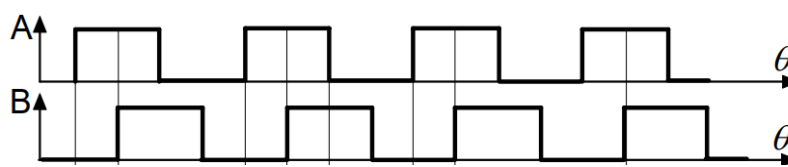
Na Figura 8 é possível observar a forma de onda do Foto-sensor A que representa a forma de onda padrão gerada por um encoder relativo simples. A forma de onda gerada pelo foto-sensor B é a adição do encoder incremental que permite que, através dele seja obtido o sentido de rotação.

Figura 7 – Ilustração da estrutura básica de um encoder óptico incremental.



Fonte: adaptado de Incze, Szabó, Imecs, 2009 (p. 98)

Figura 8 – Formas de onda em relação ao ângulo emitidas por um encoder rotativo incremental. A e B representam os sinais de saída dos foto-sensores e  $\theta$  o ângulo rotacional do encoder.



Fonte: adaptado de Incze, Szabó, Imecs, 2009 (p. 99)

Através da monitoração dos pulsos dos foto-sensores A e B – realizada por um microcontrolador, por exemplo – é possível determinar a cada borda de subida de um dos sensores o sentido da rotação utilizando como condição determinante o estado atual do outro sensor.

Figura 9 – Imagem ilustrativa de um encoder incremental.



Fonte: [www.perautomacao.com.br](http://www.perautomacao.com.br)

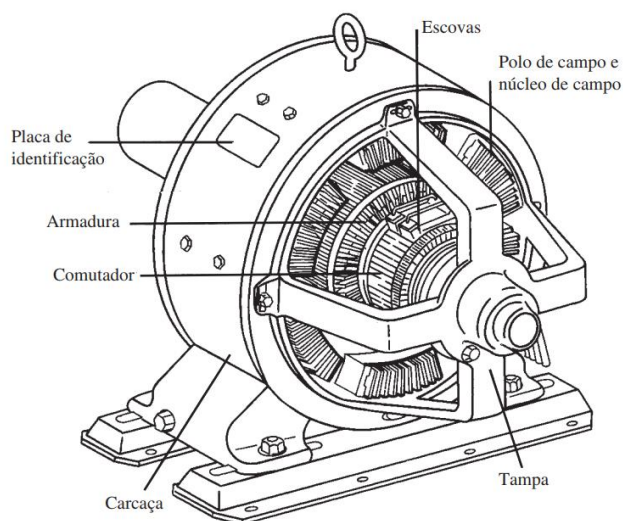
## 2.4. Atuador e Acionamentos

O sistema de atuação implementado no projeto é desenvolvido em torno da aplicação de um motor CC (corrente contínua) para a reprodução dos efeitos de *force feedback*, bem como de seus circuitos e mecanismos de acionamento.

### 2.4.1. Motores CC

Um motor CC tem sua estrutura básica formada por: armadura (estrutura com conjunto de bobinas responsáveis por gerar o campo magnético necessário para a interação magnética interna do motor), comutador e escovas (responsáveis pela comutação das bobinas da armadura), e polos de campo (responsáveis por gerar o fluxo magnético principal da máquina).

Figura 10 – Diagrama simplificado de uma máquina CC.



Fonte: Chapman, 2013 (p. 450)

Para apenas reproduzir efeitos de *force feedback*, um motor CC (de ímã permanente, por exemplo) tende a trabalhar a maior parte do tempo em um regime de rotor bloqueado. Dessa forma, considerando um fluxo magnético  $\phi$  constante, o torque induzido pelo motor em um determinado instante de tempo pode ser aproximado pela Equação 1, em que  $\tau_{ind}$  denota o torque induzido pelo motor (dado em  $N \cdot m$ ),  $K$  é

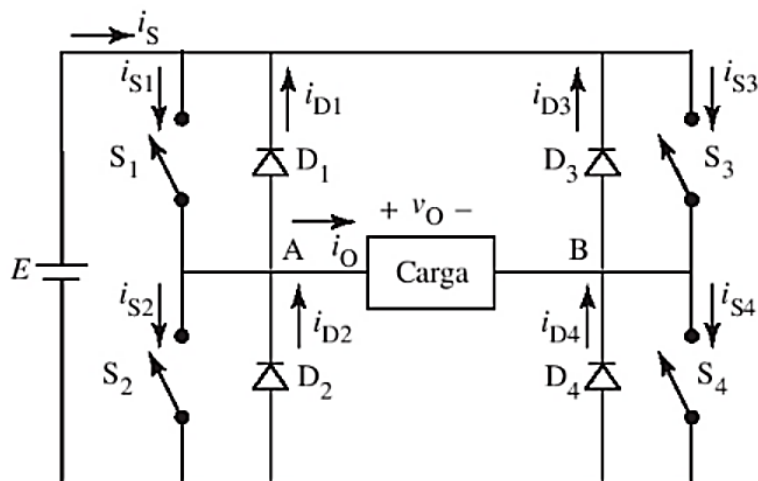
uma constante de proporcionalidade do torque (dado em  $N \cdot m \cdot A^{-1}$ ) e  $I_A$  (dada em A) é a corrente na armadura do motor.

$$\tau_{ind} = K \cdot I_A \quad (1)$$

### 2.4.2. Circuito inversor em ponte completa

Para o acionamento de um motor CC e o controle de torque em ambos sentidos ao longo do tempo, faz-se necessário o uso de um circuito eletrônico de potência capaz de realizar tal função. Um circuito típico utilizado com essa função é o circuito Inversor de tensão em ponte completa (também conhecido como ponte H).

Figura 11 – Diagrama esquemático de uma ponte H.



Fonte: Ahmed, 2000 (p. 357)

No circuito esquematizado na Figura 11 as chaves indicadas por  $S_1$ ,  $S_2$ ,  $S_3$  e  $S_4$  são as componentes comutadas para que ocorra a devida inversão ou não de tensão sobre a carga.

O acionamento das chaves ao longo do tempo sempre ocorre de três maneiras: apenas chaves  $S_1$  e  $S_4$  fechadas (e as demais abertas) resultando em uma tensão  $+E$  na carga, apenas as chaves  $S_3$  e  $S_4$  fechadas resultando em uma tensão  $-E$  na carga, e todas as chaves abertas não aplicando tensão na carga. Para garantir a tensão 0 na carga, comumente é aplicada a condição em que apenas as chaves  $S_2$

e  $S_4$  são acionadas, evitando flutuações de tensão e garantindo uma diferença de potencial 0 entre os terminais da carga.

Comumente, no lugar das chaves indicadas no circuito da Figura 11 são utilizados transistores de potência (MOSFETs ou IGBTs, por exemplo).

Para a utilização de circuitos inversores em ponte completa, comumente, é necessário não apenas a implementação de chaves, mas também a utilização de circuitos auxiliares capazes de condicionar os sinais presentes para o acionamento dos transistores utilizados, bem como para a proteção do circuito geral.

Em se tratando do condicionamento dos sinais, em muitos casos – especialmente quando a tensão de alimentação  $E$  é superior à tensão de trabalho do circuito de comando – são necessários circuitos elevadores de tensão ou circuitos de bootstrap para garantir que o sinal aplicado à base (ou gate) dos transistores tenha um potencial elétrico superior ao do emissor (ou source).

No caso da proteção dos circuitos, faz-se necessária a aplicação de diodos de chaveamento rápido em anti-paralelo com as chaves ou transistores a fim de suprimir as altas tensões reversas que podem surgir devido a auto indução causada pelo chaveamento de cargas indutivas. Também são tipicamente aplicados circuitos snubber, em que resistores e capacitores são posicionados no circuito a fim de reduzir a intensidade das espúrias de alta frequência presentes no sinal de chaveamento dos transistores, tipicamente causadas por indutâncias parasitas presentes nos circuitos.

### **2.4.3. Modulação por largura de pulso**

Um circuito de ponte H naturalmente só consegue alimentar uma carga com níveis de tensão, em módulo, iguais ao da fonte de alimentação ou zero. Sendo assim, para obter-se uma variação de tensão que transite entre os limites extremos de tensão sobre a carga, é possível a utilização da modulação por largura de pulso (PWM). Essa técnica consiste em chavear a carga ciclicamente através de pulsos de forma que a tensão média sobre a carga dentro de um ciclo seja próxima ou igual à tensão desejada.

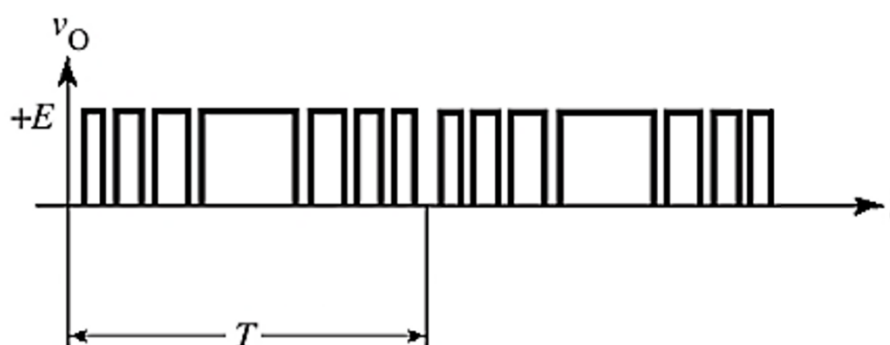
A tensão média dentro de um ciclo de sinal PWM é definida pelo duty cycle que é o tempo percentual que o pulso permanece em nível lógico alto em relação ao

tempo total do ciclo PWM (período). A Equação 2 define a tensão média na carga  $V_{om}$  sendo  $E$  a tensão da fonte constante e  $\alpha$  o duty cycle percentual de modulação.

$$V_{om} = \frac{E \cdot \alpha}{100} \quad (2)$$

A modulação PWM pode ser aplicada em uma ponte H através do chaveamento dos transistores que controlam a carga. A Figura 12 exemplifica uma modulação PWM senoidal retificada aplicada sobre o mesmo circuito de ponte H descrito na Seção 2.4.2 em que  $V_o$  é a tensão sobre a carga.

Figura 12 – Exemplo de modulação PWM de um sinal senoidal retificado de período  $T$ .



Fonte: Ahmed, 2000 (p. 369) <sup>10</sup>

Para o realizar uma modulação PWM necessita-se de um circuito de controle. Atualmente grande parte dos microcontroladores disponíveis no mercado já possuem circuitos de controle PWM internos que podem ser acionados conforme a programação realizada. O Atmega32u4 é um exemplo de circuito integrado que possui esse recurso.

## 2.5. Sistemas de Force Feedback

Os sistemas de atuação e comandos aplicados nesse trabalho têm como objetivo primário a implementação de um sistema de force feedback para o volante do simulador.

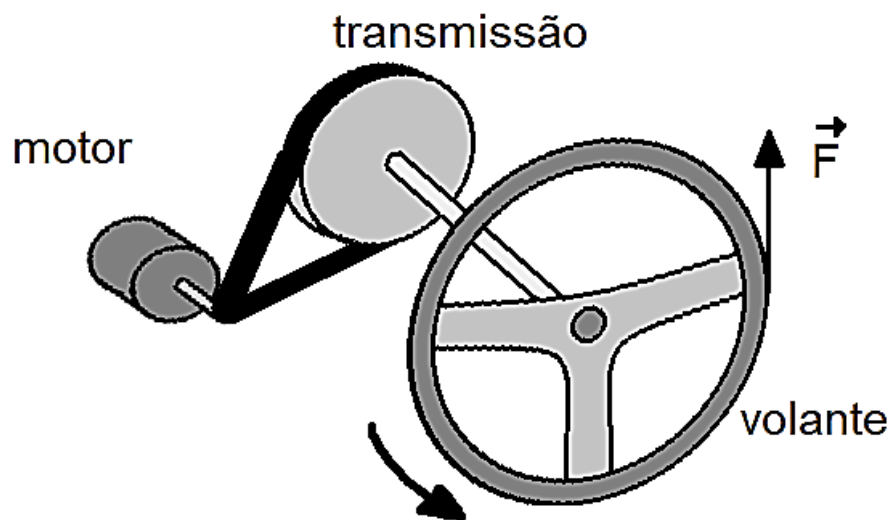
Em se tratando de um simulador automobilístico, um sistema de force feedback é o sistema responsável por reproduzir forças reativas de um veículo simulado sobre o eixo do volante. Em outras palavras, as forças que ocorrem sobre

um veículo devido ao atrito das rodas com o solo e do conjunto mecânico, oscilações do terreno, forças de aceleração lateral, entre outros fenômenos, são transmitidas no mundo real para o volante do veículo, mesmo que indiretamente através do conjunto mecânico de direção. No mundo virtual, aplicando-se em simuladores, essas forças são simuladas através do sistema de force feedback.

Dessa forma, um sistema de force feedback pode ser construído através do acoplamento de um motor – atuador capaz de reproduzir as forças que devem ser simuladas – ao eixo de direção.

Atualmente, nos simuladores disponíveis no mercado, três tipos de sistemas de force feedback são os mais utilizados: transmissão por engrenagens, transmissão por correia e sistemas de direct drive. Os dois primeiros tipos caracterizam o acoplamento indireto do motor ao eixo de direção através de um conjunto de engrenagens ou por um conjunto de correia e polias. Os sistemas de direct drive, por sua vez, possuem o motor acoplado diretamente ao eixo de direção. A Figura 13 ilustra um sistema de force feedback elementar com transmissão por correia.

Figura 13 – Sistema de force feedback por correia e polia elementar.



Fonte: Próprio autor.

## 2.6. Processamento digital de sinais

Alguns elementos relacionados ao processamento digital de sinais foram trabalhados durante o presente trabalho visando adequações dos sinais obtidos a

partir da leitura e conversão digital dos sinais analógicos fornecidos pelos elementos sensores.

A aplicação de processamento de sinais na aquisição e tratamento de valores em projetos com sistemas microprocessados é muito comum por sua simplicidade de implementação. Filtros FIR e IIR, por exemplo, podem ser implementados facilmente através de apenas uma linha de código que descreva uma equação de diferenças correspondente ao filtro projetado.

Dessa forma, no âmbito de processamento e tratamento de sinais, os filtros FIR e IIR foram as principais ferramentas utilizadas durante o desenvolvimento do projeto e serão melhor apresentados nas seções 2.5.1 e 2.5.2.

### 2.6.1. Filtros FIR

Filtros FIR (Finite Impulse Response – Filtro de resposta ao impulso finita) são filtros digitais elementares e são caracterizados, por definição, por apresentarem uma resposta ao impulso finita, isto é, a resposta ao impulso tende a zero em um intervalo de tempo finito.

Filtros FIR são estruturados tendo em vista sua característica não recursiva – a saída em um instante de tempo  $k$  depende exclusivamente dos valores de entrada em  $k$  ou instantes passados de  $k$ . A estrutura básica de um filtro FIR é dada pela equação de diferenças (3) ou pela função de transferência (4).

$$y[k] = \sum_{n=0}^{N-1} b_n \cdot x[k - n] \quad (3)$$

$$D(z) = \frac{Y(z)}{X(z)} = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_N z^{-N} \quad (4)$$

A natureza de um filtro FIR é completamente definida pelos valores dos coeficientes  $b_n$  dispostos nas equações 3 e 4 e, dessa forma, eles podem ser passa-baixa, passa-alta, passa-faixa ou rejeita-faixa, sendo que a ordem do filtro é definida pela unidade de maior atraso temporal nos elementos de entrada  $N$ .

Outra característica importante dos filtros FIR é a sua inerente estabilidade por não haver realimentação da saída para a entrada, garantindo a ausência de polos de módulo superior a 1 na dinâmica do filtro.

### 2.6.2. Filtros IIR

Também conhecidos como filtros autorregressivos ou recursivos, os filtros IIR (infinite impulse response – Filtro de resposta ao impulso infinita) são um segundo tipo de filtro amplamente utilizados em processamento de sinais. Diferentemente dos filtros FIR, os filtros IIR, por sua característica recursiva, apresentam uma resposta ao impulso de duração infinita com valores de saída tendendo a 0 apenas em tempos que tendem ao infinito.

A estrutura de um Filtro IIR é caracterizada por ser autorregressiva – a saída em um instante de tempo  $k$  depende de valores de entrada em  $k$  ou instantes passados de  $k$ , como também de valores da saída em instantes passados de  $k$ . Dessa forma os filtros IIR podem ser modelados de acordo com as equações 5 e 6.

$$y[k] = \sum_{n=1}^N a_n \cdot y[k - n] + \sum_{m=0}^M b_m \cdot x[k - m] \quad (5)$$

$$D(z) = \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}}{1 - a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}} \quad (6)$$

Assim como no caso dos Filtros FIR, a natureza do filtro (passa-baixas, passa-altas e afins) são definidas pelos coeficientes  $a_n$  e  $b_m$  presentes no numerador e denominador da função de transferência do sistema, sendo a ordem do sistema sendo definida pelo coeficiente  $N$ .

### 3. METODOLOGIA

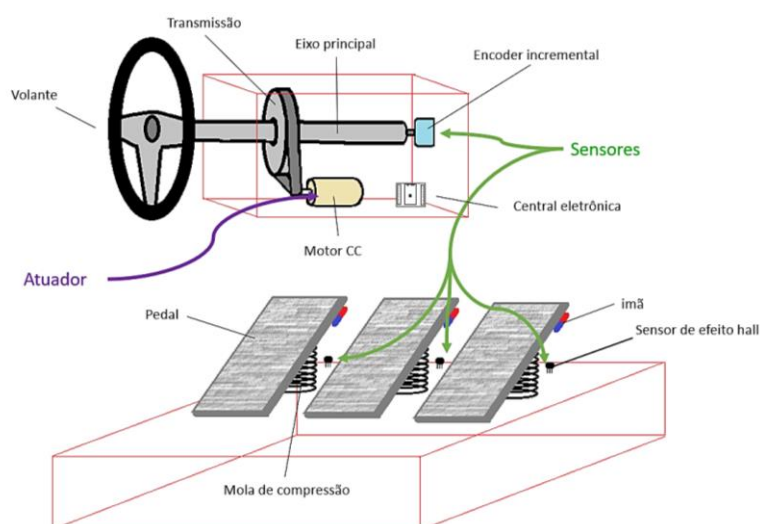
O presente trabalho é elaborado em dois principais temas: O desenvolvimento físico do projeto (*hardware*) e o desenvolvimento computacional (*software*). Por conseguinte, esta Seção do trabalho será subdividida em tópicos derivados desse modelo metodológico.

Para uma melhor compreensão, deve-se considerar que o trabalho consistiu em um desenvolvimento progressivo, isto é, o resultado final pode não ser exatamente igual ao projetado, passando por adaptações e mudanças ao longo do desenvolvimento. Muitas dessas alterações de características estão relacionadas às melhorias realizadas durante o processo de desenvolvimento e soluções de problemas encontrados.

#### 3.1. Hardware

O hardware do projeto consiste em toda parte física que compõe o simulador, isso inclui tanto partes mecânica quanto partes elétricas e afins. Visando simplificar os processos e flexibilizar a adaptação do projeto às necessidades técnicas que surgiram ao longo do projeto, toda a montagem física do projeto foi feita manualmente pelo próprio autor.

Figura 14 – Disposição dos principais elementos físicos do simulador.



Fonte: Próprio autor.

### **3.1.1. Montagem Mecânica**

A montagem mecânica do projeto, apesar de ser um dos tópicos elementares da Seção de hardware, serviu apenas como apoio para que pudessem ser trabalhados os demais tópicos relacionados às disciplinas estudadas durante o curso de engenharia de controle e automação. Assim, essa Seção visa apenas dar um panorama geral da montagem e permitir uma compreensão de como foram elaboradas as ideias postas em prática durante o desenvolvimento, não se atendo a detalhes técnicos específicos da área de mecânica.

#### **3.1.1.1. Base estrutural**

Todo o hardware correspondente a parte do volante foi montado sobre uma base que permite tanto a estruturação das partes móveis mecânicas, quanto o posicionamento dos componentes eletrônicos. A base estrutural foi montada a partir de chapas de madeira compensada com espessura de 15 mm que foram cortadas e unidas utilizando parafusos para madeira. Sobre a base de madeira foi montada a estrutura de sustentação do eixo principal do volante bem como os apêndices mecânicos e elétricos: motor, polias, suportes dos componentes eletrônicos entre outros. Foram também fixados à estrutura de madeira pequenas barras de metal conectando as extremidades, apenas para fornecer rigidez estrutural.

#### **3.1.1.2. Eixo principal**

Acoplado a base estrutural está o eixo principal do volante composto por um tubo metálico de aproximadamente 20 mm de diâmetro. Essa, após a base estrutural, pode ser considerada a principal parte mecânica do simulador, visto que nele estão conectados grande partes dos elementos atuadores e partes de instrumentação que compõem o conjunto.

Em uma das extremidades do eixo principal, encontra-se o volante, responsável pelos comandos manuais realizados pelo operador do simulador. No

meio do eixo é posicionada uma polia estriada de 130 mm de diâmetro que pertence ao conjunto mecânico do sistema de *force feedback*. Próximo a outra extremidade encontra-se uma engrenagem cilíndrica de dentes retos de 37 mm de diâmetro responsável por conectar o eixo do volante ao encoder utilizado, através de uma engrenagem secundária. Na extremidade final do eixo principal foi posicionado um conector rotativo elétrico de 6 vias, responsável por conectar eletricamente fios que passam internamente ao eixo e conectam-se aos controles instalados no volante, evitando que os cabos elétricos conectados ao eixo quebrem em decorrência da rotação do mesmo.

#### **3.1.1.3. Transmissão do *Force Feedback***

O motor utilizado para o sistema de *force feedback* é conectado ao eixo principal através de uma correia estriada. A polia acoplada ao eixo do motor é de 20 mm de diâmetro e relacionada à polia do eixo principal do volante (de 130 mm), tem-se uma relação de transmissão de 6,5:1. Dessa forma, o torque exercido pelo motor sobre o eixo principal é multiplicado 6,5 vezes pela relação de transmissão. Isso é particularmente útil pois permite que a demanda de carga do motor seja reduzida viabilizando uma redução do sistema de alimentação e atuação elétrica sobre o motor, tema que será mais detalhado na Seção 3.1.2.2.

Junto ao sistema de transmissão foram utilizadas duas polias auxiliares apenas com o objetivo de tensionar a correia do sistema de transmissão. Visto que não se encontrou disponível uma correia de tamanho mais adequado ao projeto durante o desenvolvimento.

Os demais componentes presentes na base pertencem ao sistema elétrico e eletrônico e serão discutidos individualmente ao decorrer do trabalho.

#### **3.1.1.4. Volante**

Semelhantemente aos demais componentes, o volante do protótipo foi feito de maneira artesanal tem sua maior parte construída em madeira. Atrás do volante

foram posicionadas chaves de fim de curso responsáveis por detectar a mudança de estado das borboletas de troca de marcha.

As borboletas de troca de marcha foram feitas utilizando-se chapas de metal que, conectadas a articulações metálicas acopladas ao eixo do volante, conduzem o movimento de acionamento das chaves de fim de curso. O retorno das borboletas de câmbio foi feito pelo posicionamento de pequenas molas de compressão posicionadas entre a estrutura do volante e a estrutura das borboletas.

### **3.1.1.5. Base de pedais**

Além da base com o volante, foi desenvolvido uma base externa que comporta o sistema de pedais que são utilizados para o controle do acelerador, freio e embreagem.

O sistema de pedais é também construído majoritariamente em madeira possuindo apenas algumas partes em metal como os mecanismos que fazem a rotação dos pedais. Fundamentalmente, cada pedal é composto apenas por uma chapa de madeira, que é conectada a base através de um mecanismo metálico rotativo. Os pedais são limitados em movimentação pela própria estrutura da base. Foi colocada também entre a base estrutural e cada pedal uma mola de compressão para gerar a força de compressão dos pedais. Acima de cada pedal foi colocado uma chapa de borracha para melhorar a aderência sobre os pedais e obter uma melhor usabilidade dos mesmos.

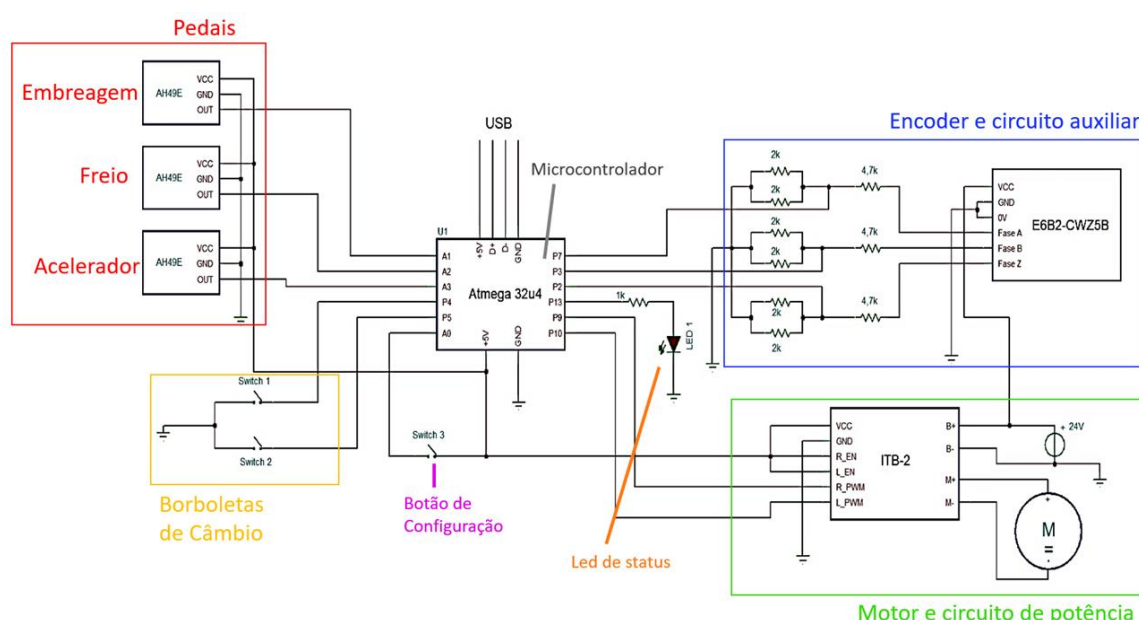
### **3.1.2. Montagem Elétrica e Eletrônica**

A montagem Elétrica e Eletrônica do projeto consiste na implementação dos sensores e atuadores, circuitos de acionamento e circuitos de condicionamento de sinais que são trabalhados pelos sistemas microprocessados. Todos os elementos eletrônicos são conectados a uma central eletrônica que é a responsável direta pela interface simulador-computador. Cada um dos tópicos que compõem a montagem elétrica e eletrônica do simulador serão abordados nas seções seguintes.

### 3.1.2.1. Circuito Geral

Com o intuito de facilitar a compreensão geral do circuito elétrico do simulador, os circuitos foram subdivididos em diferentes partes, cada uma correspondente a um determinado sistema do simulador. A Figura 15 dispõe a conexão elétrica de todos os elementos utilizados no circuito do simulador, dando um panorama geral da montagem e ilustrando as subdivisões do circuito.

Figura 15 – Circuito Elétrico Geral do Simulador.



Fonte: Próprio autor.

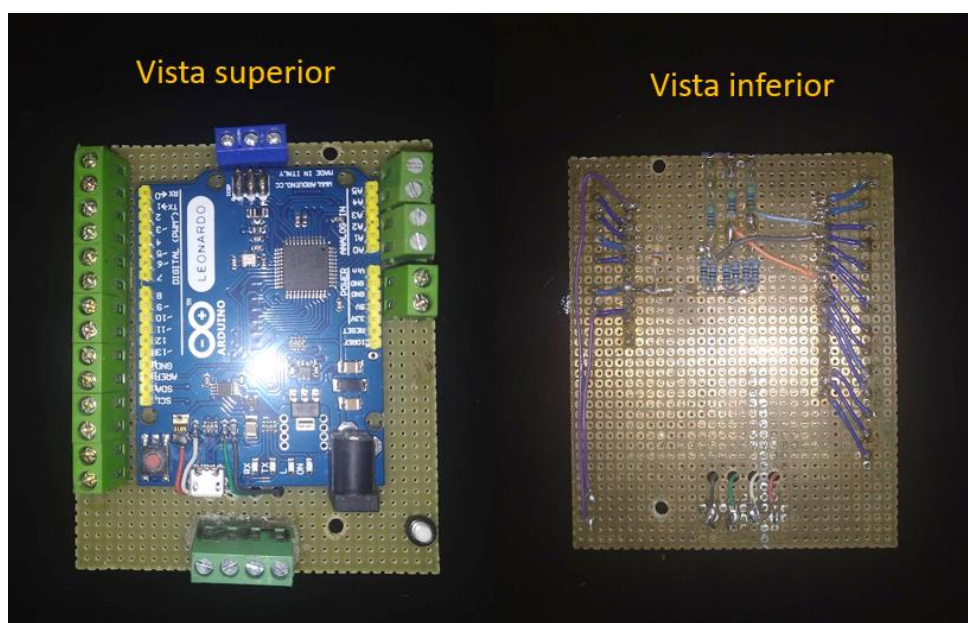
Com o objetivo de facilitar a obtenção dos parâmetros máximos e mínimos de cada pedal, bem como os pontos de referência posicional do volante, foi adicionado ao projeto uma chave táctil. A chave foi soldada junto à placa principal da central eletrônica e conectada a porta lógica A0 assumindo uma função específica dentro do software que permite calibrar o limiar das variáveis trabalhadas pelo firmware. Outro elemento utilizado foi o LED já embarcado no circuito da placa principal, que permite avaliar o status de configuração do simulador. Tanto o LED utilizado (LED 1) quanto a chave táctil (Switch 3) podem ser observados no circuito geral da placa, disposto na Figura 15.

### 3.1.2.1. Microcontrolador e Central Eletrônica

A fim de obter uma central microprocessada capaz de criar uma interface de comunicação entre os sensores, atuadores e o computador, optou-se pela utilização de um microcontrolador disponível em um sistema embarcado. O microcontrolador utilizado para o desenvolvimento do projeto foi o Atmega 32u4, disponível na placa conhecida como Arduino Leonardo. A escolha do microcontrolador foi feita devido a capacidade do mesmo de realizar uma comunicação direta via USB com o computador, como descrito na Seção 2.2.

A central eletrônica do simulador foi composta da seguinte forma: o sistema embarcado (Arduino Leonardo) foi montado e soldado em uma placa universal em que foram fixados conectores Borne com parafuso ao logo de toda placa para uma montagem mais adequada do cabeamento do chicote de cada elemento do simulador que se conecta à central eletrônica. A escolha de montagem foi feita assim para permitir uma melhor robustez das conexões e praticidade de manutenções e alterações nas conexões elétricas. Na placa da central elétrica também foram montados os circuitos eletrônicos auxiliares aos demais sistemas. A central eletrônica pode ser observada na Figura 16.

Figura 16 – Central eletrônica do simulador



Fonte: Próprio autor.

### 3.1.2.2. Motor e Alimentação

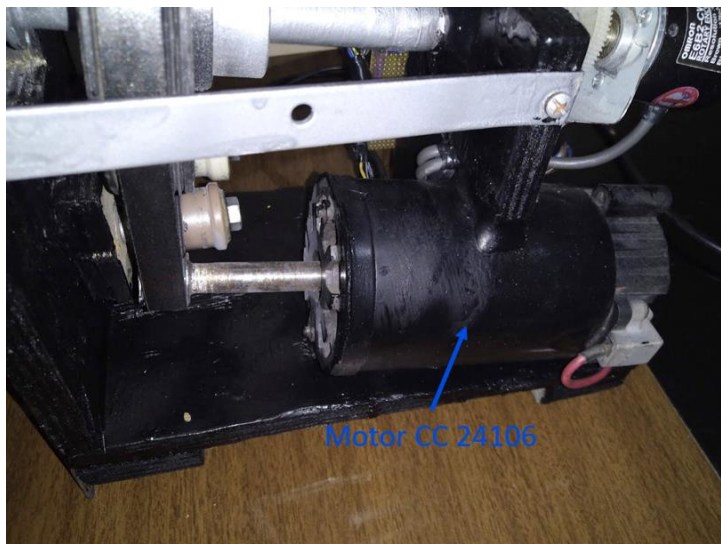
O sistema de atuação é composto especialmente por um motor de corrente contínua (CC), responsável por gerar torque e movimento ao volante em ambas as direções de rotação de acordo com o comando que lhe é fornecido.

Durante o desenvolvimento do projeto foram trabalhados alguns diferentes motores, mas foi observado que, pelo fato de os motores trabalharem grande parte do tempo em regime de rotor bloqueado, era necessário um motor de maior dimensão para que o sistema pudesse trabalhar de maneira contínua e sem a necessidade de resfriamento ativo.

Inicialmente, o projeto foi desenvolvido com um motor CC de ímã permanente 775 e tensão nominal de 12 V. Aplicando a tensão nominal, o motor foi capaz de gerar torques em níveis satisfatórios ao projeto, entretanto, a sobrecorrente gerada pelo status de rotor bloqueado no motor gerava uma sobrecarga de corrente no motor que resultava em sobreaquecimentos e, ao longo do tempo, danos ao componente. Utilizando essa primeira configuração de motor utilizou-se uma alimentação de 12 V de tensão e foram medidas correntes de até 10 A no motor utilizando-se um alicate amperímetro digital.

Tendo em vista o aspecto de dimensionamento e disponibilidade de material do autor do trabalho, o motor utilizado na versão final do projeto foi o motor de código 24106 produzido pela Universal Componentes da Amazonia Ltda., com potência nominal de 1,4 hp e tensão nominal de 90 V, comumente utilizado em esteiras ergométricas. O motor utilizado pode ser observado na Figura 17.

Figura 17 – Motor utilizado na versão final do projeto.



Fonte: Próprio autor.

A escolha dos componentes do sistema de alimentação do motor foi feita junto à escolha do mesmo, uma vez que diferentes motores podem demandar diferentes fontes de alimentação. Para versões iniciais do projeto, ainda com o motor 775, foi utilizada uma fonte chaveada de 12V de tensão e capacidade de corrente máxima de 15 A. Utilizando o motor final do projeto, por sua vez, a fonte selecionada foi de 24 V e 20 A.

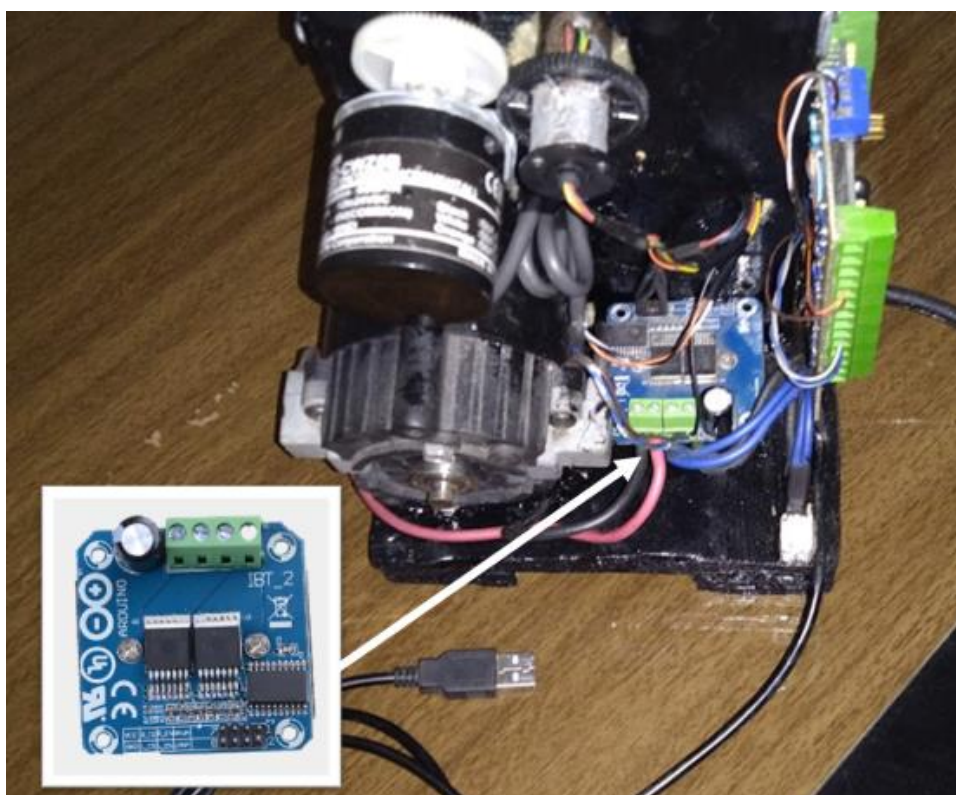
A escolha da fonte se deu da seguinte forma: Primeiramente, foi escolhida uma tensão consideravelmente menor que a tensão nominal do motor. Escolheu-se, assim uma tensão de apenas 24 V (aproximadamente  $\frac{1}{4}$  da tensão nominal de 90 V). Depois, foi medido o menor valor de resistência nos terminais do motor utilizando-se um multímetro digital, ao passo que o motor era lentamente rotacionado com as mãos. Dessa forma, era possível obter um valor de resistência mínima que seria uma referência para obter-se a corrente CC de rotor bloqueado. Como o motor é alimentado por corrente contínua, desprezou-se as indutâncias presentes no circuito. Encontrou-se, por esse método, uma medição de resistência no valor de 2,7  $\Omega$ . Portanto, pela lei de Ohm, calculou-se uma corrente de rotor bloqueado de 8,89 A. Dessa forma, a fonte de alimentação selecionada foi de 24 V e corrente máxima de 20 A, garantindo uma margem de segurança para evitar sobrecargas na fonte de alimentação.

### 3.1.2.3. Circuito de Acionamento

Visto que o motor seria comandado via sinal PWM e que deveria ser comandado nos dois sentidos de rotação, o circuito de acionamento do motor foi montado essencialmente por uma configuração de ponte completa.

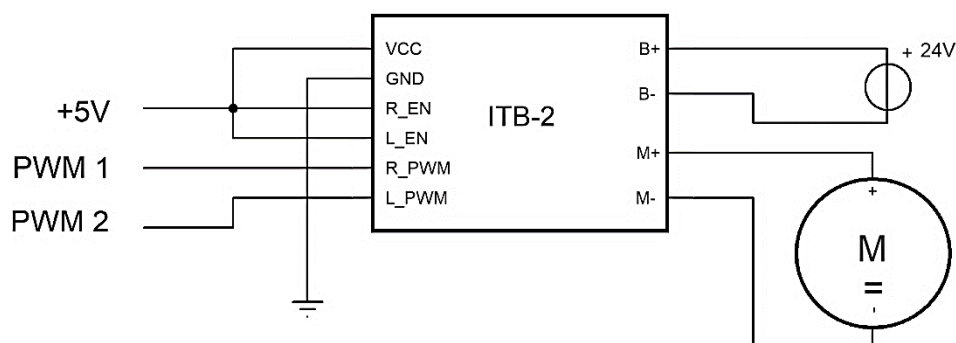
Optou-se pela utilização do módulo integrado IBT-2 (Figura 18), que implementa tanto o circuito de ponte, quanto o circuito driver da ponte, isto é, tanto os transistores de chaveamento quanto o condicionamento e proteção dos sinais de comando do circuito. O módulo é formado por dois circuitos integrados BTS7960 que são CIs que implementam uma configuração de meia ponte com driver integrado e possuem uma corrente de operação contínua máxima de 40 A, conforme indica o datasheet.

Figura 18 – Módulo IBT-2 e seu posicionamento na base do protótipo.



Fonte: Próprio autor.

Figura 19 – Esquema elétrico do sistema de comando.



Fonte: Próprio autor.

A Figura 19 mostra o esquemático de conexão do Módulo ITB-2 ao circuito geral do sistema de atuação. Observa-se na figura que os sinais PWM 1 e 2 são os sinais de comando sendo o PWM 1 o responsável pela intensidade de rotação para a direita e o PWM 2 o responsável pela rotação para a esquerda. Os sinais são fornecidos diretamente pela central eletrônica do circuito, isto é, diretamente do microcontrolador utilizado, correspondendo às portas digitais nº 9 (PWM 1) e 10 (PWM 2). A alimentação de 5V, por sua vez, vem diretamente do fornecimento da central eletrônica via USB.

#### 3.1.2.4. Encoder e Circuito Auxiliar

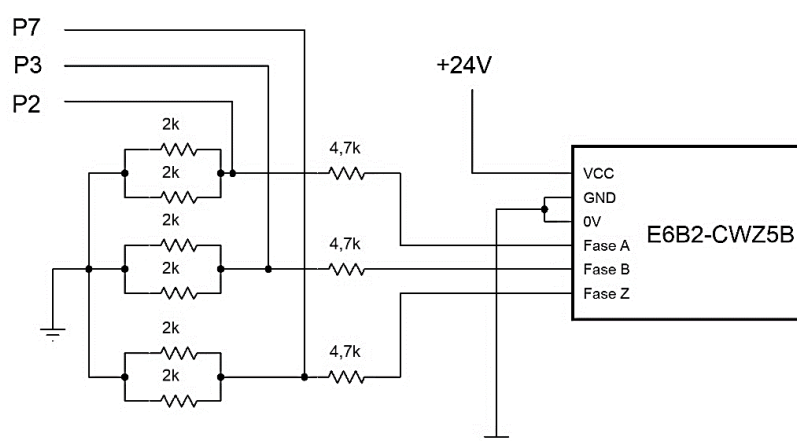
Da parte da instrumentação do volante, um dos principais elementos é o encoder, responsável pela aquisição digital das informações de rotação do volante: velocidade e posição angular.

Durante o desenvolvimento do projeto foram utilizados dois modelos de encoder: o primeiro, modelo OVW6-10-2HC possui resolução de 1000 pulsos por rotação, e o segundo modelo E6B2-CWZ5B possui resolução de 600 pulsos por rotação. Os dois encoders utilizados são incrementais, isto é, possuem duas fases de referência de rotação – permitindo assim determinar o sentido de rotação – sendo que o segundo (E6B2) possui também uma fase extra para indicar o ponto zero de rotação.

Optou-se pela troca do encoder utilizado devido a problemas de leitura e perda de referência posicional do eixo do volante. Tais problemas e soluções serão discutidos mais detalhadamente na Seção 3.2.3.

Por fim, em uma montagem elétrica final do projeto, o encoder utilizado (E6B2) foi montado conforme o seguinte esquema elétrico destacado na Figura 20.

Figura 20 – Esquema elétrico do circuito de leitura do encoder. Terminais P2, P3 e P7 indicam as portas digitais do microcontrolador.

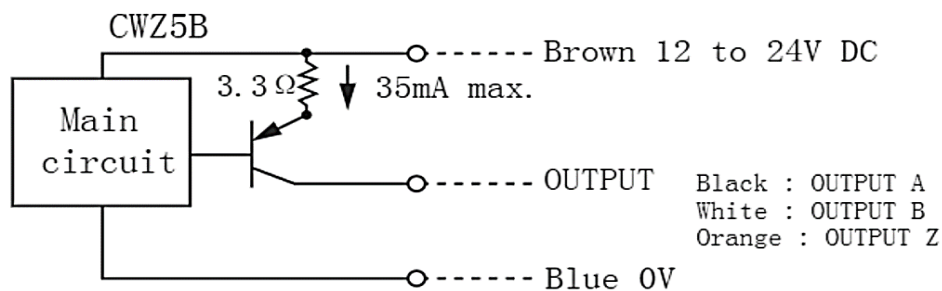


Fonte: Próprio autor.

Observa-se na Figura 20 que as saídas das fases de sinal do encoder passam por um circuito de condicionamento de sinal. O circuito é composto apenas por resistores e é necessário pois, pela alimentação do encoder ser feita com 24 V (através da mesma fonte de alimentação utilizada para o motor), o sinal de saída é feito no mesmo nível lógico. Dessa forma, para condicionar o sinal e obter uma leitura adequada do sinal na faixa de 0 a 5 V o sinal de cada fase que chega ao micro controlador é reduzido em tensão por um divisor resistivo.

Os resistores utilizados também cumprem o papel de circuito base para adquirir o sinal de saída, tendo em vista que as saídas do encoder trabalham na configuração de open collector pnp (coletor pnp aberto). Como mostra a Figura 21:

Figura 21 – Esquema elétrico interno do encoder.



Fonte: datasheet. (OMRON Industrial Automation, 2017).

Nota-se também na Figura 20 que as saídas de sinal vão conectadas aos pinos P2, P3 e P7, que se referem as portas digitais do microcontrolador.

### 3.1.2.5. Sensores Hall

Outro elemento de instrumentação utilizado no projeto foi o sensor de efeito hall, responsável pela leitura de posição dos pedais.

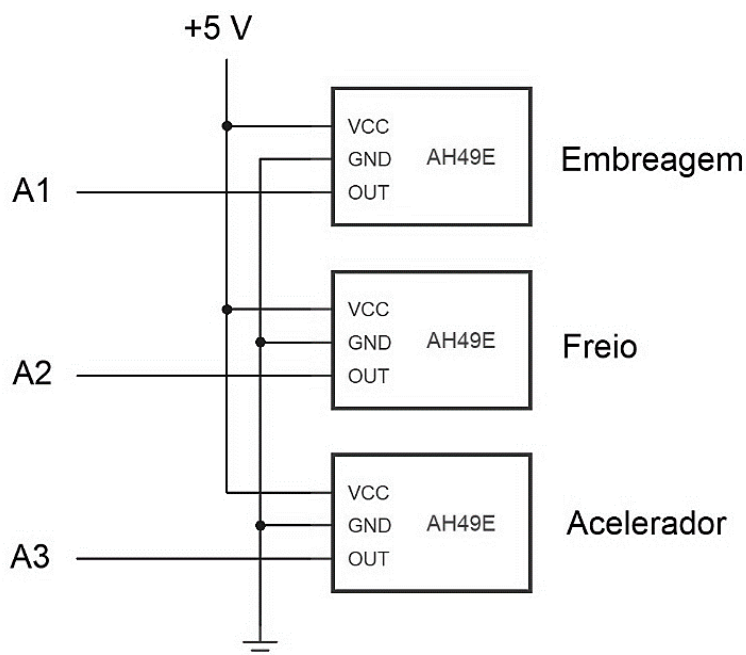
O modelo de sensor utilizado foi o AH49E (referenciar datasheet), um circuito integrado que implementa um sensor hall, amplifica e condiciona o sinal para uma saída linear de 1 a 4 V quando alimentado com uma tensão de 5 V.

No decorrer do desenvolvimento também foram utilizados potenciômetros para a leitura dos valores dos pedais. Essa opção, no entanto, foi substituída pela adoção do AH49E tendo em vista algumas dificuldades que surgiram com a utilização de potenciômetros: como o passar do tempo o acúmulo de sujeira na parte interna dos potenciômetros passava a afetar o valor das medidas tornando incerta a medida após algum tempo de uso, também houve uma dificuldade de desenvolver um sistema mecânico robusto para a leitura dos potenciômetros de maneira confiável e duradoura.

Assim, desenvolveu-se o sistema que é contemplado ao fim do projeto que é composto por 3 unidades do AH49E – um para cada pedal. Cada unidade é acoplada a base do pedal sendo inserido na face oposta do pedal um ímã que é responsável por gerar a densidade de fluxo magnético que é obtida em forma de tensão pelo sensor de acordo com a posição do pedal.

O esquema elétrico de conexão dos sensores pode ser observado na Figura 22.

Figura 22 – Esquema elétrico do circuito dos sensores AH49E.



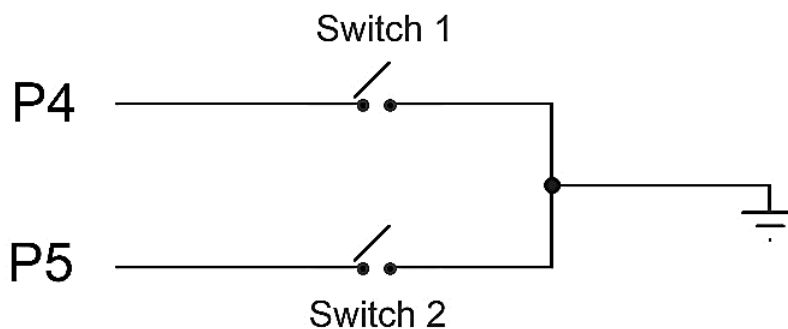
Fonte: Próprio autor.

Nota-se no esquemático que a conexão dos sensores foi feita de maneira direta com as portas analógicas do microcontrolador (A1, A2 e A3, respectivamente). Isso foi possível pelo fato de o sinal do sensor já estar dentro de uma faixa adequada para a leitura dos conversores A/D do microcontrolador 32u4, isto é, entre 0 e 5 V. Dessa forma, o tratamento do sinal foi feito digitalmente através do firmware desenvolvido para o microcontrolador.

### 3.1.2.6. Controles do Volante

Os controles de câmbio do volante foram compostos por apenas duas chaves do tipo micro switch que são responsáveis por mudar de estado elétrico quando as borboletas de troca de marcha são acionadas. Optou-se por utilizar o próprio circuito de pull-up interno do microcontrolador e, por isso, o circuito de acionamento das chaves se tornou simples como pode ser observado na Figura 23:

Figura 23 – Esquema elétrico das chaves das borboletas de câmbio.



Fonte: Próprio autor.

## 3.2. Software

A Seção de software é composta pelos elementos computacionais do presente trabalho. Ela contempla todo o desenvolvimento do firmware desenvolvido para o protótipo incluindo problemas e soluções encontrados durante seu desenvolvimento.

### 3.2.1. Requisitos e Estrutura Geral de Programação

Algumas características que o firmware deve apresentar para um bom resultado em softwares de simulação são:

- Adaptação a diferentes condicionamentos físicos;
- Compatibilidade com softwares de simulação;
- Atender a demandas do Hardware.

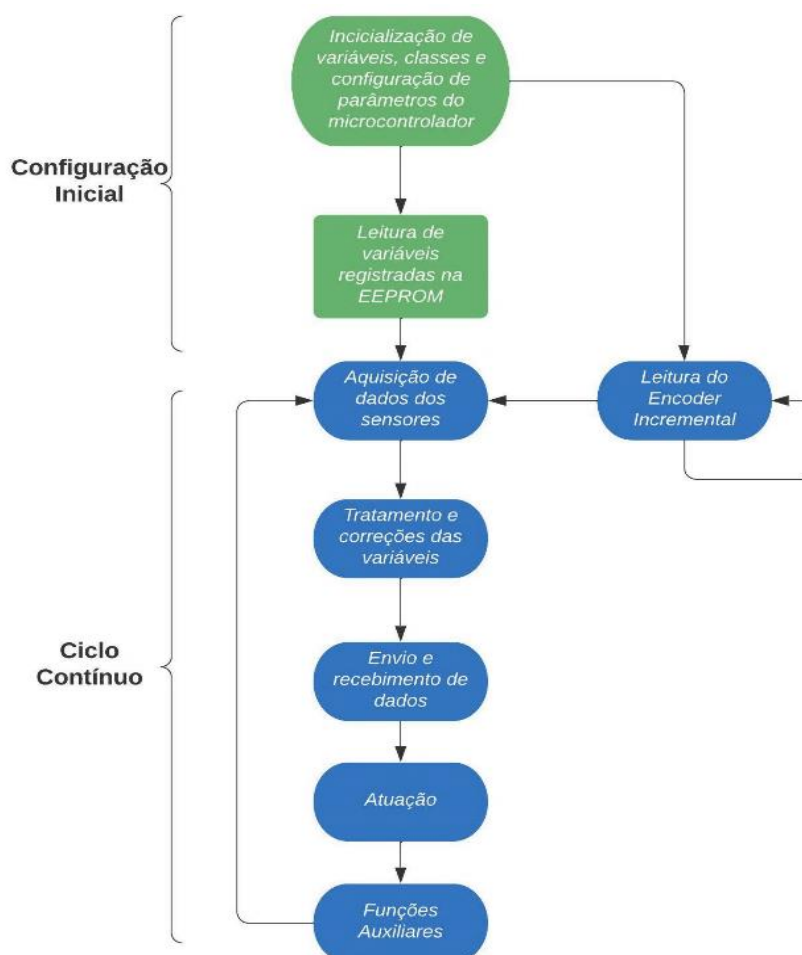
O primeiro tópico abrange a flexibilidade que o firmware necessita ter devido a diferentes condições físicas das variáveis de leitura e atuação. Mais especificamente, é notável que não é prático determinar fisicamente os valores de atuação dos conversores A/D bem como obter uma referência perfeita de posição do volante com base nas fases do encoder. Dessa forma estratégias devem ser tomadas para que tais referências sejam obtidas de maneira prática, e não a partir das variáveis físicas de montagem do projeto.

Para atender compatibilidade com os softwares de simulação, o quesito de eficiência computacional se fez imprescindível. Durante o desenvolvimento do projeto notou-se travamentos que ocorriam nos softwares de simulação devido ao tempo de ciclo de processamento do microcontrolador ser demasiadamente elevado, mais especificamente, para valores maiores que 1500  $\mu$ s.

Atender demandas do Hardware também é um importante ponto a ser levado em consideração. Nesse quesito aplica-se a capacidade do hardware de obter e processar as medidas obtidas dos sensores mantendo a fidelidade dos dados obtidos sem perda de informação nem referência física de posição.

A programação foi desenvolvida na linguagem C++ utilizando-se a plataforma Arduino IDE. A estrutura do firmware pode ser observada no fluxograma logo abaixo:

Figura 24 – Fluxograma de etapas do firmware desenvolvido.



Fonte: Próprio autor.

Tendo em vista a diagramação das etapas implementadas no firmware, algumas etapas específicas serão tratadas separadamente nos seguintes capítulos do presente trabalho permitindo assim uma compreensão logicamente contínua do projeto.

### **3.2.2. Interface e Comunicação USB**

Um dos elementos essenciais para o desenvolvimento da programação foi de antemão a escolha do microcontrolador adequado ao projeto. Como discorrido na Seção 2.2, um microcontrolador, para se comunicar com um computador externo na forma de um dispositivo de interface humana, precisa oferecer uma interface de comunicação USB própria. Assim, é possível que através de uma programação adequada o dispositivo seja reconhecido pelo computador como um mouse, teclado, joystick, controle, entre outros.

No caso particular do presente trabalho, o microcontrolador escolhido (Atmega32u4) possui uma interface de comunicação USB 2.0 própria, vide Figura 2, que o torna apropriado para a utilização como um dispositivo HID.

Toda interface de comunicação microcontrolador-computador foi realizada baseada na utilização de bibliotecas baseadas no protocolo HID que implementam classes e métodos próprios para o envio e recebimento de HID reports via comunicação USB. Dessa forma, o envio de valores correspondentes aos estados dos pedais e do volante, bem como o recebimento de dados do *force feedback* se limitou à aplicação dos métodos disponíveis durante a etapa de “Envio e Recebimento de dados” (etapa disposta no fluxograma do firmware).

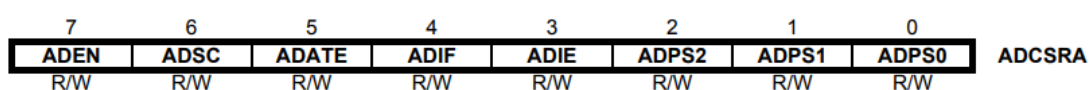
### **3.2.3. Aquisição e Parametrização de Dados dos Sensores**

A parte de aquisição de dados consiste na obtenção dos valores reproduzidos pelos sensores de efeito hall através da leitura dos conversores A/D e seus registradores. Essa etapa é a primeira do ciclo contínuo do firmware visto que as demais etapas necessitam dos valores obtidos para sua execução.

Em um primeiro momento, ainda na etapa de configuração de parâmetros do microcontrolador, foram feitos ajustes na configuração do registrador de controle e

status do conversor analógico/digital A (ADCSRA). Esse registrador é responsável por comandar a leitura, receber o status e definir a pré-escala de divisão do clock externo que comanda o microcontrolador (que normalmente é de 16 MHz) gerando um novo clock chamado de clock ADC, responsável por comandar a conversão analógico/digital. Abaixo, segue a ilustração do registrador ADCSRA dada pelo datasheet.

Figura 25 – Ilustração do registrador ADCSRA do ATmega32u4.



Fonte: adaptado do datasheet. (ATMEL, Corporation).

Os três bits responsáveis por definir a pré-escala de divisão do clock são os 3 primeiros bits do registrador: ADPS0, ADPS1 e ADPS2. Configurando esses bits é possível dividir o clock de operação de 2 até 128 vezes. Por padrão, o software da Arduino IDE define a pré-escala com um fator de divisão de 128 vezes. Entretanto tendo em vista a necessidade do firmware de possuir tempos de ciclo otimizados, optou-se por mudar o valor da pré-escala para um fator de divisão de apenas 16 vezes (bits: 0b100), garantindo assim um clock ADC de 1 MHz.

Considerando que cada conversão A/D demanda 13 ciclos do clock ADC—conforme indica o datasheet na página 301 – Cada conversão A/D do firmware passou a ocupar um tempo de aproximadamente 13  $\mu$ s. Pelo mesmo método, com uma pré-escala padrão de 128 vezes, o tempo de conversão seria de aproximadamente 104  $\mu$ s.

Em um segundo momento, já na etapa de aquisição dos dados e dentro do ciclo contínuo do firmware, os valores de cada conversão A/D referente aos valores dos pedais são lidos e armazenados em suas respectivas variáveis. Dessa forma, o próximo passo é normalizar os valores dos pedais em uma escala linear de 0 a 1. A normalização dos valores e parametrização foi feita de acordo com a seguinte equação:

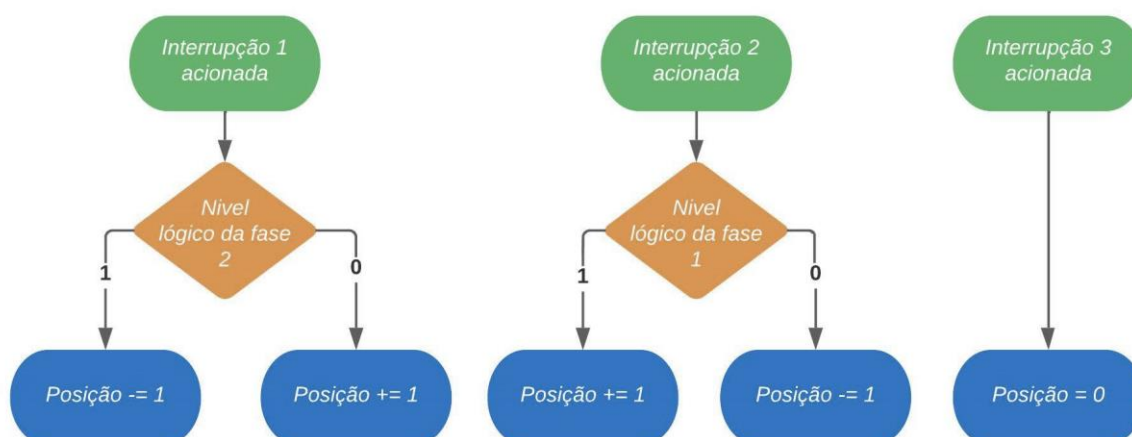
$$V_{norm} = \frac{1}{V_{max} - V_{min}} \cdot (V_{medido} - V_{min}) \quad (7)$$

Observa-se na equação 7 que os valores  $V_{norm}$ ,  $V_{max}$ ,  $V_{min}$  e  $V_{medido}$  correspondem a: o valor normalizado da leitura da variável, o valor máximo que a variável pode assumir, o valor mínimo que a variável pode assumir e o valor medido pela conversão A/D, respectivamente. Os valores máximos e mínimos das variáveis são definidos por funções auxiliares que serão discutidas em seu respectivo capítulo e são lidas logo após a inicialização do programa na etapa de leitura das variáveis registradas na EEPROM.

Na etapa de aquisição de dados também é realizada a parametrização e normalização da leitura de posição do volante, realizada pela etapa de leitura do encoder incremental.

A leitura do encoder incremental, por sua vez, ocorre de maneira paralela à execução das demais etapas do ciclo contínuo. A leitura dos pulsos do encoder incremental se dá pela utilização de rotinas de interrupções externas nas portas digitais 2, 3 e 7 do microcontrolador. Utilizou-se exclusivamente das bordas de subida das fases do encoder para que fossem definidas as interrupções que computariam uma alteração no valor de posição. As interrupções 1 e 2 referem-se as fases conectadas aos pinos 2 e 3 e a interrupção 3 refere-se a fase de referência zero conectada ao pino 7. O diagrama abaixo mostra a lógica de atuação das interrupções na variável de posição.

Figura 26 – Fluxograma das operações realizadas pelas interrupções externas.



Fonte: Próprio autor.

Com a variável de posição atualizada, a normalização dos parâmetros de posição do volante é dada pelas seguintes equações:

$$X_{dir} = \frac{1}{X_{min} - X_{mid}} \cdot (X_o - X_{mid}) \quad (8)$$

$$X_{esq} = \frac{1}{X_{max} - X_{mid}} \cdot (X_o - X_{mid}) \quad (9)$$

Nas equações XX, denotam-se as variáveis por:

- $X_{esq}$ : valor normalizado de posição à esquerda;
- $X_{dir}$ : valor normalizado de posição à direita;
- $X_{min}$ : valor absoluto mínimo de leitura do encoder;
- $X_{mid}$ : valor absoluto médio de leitura do encoder (centro);
- $X_{max}$ : valor absoluto máximo de leitura do encoder;
- $X_o$ : valor absoluto do encoder corrigido de um off-set central.

Optou-se por utilizar um cálculo separado para ambas as direções de rotação do volante devido a possibilidade de trabalhar-se com diferentes limiares para ambas as direções de rotação.

Alguns dos problemas encontrados durante o desenvolvimento do projeto estão relacionados com a referência de posição angular obtida pela leitura do encoder incremental. Nas primeiras versões do firmware – ainda utilizando o encoder OVW6 – foram percebidos desvios na leitura de posição do encoder, fazendo com que o software, com movimentações abruptas e repetitivas aplicadas ao eixo do volante, aos poucos apresentasse um erro de leitura que tornava a posição angular medida incorreta. Tal fato pode ocorrer como consequência da incapacidade do microcontrolador de captar todos os pulsos emitidos pelo encoder, haja vista as altas frequências geradas pelas fases durante movimentos rápidos.

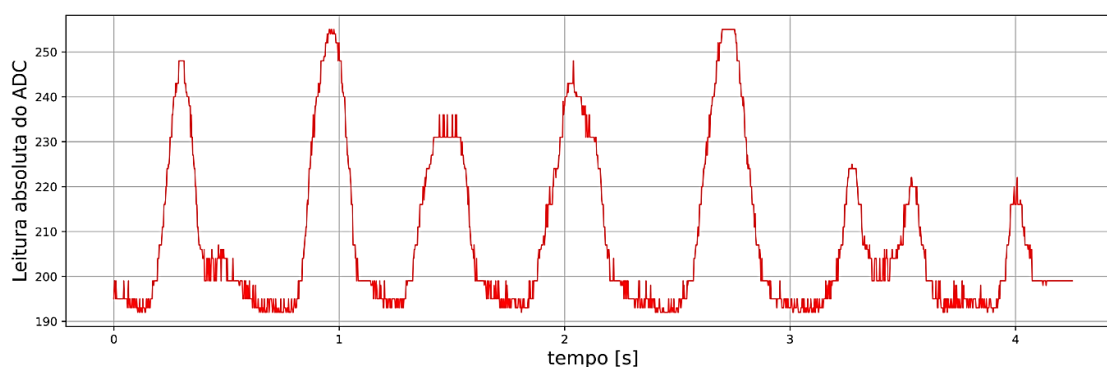
Tendo em vista os problemas encontrados, optou-se por trocar o modelo do encoder utilizado para um modelo com menor resolução – resultando assim em menores frequências dos pulsos recebidos pelo microcontrolador – e que também possuísse uma fase de referência zero. Dessa forma, garantiu-se no firmware que,

sempre que a interrupção 3 fosse acionada pela fase de referência zero, se tornaria certa a posição absoluta do encoder. Assim, justificou-se a adoção e implementação do encoder E6B2-CWZ5B utilizado ao fim do projeto.

### 3.2.4. Tratamento de Variáveis e Processamento de Sinais

Além dos erros de leitura apresentados pelo sistema do encoder, durante os primeiros ensaios executados nas versões primitivas do firmware foram detectados ruídos de alta frequência excessivos na parte de amostragem dos sinais analógicos correspondentes aos pedais de comando. Observou-se que ao realizar a amostragem dos sinais, os conversores A/D do microcontrolador não eram capazes de manter a mesma resolução de leitura e precisão quando submetidos a valores de clock ADC mais altos que o valor padrão sendo essas as possíveis causas dos ruídos existentes nas leituras dos sinais. A Figura 27 mostra um exemplo de sinal capturado enquanto o pedal da embreagem era movimentado continuamente. Nela é possível observar a presença dos ruídos, que eram mais notados ainda em situações de movimentos de baixa amplitude.

Figura 27 – Gráfico de leitura do pedal da embreagem com movimentos contínuos e repetidos.



Fonte: Próprio autor.

Surgiu-se então a necessidade de realizar-se o tratamento dos dados recebidos pelos conversores A/D a fim de obter-se leituras mais precisas e contendo menores taxas de ruídos.

Criou-se então a estratégia da implementação de filtros que permitissem a obtenção das informações dos sensores eliminando ou atenuando as componentes

espectrais de alta frequência em que se encontravam os ruídos indesejado presente no sinal.

Um primeiro tipo de filtro proposto para a solução do problema foi a implementação de um filtro do tipo média móvel. A estrutura do filtro de média móvel experimentado pode ser observada na equação abaixo:

$$y[n] = \frac{1}{N} \cdot (x[n] + x[n - 1] + x[n - 2] + x[n - 3] + \dots + x[n - N + 1]) \quad (10)$$

Observa-se na Equação 10 que N representa a quantidade de termos  $x[n-k]$  somados no cálculo de  $y[n]$ ,  $y[n]$  representa a saída do filtro digital ao passo que  $x[n-k]$  representa as amostras do sinal de entrada, o qual será filtrado. N, por sua vez representa a ordem do filtro utilizado.

Analicamente, observa-se que o filtro de média móvel implementado se trata de um filtro do tipo FIR, caracterizando assim sua inerente estabilidade dinâmica, bem como sua não dependência dos termos autorregressivos (ausência de realimentação).

Foram implementados filtros de média móvel de ordem 15, 25 e 40. Em termos de filtragem do sinal, os resultados da filtragem se mostraram promissores, entretanto, foi observado que o tempo de processamento demandado pelo microcontrolador era demasiadamente alto, visto que era demandada a criação e deslocamento completo de um buffer de valores do sinal de entrada a cada ciclo em que os sinais fossem filtrados.

Tendo em vista a necessidade de um firmware que possuísse baixos tempos de amostragem e que fosse computacionalmente eficiente, optou-se pela adoção de filtros IIR para o processamento dos sinais ruidosos. Dessa forma, foram implementados filtros IIR discretos de 1ª ordem que foram os filtros utilizados na versão final do trabalho.

O filtro projetado tem a função e transferência como a mostrada na Equação 11:

$$H_f(z) = \frac{k_f z}{z + (k_f - 1)} \quad (11)$$

Nota-se na função de transferência que se trata de um filtro passa baixas discreto (para  $0 < k_f < 1$ ) de primeira ordem e que, por isso, apresenta uma atenuação do sinal de entrada a partir de uma determinada frequência de corte, que é definida pelo valor de  $k_f$ .

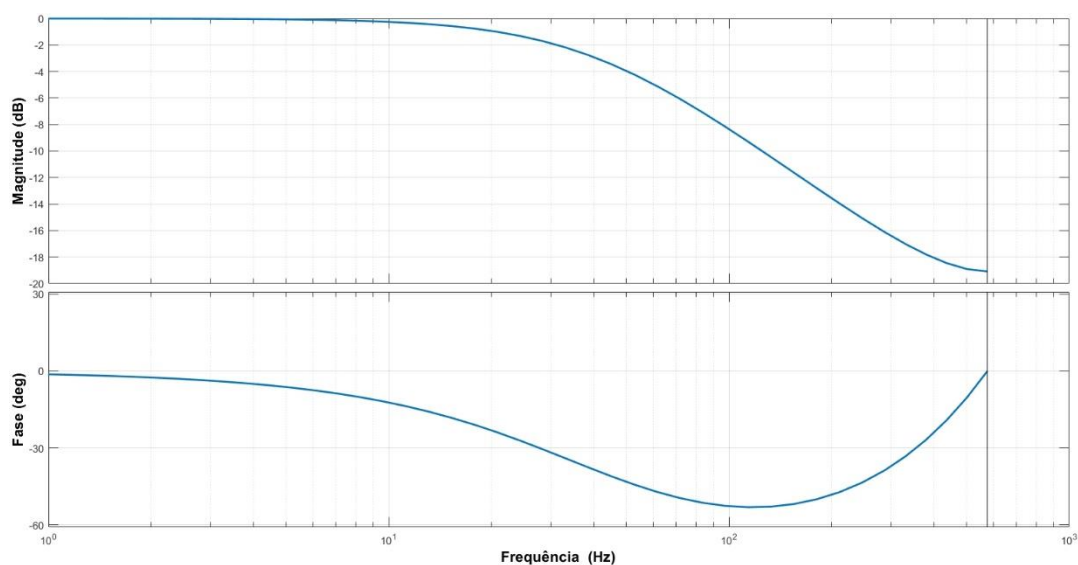
O valor de  $k_f$  foi obtido empiricamente através de ensaios em que foram verificados e comparados os sinais filtrados e não filtrados obtidos nas leituras dos sensores. Muitos valores de  $k_f$  foram testados, entretanto os melhores resultados surgiram com  $k_f = 0,2$ . Os principais fatores considerados na escolha dos possíveis valores de  $k_f$  foram: atenuação dos ruídos presentes no sinal e o desfasamento gerado no sinal de saída em relação ao sinal de entrada do filtro, que pode ser tido como o ônus a aplicação de filtros passa baixas em tempo real no projeto.

Assim, a função de transferência dos filtros utilizados para a leitura dos pedais do simulador é dada por:

$$H_f(z) = \frac{0,2z}{z - 0,8} \quad (12)$$

O diagrama de Bode do filtro utilizado pode ser observado na Figura 28, logo abaixo:

Figura 28 – Gráfico de módulo (acima) e fase (abaixo) do diagrama de bode da função de transferência do filtro digital aplicado.



Fonte: Próprio autor.

O filtro passa baixas foi aplicado no sinal dos pedais através da seguinte equação de diferenças:

$$y_f[n] = y_f[n - 1] + k_f \cdot (x[n] - y_f[n - 1]) \quad (13)$$

Na equação acima, leia-se  $y_f[n]$  pela saída do filtro digital, e  $x[n]$  é a amostra do sinal não filtrado.

O período de amostragem do filtro digital aplicado é dado pelo tempo de ciclo total do software que, como elucidado na Seção 4.4, é de 870  $\mu$ s.

Por fim, o filtro desenvolvido foi aplicado a cada variável responsável pela leitura de um dos pedais. Em outras palavras, a leitura de todos os pedais do simulador passou pelo mesmo processo de filtragem de sinais.

Os resultados do processo de filtragem são discutidos na Seção 4.2.1.

### 3.2.5. Atuação

Após a etapa de envio e recebimento de dados, os valores referentes ao *force feedback* são mapeados e aplicados nos canais de saída para o comando do motor.

Toda etapa de atuação é realizada através de modulação PWM. Sendo a disponibilidade do microcontrolador utilizado, a modulação PWM aplicada é definida em 8 bits a uma frequência de 490 Hz.

Por utilizar duas portas digitais diferentes (P9 e P10) os sinais de *force feedback* são representados com valores entre -255 e 255 sendo que valores negativos são aplicados em módulo ao canal de comando de rotação para a esquerda e os valores positivos são aplicados em módulo para o canal de comando de rotação para a direita.

### 3.2.6. Funções auxiliares

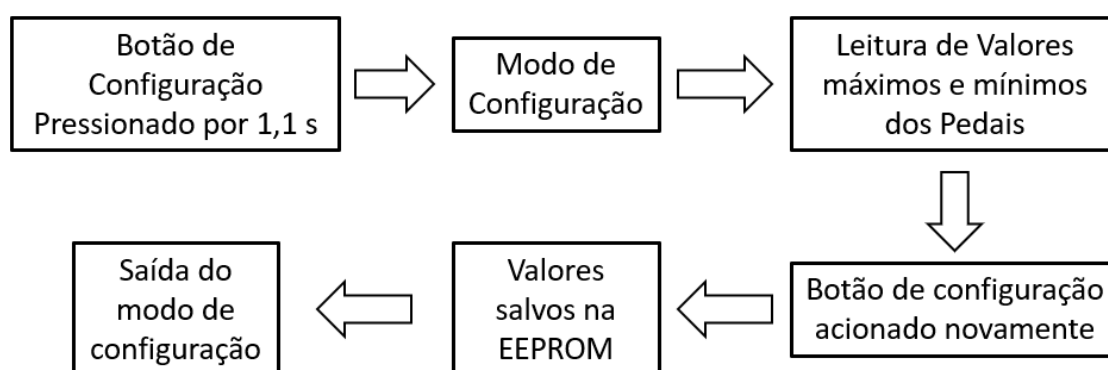
Para uma melhor usabilidade do protótipo, bem como para obter ajustes de parâmetros de maneira mais empírica, foram implementadas funções de calibração especiais que podem ser acessadas através do botão de configuração.

O firmware, em sua versão final, foi programado para que, ao pressionar-se o botão, o ponto de referência central do volante fosse sobrescrito nos parâmetros básicos salvos na EEPROM do microcontrolador. Outra função criada foi a de calibração dos pedais que permite que, pressionando e segurando o botão acionado por cerca de 1,1 s, o software entre no modo de calibração.

O modo de calibração, por sua vez, é um estado em que o firmware registra todos os valores máximos e mínimos obtidos nas leituras dos pedais – permitindo assim que as demais funções de aquisição e parametrização de variáveis trabalhem com os valores físicos obtidos na leitura dos pedais. Ao pressionar novamente o botão, enquanto no modo de calibração, o software salva todos os valores máximos e mínimos de cada pedal detectado na EEPROM do microcontrolador e logo após deixa o modo de calibração, voltando a operar novamente no ciclo contínuo.

A Figura 29 mostra o fluxo de operações que descreve a função auxiliar de configuração e calibração.

Figura 29 – Fluxograma do modo de configuração.



Fonte: Próprio autor.

## 4. RESULTADOS

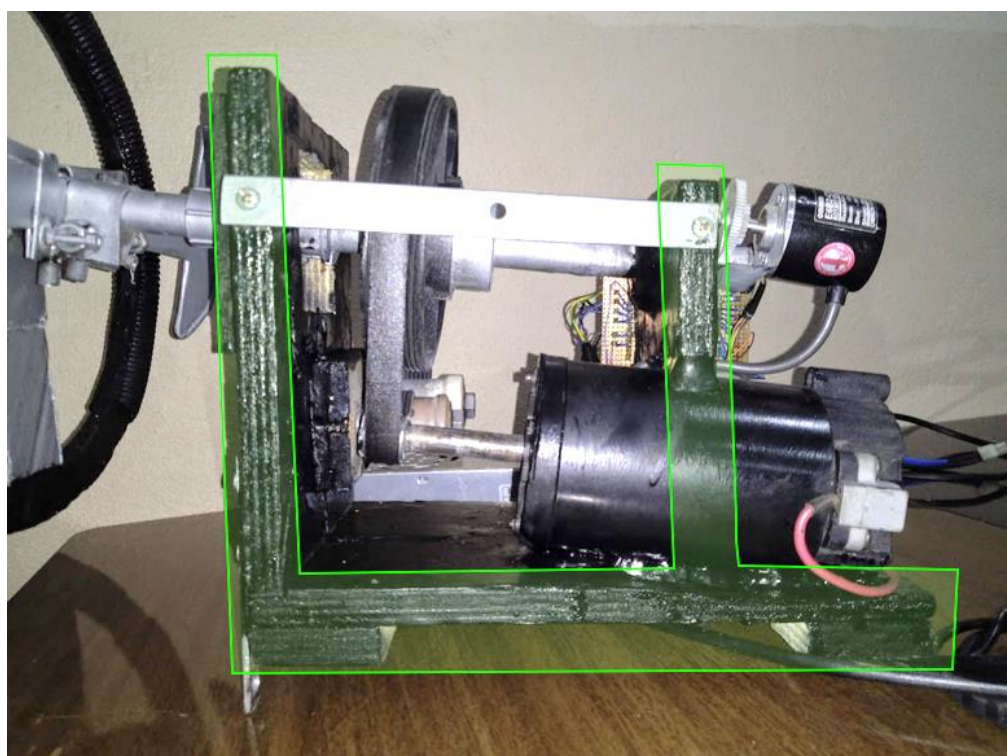
Os resultados do trabalho foram observados individualmente de acordo com os seguintes tópicos: panorama da montagem física do simulador, operação dos sensores, operação dos atuadores, comunicação e aplicação do simulador.

### 4.1. Montagem física do simulador

#### 4.1.1. Base Estrutural

A Figura 30 destaca a base estrutural montada para a parte mecânica do simulador.

Figura 30 – Base estrutural com componentes mecânicos e elétricos montados. Contornos da base estrutural destacados em verde.

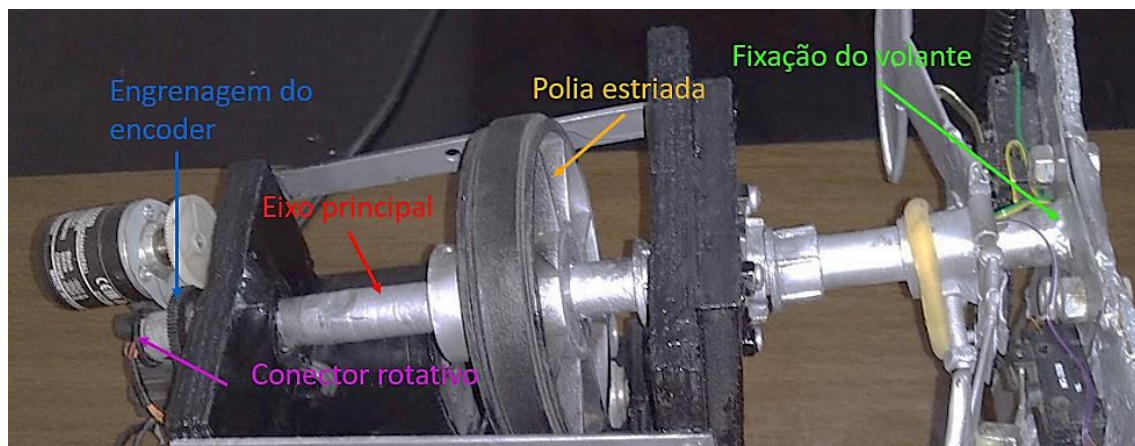


Fonte: Próprio autor.

#### 4.1.2. Eixo Principal

O eixo principal e seus elementos podem ser observados na Figura 31.

Figura 31 – Elementos do eixo principal.

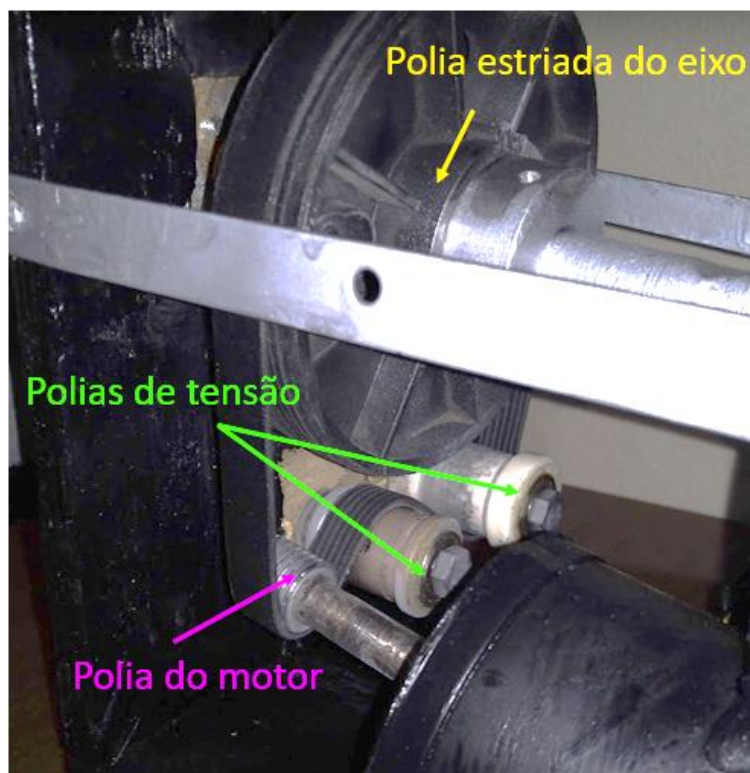


Fonte: Próprio autor.

#### 4.1.3. Sistema de Transmissão

O sistema de transmissão desenvolvido para o sistema de force feedback pode ser observado na Figura 32.

Figura 32 – Elementos do sistema de transmissão.

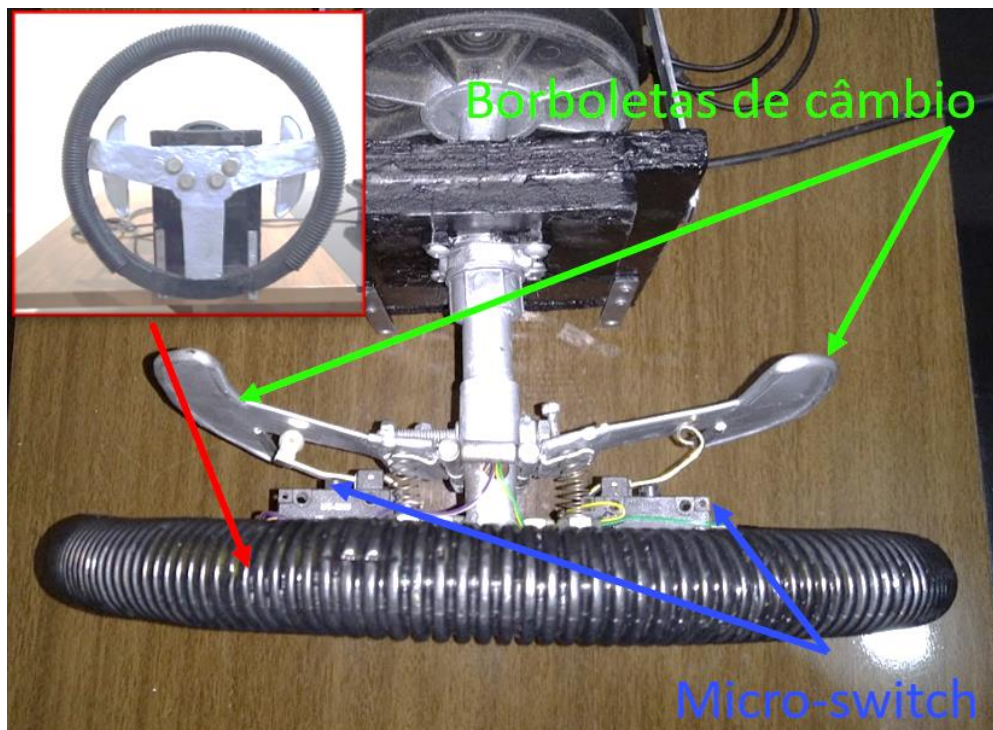


Fonte: Próprio autor.

#### 4.1.4. Volante e Câmbio

A Figura 33 mostra o volante construído com o câmbio montado.

Figura 33 – Volante do simulador e câmbio montado.



Fonte: Próprio autor.

#### 4.1.5. Panorama geral do simulador

Ao fim do projeto, foi obtido um conjunto físico que compõe o simulador desenvolvido. O conjunto do simulador é formado basicamente por 3 principais elementos: Base do volante, base dos pedais e fonte de alimentação externa. Os três elementos podem ser observados nas figuras 34 a 36:

Figura 34 – Base do volante do simulador.



Fonte: Próprio autor.

Figura 35 – Base de pedais.



Fonte: Próprio autor.

Figura 36 – Conjunto completo do simulador.



Fonte: Próprio autor.

Os elementos do simulador trabalham apenas em conjunto não podendo ser utilizados individualmente.

Com o simulador em operação, a fonte de alimentação é alimentada pela rede elétrica de 127 V ou 220 V e é conectada ao simulador através de um conector ATX de 4 pinos. A base do volante é conectada ao computador através de um cabo usb e a base de pedais é conectada à base do volante através de um cabeamento fixo de 5 vias.

## **4.2. Sensores**

Em operação, o simulador realiza a leitura dos sensores dispostos nos pedais para a obtenção da posição dos pedais bem como a leitura do encoder para obter a posição angular do volante.

### **4.2.1. Leitura dos pedais e calibração**

A leitura dos valores dos pedais é realizada pelo microcontrolador e tem por base a obtenção dos valores máximos e mínimos para cada pedal (lidos pelo

conversor AD) que são obtidos pela função de calibração implementada. Dessa forma, por se tratar de uma conversão analógico-digital de 10 bits, os valores absolutos de conversão podem variar de 0 a 1023. Ao fim do projeto, os valores obtidos para cada pedal podem ser observados na Tabela 1.

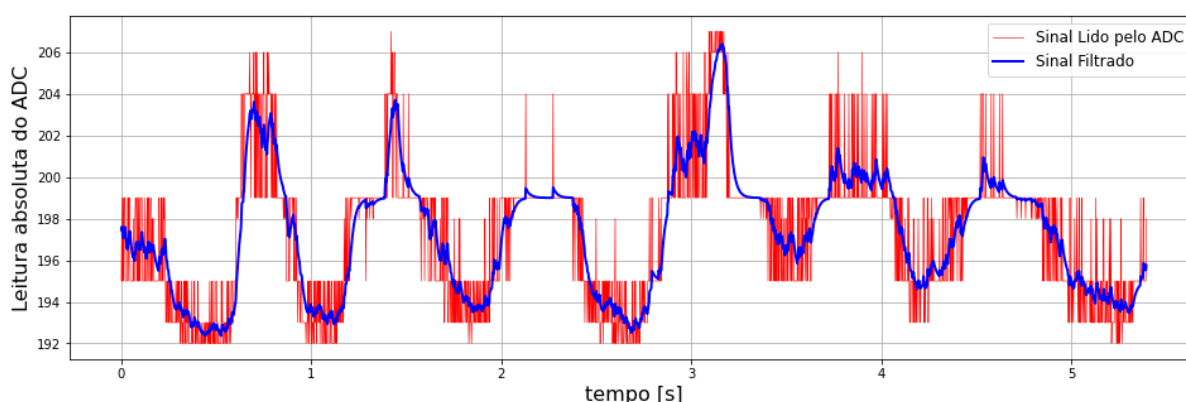
Tabela 1 – Valores obtidos em calibração para a conversão A/D dos valores de posição dos pedais do simulador.

	Valor mínimo	Valor máximo
Acelerador	528	711
Freio	851	972
Embreagem	175	274

Fonte: Próprio autor.

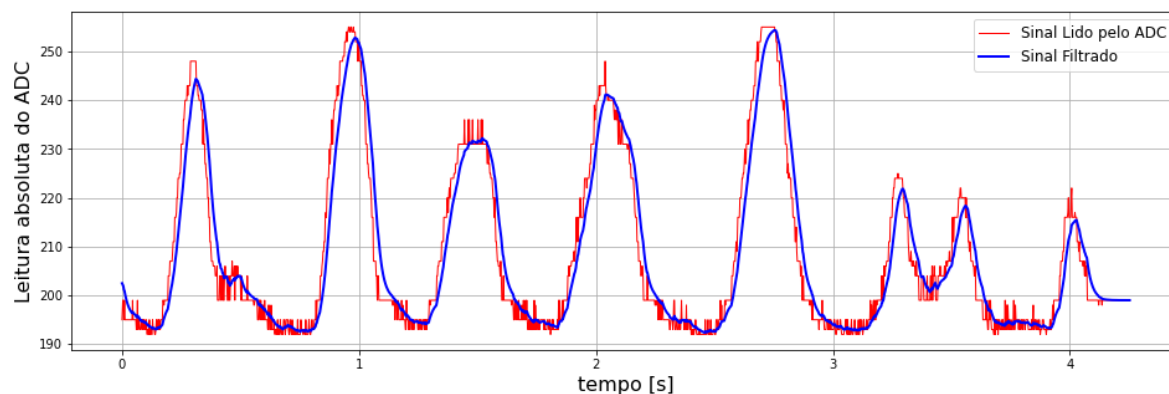
Outro processo que envolve a leitura dos pedais é a filtragem digital realizada após a aquisição dos valores. Os resultados do processo de filtragem podem ser observados nos gráficos dispostos nas figuras 37 e 38. A Figura 37 compara a leitura crua do conversor AD para movimentos de baixa amplitude. A Figura 38, por sua vez realiza a mesma comparação para movimentos de grande amplitude.

Figura 37 – Gráfico de leitura do pedal de embreagem e filtragem do sinal para movimentos de baixa amplitude.



Fonte: Próprio autor.

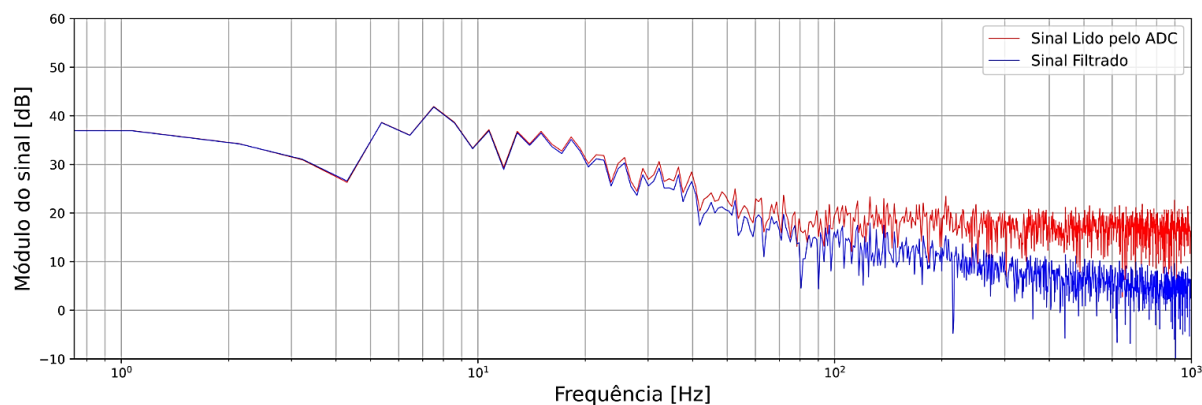
Figura 38 – Gráfico de leitura do pedal e filtragem do sinal para movimentos de grande amplitude.



Fonte: Próprio autor.

Analisando os gráficos, é possível observar que a filtragem dos sinais foi relevante permitindo amenizar os ruídos de alta frequência derivados da redução de resolução e amostragem do conversor AD. A Figura 39 mostra a FFT do sinal filtrado e não filtrado (mesmo sinal da Figura 38), em que é possível observar que as altas frequências do sinal foram atenuadas em mais de 10 dB.

Figura 39 – FFT dos sinais do pedal filtrado e não filtrado.



Fonte: Próprio autor.

A defasagem do sinal filtrado, por sua vez, apesar de não ter sido avaliada quantitativamente, pode ser observada na Figura 38 pelo pequeno atraso que o sinal filtrado (azul) apresenta em relação ao sinal não filtrado (vermelho). Em termos de aplicação prática, o atraso de resposta gerado pela filtragem do sinal não se mostrou significativo ao ponto de ser notado ou afetar a usabilidade do simulador.

Também ocorreu o fenômeno (pode ser melhor observado na Figura 37) de o sinal filtrado ser capaz de assumir valores intermediários entre os valores quantizados da leitura digital, efeito que se assemelha ao aumento da resolução do sinal digital.

#### 4.2.2. Encoder

Em termos da aquisição de posição do volante, os problemas descritos na Seção 3.2.3 foram solucionados através da utilização da fase de posição zero do encoder de forma que, toda vez que o volante passasse pela posição 0 – isto é, o volante em posição centralizada – o software obtinha a referência de posição central do volante, corrigindo assim o erro acumulado pela perda de pulsos lidos do encoder pelo microcontrolador.

Utilizando essa técnica, não foram mais notados erros de leitura de posição ao longo do tempo de uso do simulador. A leitura do encoder, dessa forma, por não depender de variáveis analógicas não se caracterizou por apresentar ruídos ou erros de leitura, sendo assim adequada à utilização do simulador.

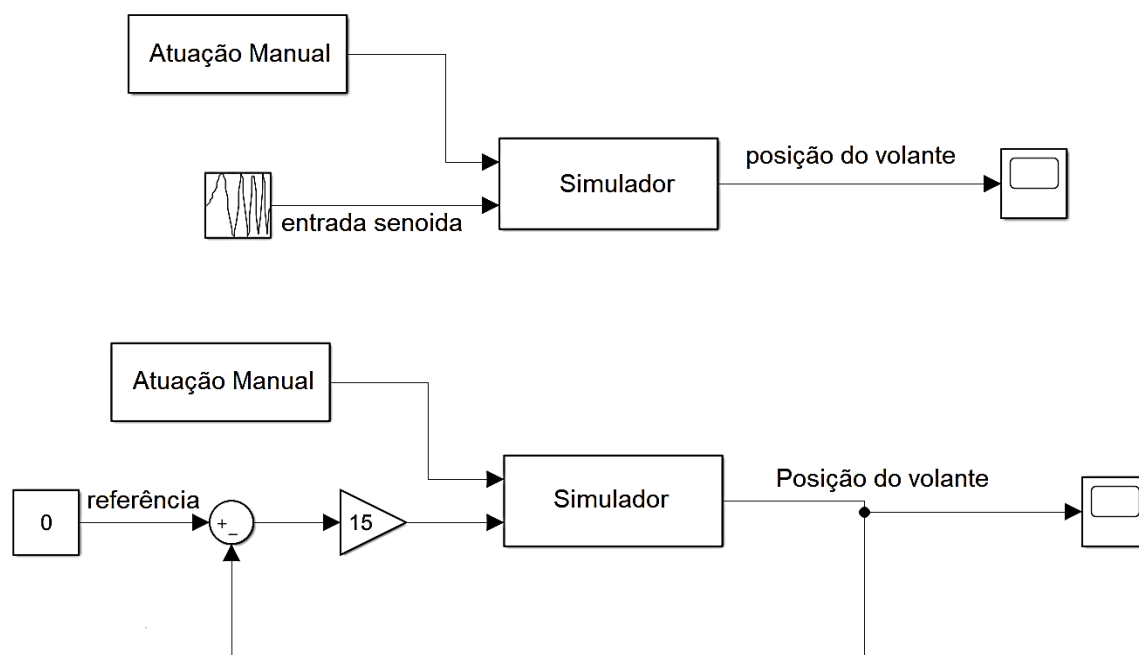
Alguns gráficos dispostos nas figuras 42 e 44 (Seção 4.3) demonstram a leitura do encoder em conjunto com a atuação do motor de *force feedback*.

#### 4.3. Atuadores

O sistema de atuação do simulador, composto pelo motor e seus dispositivos de acionamento, foi avaliado realizando-se ensaios dinâmicos em malha aberta e fechada bem como através da utilização prática do simulador em softwares de simulação.

Os ensaios dinâmicos foram realizados com o auxílio do software “*iRacing Force Feedback Test 1.72*”, ferramenta de teste para sistemas de *force feedback* disponibilizada pelo simulador *iRacing*. Essa ferramenta permitiu a configuração da malha de teste bem como a aplicação de sinais específicos na entrada do sistema. A Figura 39 mostra os diagramas de blocos das configurações de malha utilizadas para a realização dos ensaios dinâmicos.

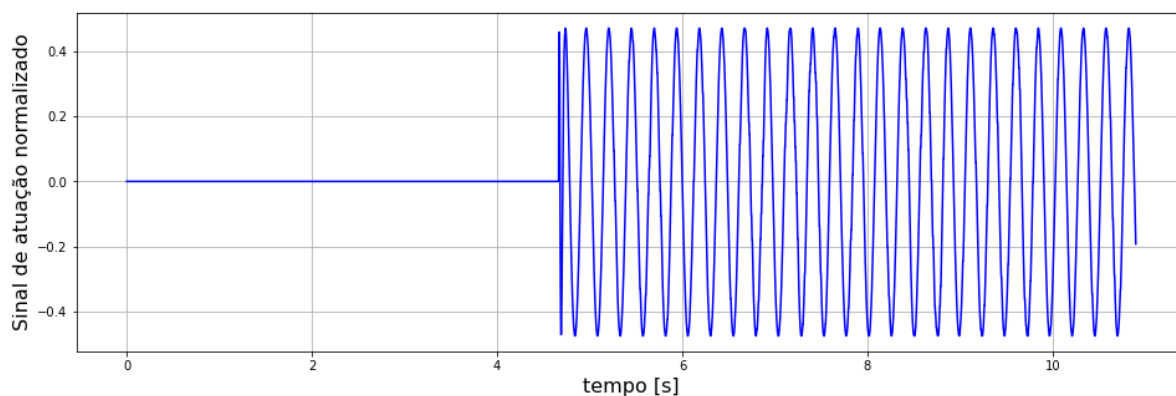
Figura 40 – Diagramas de blocos de malha aberta e fechada das configurações de malha utilizadas nos ensaios dinâmicos. Configuração em malha aberta com entrada senoidal (cima) e configuração em malha fechada com referência zero (baixo).



Fonte: Próprio autor.

Em um primeiro momento foi realizado um ensaio em malha aberta (Figura 40, diagrama superior) aplicando-se uma senoide na entrada do sistema de atuação. As figuras 41 e 42 mostram o sinal de entrada e a resposta do volante.

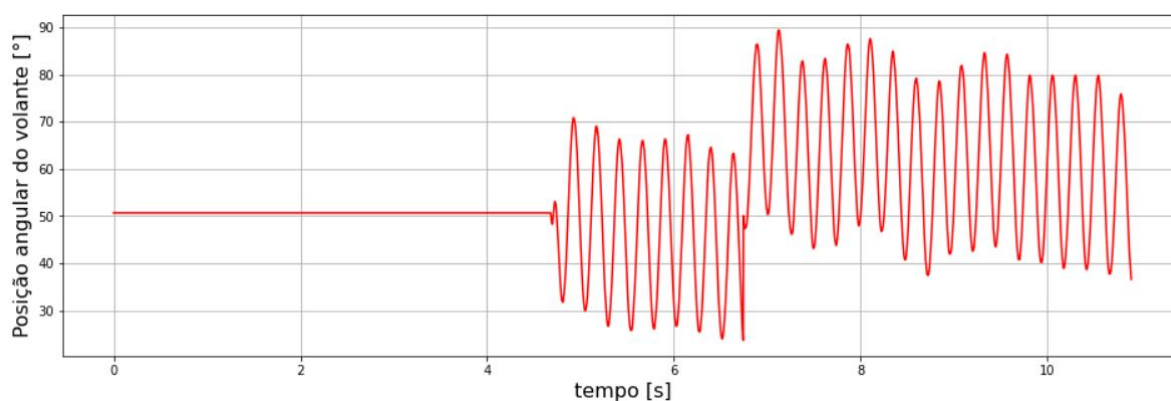
Figura 41 – Gráfico de entrada do sistema de atuação para um ensaio dinâmico em malha aberta e com entrada senoidal.



Fonte: Próprio autor.

O sinal de entrada é caracterizado por uma senoide de 4,1 Hz e 0,47 de amplitude normalizada (1 é o valor máximo de força aplicada no motor) que se inicia entre os instantes 4 e 5 segundos do ensaio dinâmico.

Figura 42 – Resposta do volante para entrada senoidal partindo do estado de equilíbrio dinâmico em 50°.



Fonte: Próprio autor.

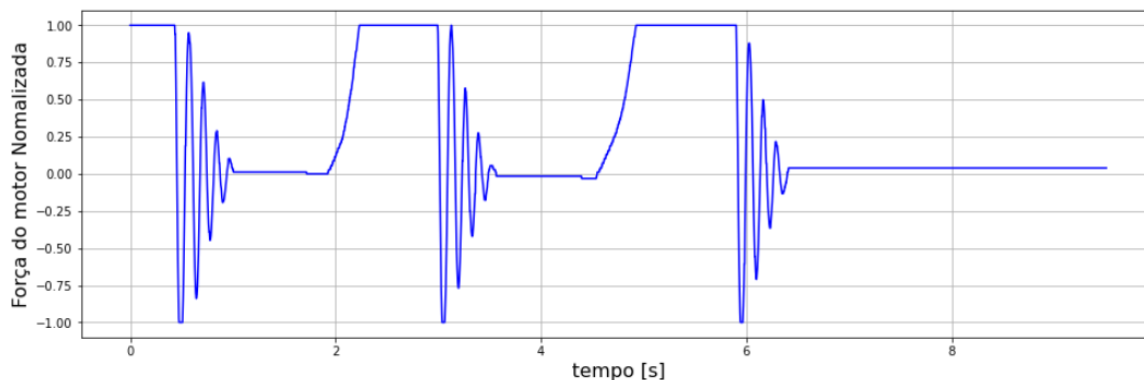
Nota-se na Figura 42 que o volante foi capaz de responder ao sinal senoidal aplicado na entrada apresentando características semelhantes à entrada aplicada ao mesmo. Observa-se também no gráfico que por volta do instante de 7 segundos foi aplicada propositalmente uma perturbação ao sistema, deslocando a atuação do volante em termos de posição angular.

Além do efeito causado pela inserção da perturbação no sistema, foi observada uma tendência maior de rotação do volante para o sentido horário (menor grau de posição no gráfico) o que pode ser causado por algum aspecto físico da montagem mecânica do simulador que facilita a rotação para um lado em detrimento do outro.

No último ensaio dinâmico realizado, o sistema foi posto em malha fechada com realimentação negativa e ganho proporcional igual a 15 (Figura 40, diagrama inferior). Os ensaios foram realizados fixando-se a referência em 0 e rotacionando o volante manualmente e, logo após, permitindo que o mesmo retornasse sozinho à posição nula através do controle proporcional.

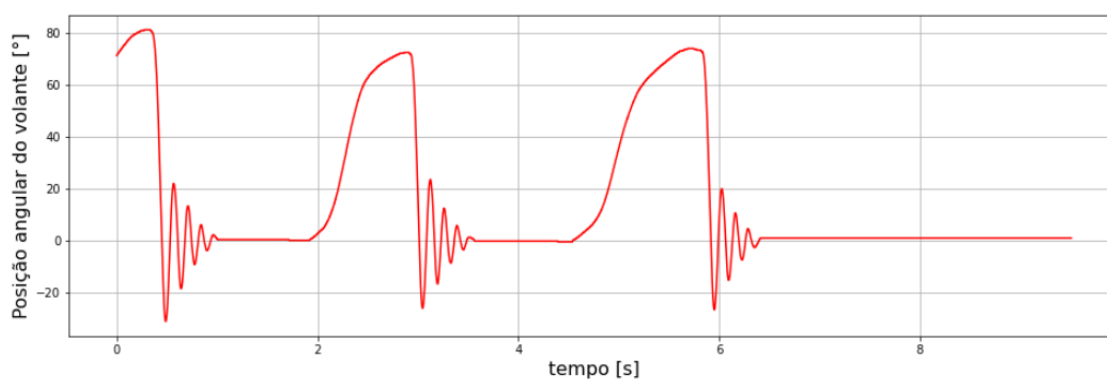
Os gráficos de atuação do motor e resposta do volante podem ser observados nas figuras 43 e 44.

Figura 43 – Atuação do sistema em malha fechada para uma referência em 0.



Fonte: Próprio autor.

Figura 44 – Resposta do volante em malha fechada.



Fonte: Próprio autor.

É importante ressaltar que o ensaio em malha fechada foi realizado utilizando o valor de posição do volante normalizado para 1 ser o valor equivalente a uma volta completa do volante ( $360^\circ$ ). Nos gráficos destacados os valores estão descritos na forma de graus para melhor compreensão.

Observa-se nos gráficos que a movimentação manual do volante chegou a causar a saturação do sistema de atuação (entre os instantes 2 a 3s e 5 a 6s da Figura 43). A resposta do volante, por sua vez, se mostrou rápida e pela sua característica subamortecida caracteriza um sistema dinâmico de ordem 2 ou superior (quando posto em malha fechada). Nota-se também a capacidade do sistema de retornar à posição 0 após as perturbações apresentando valores de erro de regime permanente pequenos o bastante para não serem visualmente relevantes.

#### 4.4. Comunicação e aplicação do simulador

Tendo em vista os resultados obtidos nos sistemas de aquisição de dados e atuação, foi possível realizar testes práticos do simulador utilizando softwares de simulação disponíveis no mercado.

Com todos os sistemas do simulador operacionais, o software desenvolvido foi capaz de funcionar com um tempo de ciclo médio de 870  $\mu$ s. Ao longo do projeto alguns softwares de simulação (Automobilista 2, por exemplo) apresentaram travamentos e mal funcionamento quando o tempo de ciclo médio era próximo ou superior a 1500  $\mu$ s. Dessa forma, o resultado de desempenho do firmware desenvolvido é satisfatório visto sua capacidade de atuar com altas taxas de amostragem e assim, minimizando atrasos de resposta e comunicação entre a interface simulador-computador.

O simulador foi testado utilizando-se alguns dos principais softwares de simulação presentes no mercado. Os softwares utilizados foram: iRacing, Assetto Corsa, Assetto Corsa Competizione e Automobilista 2.

Em todos os casos, por ser reconhecido apenas como um dispositivo HID genérico, do tipo Joystick, pelo computador, foi necessária apenas a configuração de cada tipo de controle (atribuição dos pedais, rotação do volante, etc) para que fosse possível a utilização do simulador junto aos softwares.

Nos softwares utilizados, o simulador foi capaz de ser devidamente reconhecido sendo possível o envio de comandos bem como a aquisição dos dados referentes ao *force feedback* do ambiente de simulação, permitindo assim uma aplicação completa de todos os elementos presentes no simulador.

## 5. CONCLUSÃO

O presente trabalho objetivou primariamente o desenvolvimento de um simulador automobilístico com *force feedback* capaz de ser utilizado em softwares de simulação modernos.

Tendo em vista os resultados apresentados na Seção 4, é notável que o projeto alcançou totalmente seus objetivos gerais e específicos.

Em um primeiro momento, os resultados obtidos ao fim do projeto mostraram o êxito na construção mecânica do simulador. A estrutura mecânica do simulador foi capaz de abrigar e dar suporte físico à implementação de todos os sensores e atuadores utilizado no projeto.

A implementação dos sensores, por sua vez, foi bem sucedida proporcionando uma adequada leitura da posição dos pedais e do volante do simulador. O bom desempenho da aplicação dos sensores se deu, em parte, pela utilização de recursos de processamento digital de sinais e algoritmos de correção posicional baseados em referências físicas dos elementos de instrumentação utilizados.

Os resultados relacionados ao sistema de atuação também se mostraram satisfatórios. Através de ensaios dinâmicos realizados no simulador, foi possível constatar a capacidade do sistema de atuação de aplicar forças ao eixo do volante, replicar movimentos dos sinais de entrada, bem como posicionar o volante em referenciais a partir de uma configuração em malha fechada.

No uso prático o simulador pôde ser aplicado comunicando adequadamente com todos os softwares de simulação testados. Em cada um dos softwares utilizados o simulador foi capaz de enviar informações de condução e receber os dados de *force feedback*, bem como reproduzi-los através do sistema de atuação.

## REFERÊNCIAS

GIMENEZ, S. P. **Microcontroladores 8051**. 1. ed. São Paulo: Pearson Universidades, 2010. p.4.

AXELSON, JANET L. **USB Complete: The Developer's Guide**. 4. ed. Madison: Lakeview Research, 2009. P. 180-182,278-293.

HALLIDAY, D; RESNICK, R; WALKER, J. **Fundamentos de Física. Eletromagnetismo – Volume 3**. 9. ed. Rio de Janeiro: LTC, 2012. P.195-196.

RAMSDEN, R. **Hall-Effect Sensors: Theory and Applications**. 2. ed. Oxford: Newnes, 2006. P.1-2.

BCD Semiconductor Manufacturing Limited. **LINEAR HALL-EFFECT IC AH49E**. 2008. Disponível em: < <https://www.digchip.com/datasheets/parts/datasheet/867/AH49E-pdf.php>>. Acesso em: 20 nov. 2022.

Incze, J. J; Szabó, C.; Imecs, M. Modeling and Simulation of an Incremental Encoder Used in Electrical Drives. In: International Symposium of Hungarian Researchers on Computational Intelligence and Informatics, 10., 2009, Cluj-Napoca. **10 th International Symposium of Hungarian Researchers on Computational Intelligence and Informatics**. Budapeste: Budapesti Műszaki Főiskola, 2009. p. 97-99.

CHAPMAN, Stephen J. **Fundamentos de Máquinas Elétricas**. 5. ed. Porto Alegre: AMGH, 2013. p. 464-564.

AHMED, A. **Eletrônica de Potência**. 1. ed. São Paulo: Pearson Prentice Hall, 2000. p. 357-359, 366-369.

PINHEIRO, C. A; MACHADO, J. B; FERREIRA, L. H. **Sistemas de Controles Digitais e Processamento de Sinais: Projetos, Simulações e Experiências de Laboratório**. Rio de Janeiro: Editora Interciência Ltda.: 2017. p. 213-223.

LATHI, B. P. **Signal Processing and Linear Systems**. Oxford: Oxford University Press, 2000. p. 729-731.

MICROCHIP, Technology Inc. **PIC18F2455/2550/4455/4550 DATASHEET**.

Disponível em: < <http://ww1.microchip.com/downloads/en/DeviceDoc/39632e.pdf>>.

Acesso em: 20 nov. 2022.

ATMEL, Corporation. **ATmega16U4/ATmega32U4**. Disponível em: < [https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7766-8-bit-AVRATmega16U4-32U4\\_Datasheet.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7766-8-bit-AVRATmega16U4-32U4_Datasheet.pdf)>. Acesso em: 20 nov. 2022.

INFINEON, Technologies. **High Current PN Half Bridge**. Disponível em:

<<https://pdf1.alldatasheet.com/datasheet-pdf/view/152657/INFINEON/BTS7960.html>>. Acesso em: 20 nov. 2022.

NEMICON, Corporation. **OVW-2 Model**. Disponível em: <<http://e-pwr.net/PDF/nemicon/OVW2.pdf>>. Acesso em: 20 nov. 2022.

OMRON Industrial Automation. Incremental 40-mm-dia. Rotary Encoder: E6B2-C. 2017. Disponível em: <[https://www.ia.omron.com/data\\_pdf/cat/e6b2-c\\_ds\\_e\\_6\\_1\\_csm491.pdf](https://www.ia.omron.com/data_pdf/cat/e6b2-c_ds_e_6_1_csm491.pdf)>. Acesso em: 20 nov. 2022.