

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

Matheus Ferreira Costa

CONSTRUÇÃO E CONTROLE DE UM SISTEMA DO TIPO PÊNDULO INVERTIDO  
SOBRE DUAS RODAS



Divinópolis, MG  
2022

Matheus Ferreira Costa

CONSTRUÇÃO E CONTROLE DE UM SISTEMA DO TIPO PÊNDULO INVERTIDO  
SOBRE DUAS RODAS

Monografia de Trabalho de Conclusão de Curso  
apresentada ao Colegiado de Graduação em Engenharia Mecatrônica como parte dos requisitos exigidos para a obtenção do título de Engenheiro Mecatrônico.

Áreas de integração: Controle e Mecânica.

Orientador: Prof. Dr. Luís Filipe Pereira Silva

Divinópolis, MG  
2022

Matheus Ferreira Costa

CONSTRUÇÃO E CONTROLE DE UM SISTEMA DO TIPO PÊNDULO INVERTIDO  
SOBRE DUAS RODAS

Monografia de Trabalho de Conclusão de Curso  
apresentada ao Colegiado de Graduação em En-  
genharia Mecatrônica como parte dos requisitos  
exigidos para a obtenção do título de Engenheiro  
Mecatrônico.

Áreas de integração: Controle e Mecânica.

Comissão Avaliadora:

Prof. Dr. Luís Filipe Pereira Silva  
CEFET/MG *Campus* Divinópolis

Prof. Dr. Lucas Silva de Oliveira  
CEFET/MG *Campus* Divinópolis

Prof. Dr. Jean Carlos Pereira  
CEFET/MG *Campus* Divinópolis

Divinópolis, MG  
2022

AO MEU PAI.

# Agradecimentos

Agradeço,

- aos meus avós João Custódio (vô Ló) e Maria das Dores (vó Lia), que sempre cuidaram de mim, da minha mãe e meus irmãos quando mais precisamos. Além de tudo, agradeço por me apoiarem e terem feito eu seguir o caminho dos estudos.
- à minha mãe Cintia Aparecida e meu padrasto Roni, por sempre me apoiarem, ajudarem e serem os exemplos ao qual me espelho. Eu amo vocês incondicionalmente.
- à minha namorada Ruanna, pelo companheirismo inigualável e por sempre acreditar em mim e me fazer acreditar que sou capaz o tempo todo. Com você eu tenho certeza e clareza dos meus objetivos. Obrigado por aguentar todos esses anos de estudos ao meu lado e sempre me apoiar nesse meu sonho, amo você.
- ao Professor Dr. Luís Filipe Pereira Silva, pelo suporte, orientação e pelo compartilhamento de conhecimento para este trabalho e outros para a vida como a “bastter.com”.
- ao professor Dr. Jean Carlos Pereira, que me identifiquei bastante e pude trabalhar em projetos de pesquisa ao longo do curso, inclusive este que antes de se tornar um trabalho final de curso foi tema de pesquisa.
- aos meus irmãos Ana Paula e Yuri, que são tudo para mim. Eu, por ser o mais velho, tenho o sentimento de sempre cuidar de vocês, então é isso que sempre tento fazer. E por mais que temos nossas brigas de irmãos, sou muito sortudo por ter vocês em minha vida.
- à turma 8, por todos os “burgões”, pelos momentos de descontração e pelas amizades que levarei para sempre no coração. Vocês tornaram a caminhada mais leve e divertida.
- ao meu companheiro Delgado, principalmente, durante este trabalho. Foram algumas reuniões bem longas após o fim de trabalho e você sempre se prontificou a me ajudar.
- ao meu amigo de longa data Mateus Pereira, em especial, por ter emprestado sua fresa e seu galpão mecânico para construção da estrutura física da planta.
- aos meus colegas de turma, Duda, Maria Vitória e Kesley, que hoje está se tornando uma grande amizade. Obrigado por confiarem que eu seria capaz, me apoiarem e até por emprestarem uma ponte H aos 45 do segundo tempo.
- ao corpo docente e funcionário do CEFET-MG *Campus* Divinópolis, pela colaboração e atenção.

Que todos os nossos esforços estejam sempre focados no desafio à impossibilidade. Todas as grandes conquistas humanas vieram daquilo que parecia impossível.

Charles Chaplin

# Resumo

O sistema robotizado de pêndulo invertido se tornou largamente conhecido através do Segway nas últimas décadas, inspirando diversas pesquisas por sistemas de controle eficazes e robustos. Paralelamente, a versão autônoma do veículo também tem atraído a atenção da comunidade acadêmica por se tratar de um sistema não linear e instável em malha aberta. Dessa maneira, este trabalho teve por objetivo a construção e o controle de um sistema do tipo pêndulo invertido sobre duas rodas. Para isso, foram necessários vários estudos a respeito de controladores que consigam trabalhar em sistemas não lineares. Uma das técnicas de controle amplamente utilizadas em sistemas não lineares e com múltiplos estados é o LQG, que advém da teoria do controle ótimo. Este controlador, diferente de muitos que há na literatura, é bastante robusto o que faz com que o mesmo lide bem com ruídos e distúrbios. Após análise e desenvolvimento do projeto CAD, realizou-se a construção física da planta. De forma a simplificar os cálculos, descartou parâmetros como fricção e ruídos da modelagem matemática. As técnicas utilizadas na modelagem nos fornece inicialmente um modelo não linear contínuo no tempo, sendo necessário a realização da linearização utilizando equações Jacobianas e obtendo assim o espaço de estados do sistema. Como a placa que comandará a planta é um microcontrolador, realizou-se a discretização do sistema. Após isso, obteve-se o vetor de ganhos ótimos do LQR e do estimador. As primeiras validações se deu em ambiente simulado aplicando os ganhos do controlador na equação não linear contínua obtida na etapa da modelagem. Com mais essa etapa finalizada, realizou-se testes. Primeiramente, foi desenvolvido um código no Arduino em conjunto com Processing e VS Code, sendo possível validar os sinais do sensor e os motores. Após as validações, desenvolveu-se um código completo no Arduino para controle da planta.

Palavras-chave: Segway. Pêndulo Invertido Sobre Duas Rodas. Controle Ótimo. Microcontrolador. LQG.

# Abstract

The inverted pendulum robotic system has become widely known through the Segway in recent decades, inspiring several researches for effective and robust control systems. At the same time, the autonomous version of the vehicle has also attracted the attention of the academic community because it is a non-linear and unstable open-loop system. Thus, this work aimed to build and control an inverted pendulum-type system on two wheels. For this, several studies were needed about controllers that can work in non-linear systems. One of the control techniques widely used in nonlinear and multi-state systems is LQG, which comes from the theory of optimal control. This controller, unlike many in the literature, is quite robust, which makes it handle noise and disturbances well. After analysis and development of the CAD project, the physical construction of the plant was carried out. In order to simplify the calculations, parameters such as friction and noise were excluded from the mathematical modeling. The techniques used in the modeling initially provide us with a nonlinear model continuous in time, being necessary to carry out the linearization using Jacobian equations and thus obtaining the state space of the system. As the board that will control the plant is a microcontroller, the system was discretized. After that, the vector of optimal gains of the LQR and the estimator was obtained. The first validations took place in a simulated environment by applying the controller gains in the continuous nonlinear equation obtained in the modeling stage. With this step completed, tests were carried out. First, a code was developed in Arduino together with Processing and VS Code, being possible to validate the sensor signals and the motors. After the validations, a complete code was developed in Arduino to control the plant.

Key-words: Segway. Two-Wheeled Inverted Pendulum. Microcontroller. Optimal Control. LQG.



# Sumário

<b>Lista de Figuras</b>	<b>xii</b>
<b>Lista de Tabelas</b>	<b>xiii</b>
<b>Lista de Acrônimos e Notação</b>	<b>xiv</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Definição do Problema . . . . .	1
1.2 Motivação . . . . .	2
1.3 Objetivos do Trabalho . . . . .	2
1.3.1 Objetivo Geral . . . . .	2
1.3.2 Objetivos específicos . . . . .	2
1.4 Organização do Documento . . . . .	3
<b>2 Fundamentos</b>	<b>4</b>
2.1 Revisão de Literatura . . . . .	4
2.1.1 Trabalhos Similares . . . . .	6
2.2 Fundamentos Clássicos da Mecânica . . . . .	9
2.3 Linearização . . . . .	10
2.3.1 Ponto de equilíbrio . . . . .	10
2.3.2 Linearização em torno de um ponto de equilíbrio . . . . .	10
2.4 Sistema em Espaço de Estados . . . . .	11
2.4.1 Caso contínuo no tempo . . . . .	11
2.4.2 Caso discreto no tempo . . . . .	11
2.5 Realimentação de Estados . . . . .	12
2.5.1 Controlabilidade . . . . .	12
2.5.2 Observabilidade . . . . .	12
2.6 Controlador Linear Gaussiano Quadrático (LQG) . . . . .	13
2.6.1 Regulador Linear Quadrático (LQR) . . . . .	13
2.6.2 Estimador de Estados Quadrático (LQE) . . . . .	14
<b>3 Sistema Mecatrônico</b>	<b>17</b>
3.1 Estrutura Adotada . . . . .	17
3.1.1 Parâmetros Mecânicos da Estrutura . . . . .	18
3.2 Instrumentação . . . . .	18
3.2.1 Arduino Nano . . . . .	18
3.2.2 Driver Motor CC . . . . .	19
3.2.3 Atuador . . . . .	21

3.2.4	Transdutor de Orientação . . . . .	22
3.2.5	Bateria . . . . .	25
3.2.6	Representação da Montagem do Circuito Eletrônico . . . . .	26
3.3	<i>Softwares</i> . . . . .	26
3.3.1	SolidWorks . . . . .	27
3.3.2	MATLAB . . . . .	27
3.3.3	Arduino IDE . . . . .	27
3.3.4	Processing . . . . .	27
3.3.5	Visual Studio Code (VS Code) . . . . .	27
3.3.6	<i>Fritizing</i> . . . . .	28
3.4	Custo dos Materiais Adquiridos . . . . .	28
<b>4</b>	<b>Modelagem do Sistema e Projeto do Controlador LQG</b>	<b>29</b>
4.1	Modelagem do Motor de Corrente Contínua . . . . .	29
4.1.1	Estimando as Constantes Mecânica $k_m$ e Elétrica $k_e$ do Motor . . . . .	30
4.2	Modelagem do Sistema baseada na Metodologia de Euler-Lagrange . . . . .	31
4.3	Projeto do Controlador de Estabilização . . . . .	34
4.3.1	Controlador LQR . . . . .	35
4.3.2	Estimador de Estados (Filtro de Kalman) . . . . .	37
<b>5</b>	<b>Resultados Práticos</b>	<b>42</b>
5.1	Montagem da Planta . . . . .	42
5.2	Validação do Sensor e Motores . . . . .	43
5.2.1	Validação do Sensor . . . . .	43
5.2.2	Validação dos Motores . . . . .	44
5.3	Aplicando o Controlador na Planta . . . . .	45
<b>6</b>	<b>Considerações Finais</b>	<b>47</b>
6.1	Conclusões . . . . .	47
6.2	Propostas de Trabalhos Futuros . . . . .	48
	<b>Referências</b>	<b>49</b>



# Lista de Figuras

2.1	Segway X2 SE (SEGWAY, 2019). . . . .	6
2.2	Protótipo Pêndulo Invertido (PEDROSO; MODESTO, 2017). . . . .	7
2.3	Protótipo renderizado (MELO, 2017). . . . .	8
2.4	Implementação do controlador LQR (ENGIN, 2018). . . . .	8
2.5	Planta física construída (ENGIN, 2018). . . . .	9
2.6	Estrutura de um controlador LQG. . . . .	13
2.7	Estrutura do Filtro de Kalman discreto em Espaço de Estados. Adaptado de BRUNTON <i>et al.</i> (2017). . . . .	14
3.1	Estrutura real do pêndulo na visão frontal e lateral. . . . .	17
3.2	Visão da placa responsável pela leitura dos sinais do sensor e processamento dos sinais de controle. . . . .	19
3.3	Visualização comercial do <i>driver</i> Ponte H L298N. . . . .	20
3.4	Representação de um sinal PWM. . . . .	21
3.5	Conjunto completo do atuador composto por motor, roda, suporte de acoplamento da roda ao motor, suporte de acoplamento do motor à base e parafusos. . . . .	21
3.6	A placa comercial é vista em (a) e os detalhes dos eixos do sensor de orientação utilizado para medição da inclinação do robô é visto em (b). . . . .	23
3.7	Detalhamento de forma gráfica dos 3 eixos do acelerômetro. . . . .	24
3.8	Bateria Lipo de duas células ( $7.4 V_{cc}$ ) com capacidade de $2200 mAh$ e taxa de carga e descarga $20 C$ . . . . .	25
3.9	Representação do circuito eletrônico montado para controle da posição angular do pêndulo. . . . .	26
4.1	Representação esquemática do circuito elétrico do motor CC (SILVA, 2017). . . . .	29
4.2	Diagrama corpo livre do pêndulo invertido. Adaptado de (SILVA, 2017). . . . .	31
4.3	Círculo unitário de pólos e zeros do modelo discretizado. . . . .	35
4.4	Modelo não linear realimentado pelo controlador <i>LQR</i> . . . . .	36
4.5	Respostas dos estados do sistema para a condição inicial estabelecida. . . . .	37
4.6	Sinal de controle do sistema para a condição inicial estabelecida. . . . .	37
4.7	Modelo não linear realimentado pelo controlador <i>LQG</i> . . . . .	38
4.8	Detalhamento do controlador <i>LQG</i> . . . . .	38
4.9	Respostas dos estados do sistema para a condição inicial estabelecida. . . . .	38
4.10	Sinal de controle do sistema para a condição inicial estabelecida. . . . .	39
4.11	Sinal do ruído branco de medição. . . . .	39
4.12	Modelo não linear realimentado pelo controlador <i>LQG</i> com inserção de distúrbio e ruídos. . . . .	40

4.13	Detalhamento do controlador $LQG$ com a inserção de distúrbio e ruído de medição. . . . .	40
4.14	Respostas dos estados do sistema para a condição inicial estabelecida. . . . .	41
4.15	Sinal de controle do sistema para a condição inicial estabelecida. . . . .	41
5.1	Vistas frontal e lateral da montagem da planta física. . . . .	42
5.2	Comportamento de forma macro do sinal do sensor filtrado, curva azul e do sinal real, curva vermelha. . . . .	44
5.3	Comportamento de forma micro dos dois sinais do sensor. . . . .	44
5.4	Trecho do código para validar os sentidos de giro dos motores. . . . .	45
5.5	Resultado da planta ao aplicar o controlador projetado. . . . .	46
5.6	Sinal de controle ao aplicar o controlador projetado. . . . .	46

# Lista de Tabelas

3.1	Parâmetros mecânicos obtidos através de medições práticas e via cálculos. . .	18
3.2	Dados do fabricante sobre a placa Arduino Nano. . . . .	19
3.3	Parâmetros mecânicos da estrutura. . . . .	20
3.4	Dados do fabricante em relação ao motor. . . . .	22
3.5	Parâmetros da roda obtidos experimentalmente. . . . .	22
3.6	Principais características da placa MPU6050 (INVENSENSE, 2019). . . . .	23
3.7	Parâmetros do giroscópio: velocidade de giro e nível de sensibilidade. . . . .	24
3.8	Parâmetros do giroscópio: velocidade de giro e nível de sensibilidade. . . . .	25
3.9	Consumo dos Motores CC. . . . .	26
3.10	Tabela de custo dos materiais utilizados na construção do protótipo. . . . .	28
5.1	Pinagem entre Arduino e Ponte H L298N. . . . .	45

# Lista de Acrônimos e Notação

LQR	Linear Quadratic Regulator (regulador linear quadrático)
LQE	Linear Quadratic Estimator (estimador linear quadrático)
LQG	Linear Quadratic Gaussian (linear gaussiano quadrático)
LTI	Linear Time-Invariant (linear invariante no tempo)

$A$	notação para matrizes (letras maiúsculas do alfabeto latino)
$\in$	símbolo “contido em”
$\forall$	quantificação universal (“para todos; para qualquer; para cada”)
$\mathbb{R}$	conjunto dos números reais
$\mathbb{E}$	símbolo para expectância ou esperança de uma variável aleatória
$\mathbf{I}$	matriz identidade de dimensão apropriada
$n$	especialmente utilizada para representar a ordem uma matriz quadrada
$\int$	símbolo de integral
$\sum$	notação de somatório
$\alpha$	símbolos utilizados para simplificar os termos do modelo (letras gregas)

# Introdução

Neste capítulo introdutório tem como finalidade a apresentação do tema, os objetivos do trabalho realizado e relata-se a sequência em que foi desenvolvido. Primeiramente, a situação problema será apresentada. Em seguida, as principais motivações e objetivos propostos neste trabalho. Após esses pontos e para finalizar, a estrutura dos capítulos dessa dissertação será apresentada.

## 1.1 Definição do Problema

A realização do controle do pêndulo invertido, seja qual for sua variação, é um dos exemplos mais importantes na teoria de controle moderno. Sua estrutura resulta em um sistema não linear bastante complexo. Isso dificulta bastante a utilização de técnicas clássicas de teoria de controle.

O controle de sistemas não linear é uma técnica na qual um procedimento ou processo é conseguido sem a interferência humana. Para realizar a mesma, é preciso que um programa de instruções seja escrito combinado a um sistema de controle que executa as tarefas. De acordo com GROOVER (2013), um processo automatizado necessita de energia não só para operar a condução do processo como para operar o programa e o sistema de controle. Ademais, considera-se uma grande aliada da otimização do desempenho, já que com essa tecnologia é possível conhecer indicadores que auxiliam a gestão, acelera os processos e remove trabalhos repetitivos, dispensáveis e não aceitáveis ergonomicamente.

Uma grande área em ascensão da robótica é a robótica móvel. Sendo assim, os robôs se locomovem dentro de um ambiente qualquer de diversas formas, com rodas, esteiras, pernas mecânicas, dentre outras. Estes robôs são caracterizados pela capacidade de se deslocar, podendo ser de modo guiado, semiautônomo ou totalmente autônomo (JUNG *et al.*, 2005). Algumas de suas mais variadas aplicações são: aspiração de pó, entrega de alimentos, vigilância predial, busca e salvamento, etc.

Assim sendo, neste trabalho, realizou-se o desenvolvimento de um veículo do tipo pêndulo invertido sobre duas rodas, que pode ser classificado como um robô do tipo móvel. Para a

estabilização na vertical deste sistema, utilizou-se de técnicas de controle robustas que irão conseguir tratar as simplificações e considerações de seu modelo matemático.

## 1.2 Motivação

O estudo da Teoria de Controle em sistemas dinâmicos e a implementação de controladores em uma planta física, complexa e com características não lineares, além da concretização de um projeto completo é de **suma importância** na consolidação dos conceitos assimilados ao longo da formação acadêmica de um profissional da área de Mecatrônica.

Além de tudo, após o estudo de três disciplinas seguidas da área de controle e a vontade do autor de conseguir realizar o controle de uma planta clássica, porém com um nível de dificuldade elevada bem como a construção de um projeto puramente mecatrônico que aborda as quatro grandes áreas: mecânica, controle, computação e eletrônica.

## 1.3 Objetivos do Trabalho

Aqui é descrito de forma sucinta o objetivo geral e os objetivos específicos do trabalho. Fazendo-se cumprir esses objetivos específicos, espera-se alcançar o objetivo geral.

### 1.3.1 Objetivo Geral

Construir um sistema denominado pêndulo invertido sobre duas rodas e controlar a posição de sua estrutura para que a mesma fique paralela com a vertical ( $0^\circ$  ou 0 radianos) utilizando controlador ótimo.

### 1.3.2 Objetivos específicos

Listam-se os seguintes objetivos específicos para o projeto completo:

- desenvolver por meio do SolidWorks o conceito do protótipo;
- **levantar** a lista e o custo dos materiais;
- obter o modelo não linear do sistema e **linearizar a partir de técnicas jacobianas**;
- discretizar o modelo linear obtido;
- projetar o controlador LQG com base no modelo linear;
- avaliar a resposta do controlador aplicando ao modelo não linear;
- realizar a construção da planta física;
- realizar a calibração do sensor e testes nos componentes eletrônicos da planta;



- validar o modelo obtido;
- implementar/embarcar o controlador no sistema.

## 1.4 Organização do Documento

Este documento é dividido em quatro capítulos além das considerações finais.

O presente capítulo, apresenta a definição do problema que foi motivo de estudo e a motivação pela qual deu origem ao desenvolvimento do trabalho. Além do mais, apresenta os objetivos gerais e específicos definidos para que o projeto em questão seja realizado. Por fim, discorre sobre estado da arte do presente tema.

No segundo capítulo são apresentados os fundamentos, que consistem na revisão de literatura e fundamentação teórica, esse tem por objetivo histórica e teórica do trabalho.

O terceiro capítulo consiste em apresentar o sistema mecatrônico, descrevendo a parte estrutural do protótipo. É neste capítulo também que é realizado as justificativas dos materiais escolhidos, suas características. Ao final, é apresentado a tabela de custo total.

O capítulo seguinte, o quarto deste trabalho, é encontrado o modelo não linear para o pêndulo sobre duas rodas. Em seguida, é feito a linearização e discretização do modelo para que o projeto do controlador seja possível. Em seguida, em ambiente de simulação, implementa o controlador ao sistema não linear e apresenta os resultados obtidos.

O quinto capítulo é dedicado basicamente a implementação do controlador projetado a planta real.

Por último, no capítulo de considerações finais, é descrito as conclusões deste trabalho, detalhando as dificuldades encontradas para realização do mesmo e propostas futuras.

# Fundamentos

Este capítulo tem como objetivo apresentar a parte histórica e teórica deste trabalho. A revisão de literatura fica a cargo do conteúdo histórico, descrevendo assim uma linha temporal do tema deste TCC. Na fundamentação teórica, serão descritos de forma clara as teorias mais importantes e necessárias para a realização do projeto, como por exemplo as técnicas de modelagem de sistemas e projeto do controlador ótimo.

## 2.1 Revisão de Literatura

Segundo OGATA (2010), trabalhos importantes como de Minorsky, Nyquist e Hazen, datados entre as décadas de 20 e 30, contribuíram para o progresso da teoria de controle. Minorsky, em 1922, trabalhou com pilotagem de embarcações utilizando controladores automáticos e demonstrou por meio de equações diferenciais a estabilidade do sistema. Em 1932, Nyquist desenvolveu um procedimento no qual determina a estabilidade de sistemas em malha fechada com base na resposta em malha aberta. E Hazen, em 1934, introduziu o termo *servomecanismos* para sistemas de controle de posição e analisou o projeto de servomecanismos a relé.

No século XX, acontecimentos como a Segunda Guerra Mundial estimularam as pesquisas em sistemas de controle. No final dos anos 50, a teoria de controle já era bastante consolidada, sendo que o carro chefe era o método que utilizavam a resposta em frequência e com muitas aplicações industriais, como por exemplo: a utilização de controladores PID para o controle de pressão, temperatura, etc. Os métodos de resposta em frequência e do lugar das raízes, os quais são a base da teoria clássica de controle, conduziram os sistemas que são estáveis e satisfatórios para um determinado conjunto de condições. Esses sistemas são aceitáveis, porém não são ótimos no sentido literal do termo.

Entre as décadas de 60 e 80, segundo OGATA (2010), o controle ótimo de sistemas determinísticos, controle adaptativo e de aprendizagem foi altamente pesquisado. Diferentemente da teoria clássica, estes métodos se enquadram na teoria de controle moderno. Essa teoria se baseia no domínio do tempo em sistemas de equações diferenciais. Como vantagem, temos

que essa teoria simplificou a realização de projetos de sistemas de controle já que se baseia no modelo de um sistema de controle real. Em contrapartida, a estabilidade é sensível ao erro entre o sistema real e do seu modelo tal como na teoria clássica. Dessa forma, quando o controlador projetado for aplicado no sistema, o mesmo poderá ficar instável devido principalmente ao erro de modelagem. Esse problema é solucionado estabelecendo uma série de possíveis erros no sistema para depois projetar o controlador. Sendo assim, se o erro estiver dentro da predição o sistema será sempre estável. A essa metodologia de projeto baseado nesse princípio é chamado teoria do controle robusto.

Em OOI (2003) a técnica utilizada para o controle do pêndulo foi o LQR. Também desenvolveu-se um controlador por alocação de polos para a estabilização do sistema. Dessa forma, foi capaz realizar a comparação entre esses dois métodos e saber qual dos dois foi mais eficaz. Pelas simulações concluiu-se que os dois métodos atenderam, com ressalva de que o controlador LQR ofereceu maior confiabilidade.

A técnica *PID Backstepping*, que é não linear, foi utilizada por NGUYEN GIA MINH THAO; PHUCAND (2010) para controlar um pêndulo invertido sobre duas rodas (PIDR). Essa técnica consiste basicamente em uma estrutura de controle que possui três malhas de controle, sendo elas: 1) a malha principal, desenvolvida a partir da técnica não linear *backstepping* que manterá o pêndulo em equilíbrio; 2) a segunda malha que utiliza um PD para controlar a posição do robô e; 3) um controlador PI para o controle de movimentação.

Em JUNFENG; WANYING (2011) novamente é feito uma comparação entre o controlador ótimo LQR com o controlador por alocação de polos. Contudo, neste artigo, conclui-se que o *overshoot* foi menor por meio da técnica de alocação de polos e que o tempo de estabilização foi quase comparável ao do LQR.

No artigo de JUANG; LURRR (2013), os autores propuseram a desenvolver um protótipo utilizando o Arduino como microcontrolador e a implementação de duas técnicas de controle, sendo elas o PID e LQR na forma de PI-PD. Os resultados obtidos deixam bem claro que a técnica PI-PD foi muito mais eficaz em todos sentidos do que a utilização pura do PID. Como concluído, a estabilização por PID é marginal e as oscilações angulares excedem o limite máximo de torque que os motores podem oferecer. Já no PI-PD ou LQR a estabilidade é alcançada e ainda o controlador é capaz de compensar o desalinhamento do CG, fazendo com que o robô retorne para a posição angular desejada.

Em PAULA (2014), o autor utilizou duas técnicas de controle discreto, LQR com ação integral e alocação de polos. A primeira técnica utilizada teve como objetivo a estabilização do pêndulo e a segunda o controle de velocidade dos motores. A técnica de alocação de polos além de realizar o controle dos motores, teve como objetivo interfacear o controle de ângulo e o sinal enviado para os motores.

Por último, ALVES (2018), além de projetar um controlador LQR, apresenta duas técnicas de linearização, sendo elas: a jacobina e a por realimentação. A modelagem do sistema baseou-se na equação Lagrangiana e após encontrar as equações de movimento, passou-se

as mesmas para o Espaço de Estados. Os resultados obtidos foram satisfatório, contudo, ao sofrer interferências, como distúrbio e ruídos o sistema passa a ser mais oscilante.

Como dissertado acima, há vários trabalhos com diferentes resultados para o mesmo tema. Por se tratar de um sistema naturalmente instável, não linear e bastante complexo para controlar, é lógico que haverão trabalhos com resultados distintos. Entretanto, quando há domínio de um determinado assunto, é possível transformar os resultados teóricos em algo para a sociedade, como por exemplo o *Segway*, que é um veículo elétrico de transporte humano e seu funcionamento é baseado no pêndulo invertido sobre duas rodas.

O *Segway*, visto na Figura 2.1, inventado por Dean Karmen e revelado em dezembro de 2001 tem o funcionamento idêntico ao do pêndulo invertido, com adição que o mesmo está sobre duas rodas. A função básica deste veículo é o transporte de uma única pessoa. O sistema entende que deve-se locomover para frente ou mesmo frear, devido à inclinação que o passageiro impõe sobre o veículo. Hoje em dia, existem vários modelos e cada um com uma especificação diferente um do outro.



Figura 2.1: Segway X2 SE (SEGWAY, 2019).

Logo abaixo, segue uma série de trabalhos similares a este aqui proposto. Será descrito de forma breve um pouco sobre os trabalhos em si e a estratégia de controle utilizada, bem como a apresentação da planta física construída.

### 2.1.1 Trabalhos Similares

O trabalho de PEDROSO; MODESTO (2017) consistiu na construção de um protótipo projetado por meio de um *software* CAD, visto na Figura (2.2), e no desenvolvimento de um sistema de controle para esse projeto. Os autores se engajaram em controlar o sistema com um controlador PID, obtendo seus coeficientes por meio do método de sintonização de *Ziegler-Nichols*. Como sensor utilizaram a placa MPU6050, que possui embutida na mesma o giroscópio e acelerômetro. Contudo, se faz necessário a fusão dos sinais provenientes desses dois sensores para obtenção de uma maior confiabilidade e certeza da medição. Sendo assim,

os autores optaram por utilizar o filtro complementar, que nada mais é do que um filtro de média.

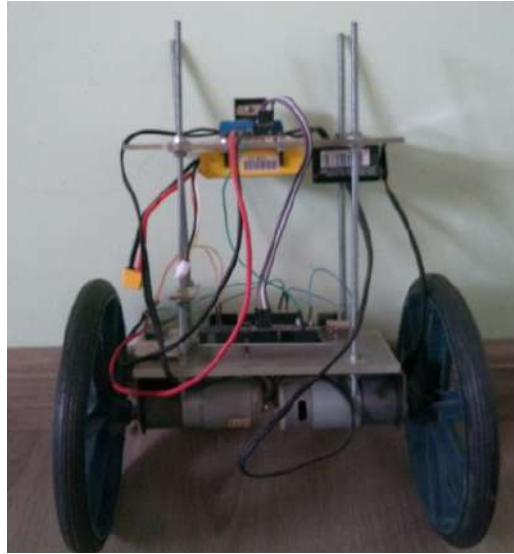


Figura 2.2: Protótipo Pêndulo Invertido (PEDROSO; MODESTO, 2017).

O artigo de MELO (2017) teve como objetivo a identificação e o controle de um veículo do tipo Segway para fins educacionais. Dessa maneira, o autor dividiu o artigo em partes como: o sistema eletrônico e mecânico, linearização do modelo matemático e discretização, implementação de filtros para a leitura dos sinais provindo do acelerômetro e giroscópio, identificação do sistema e a implementação de um controlador PD (Proporcional-Derivativo), sendo que seus coeficientes foram obtidos por meio de inspeção gráfica utilizando o lugar das raízes. O diagrama de malha fechada padrão, com o controlador em série com o sistema. Os resultados obtidos foram apenas em cima do modelo, sendo assim, não houve a implementação física ou mesmo a montagem do protótipo proposto, como visto na Figura (2.3). Os resultados obtidos na simulação foram bem satisfatórios para os critérios adotados, sendo que em malha fechada o autor conseguiu fazer com que o sistema seguisse a referência senoidal aplicada e após aplicação de um distúrbio, o sistema conseguiu se recompor rapidamente.

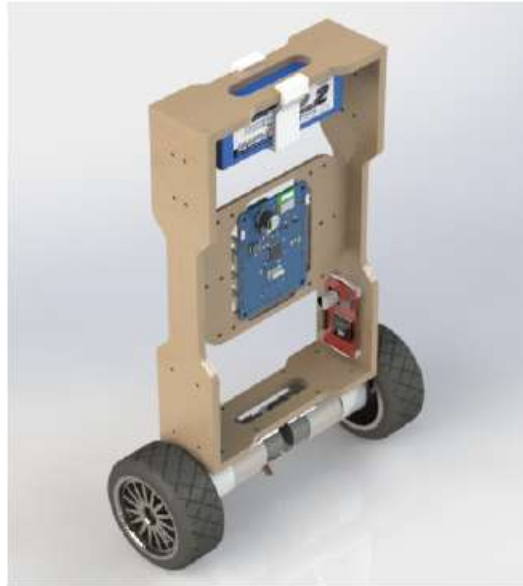


Figura 2.3: Protótipo renderizado (MELO, 2017).

Por último, o artigo de ENGIN (2018) propôs a implementação de um controlador LQR. A modelagem do sistema foi feita utilizando equações lagrangianas e, depois que encontrou uma equação não linear para o sistema, linearizou a mesma e passou para a forma de espaço de estados, chegando assim nas matrizes A, B, C e D. Aplicou a essas matrizes a propriedade de controlabilidade e observabilidade, concluindo assim, que é um sistema controlável e observável. O pêndulo invertido sobre duas rodas, é um sistema de uma única entrada e múltiplas saídas (SIMO). Dessa forma, para lidar com esse tipo de sistema é mais simples com o LQR baseado pelo controle de velocidade como propõe o autor. Para utilizar esse controlador, é necessário encontrar um vetor de ganhos  $K$  e consequentemente, minimizará a função de custo  $J$ . Na Figura (2.4) é mostrada a implementação do controlador LQR.

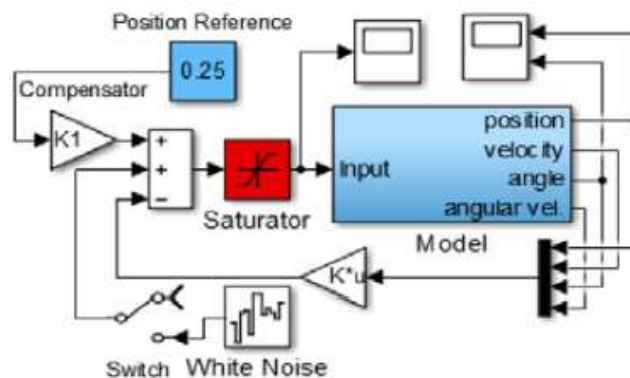


Figura 2.4: Implementação do controlador LQR (ENGIN, 2018).

O filtro que foi implementado também foi o complementar, sendo um filtro passa-baixas para o acelerômetro e um passa-altas para o giroscópio. As principais variáveis que foram medidas ou estimadas foram: a posição da estrutura,  $\theta$ , a velocidade angular,  $\dot{\theta}$  e, velocidade da roda,  $\omega$ . O protótipo montado pode ser visto na Figura (2.5).



Figura 2.5: Planta física construída (ENGIN, 2018).

## 2.2 Fundamentos Clássicos da Mecânica

O sistema em estudo conforme já citado é descrito como instável e não linear. Porém para encontrar as equações de movimento que regem o sistema, é preciso escolher um tipo de método. Para o trabalho em questão, escolheu-se representar o sistema por meio da segunda lei de Newton descrita como sendo

$$\sum \vec{F}_i = m \vec{a} \quad (2.1)$$

$$\sum \tau_{iz} = J \alpha_z \quad (2.2)$$

em que os termos do lado esquerdo das Equações (2.1) e (2.2) representam todas as forças ( $F_i$ ) e torques ( $\tau$ ) que atuam sobre o sistema. Já no lado direito da Equação (2.1), temos a massa total do corpo representada no centro de massa ( $m$ ) e todas as acelerações lineares ( $a$ ). Na Equação (2.2), as acelerações agora são as angulares representadas por ( $\alpha_z$ ) e o momento de inércia ( $J$ ) é dado como sendo

$$J = \sum m_i r_i^2 \quad (2.3)$$

em que  $m_i$  é a massa e  $r$  a distância do objeto até o eixo de rotação.

Um outro fundamento que precisa ser descrito é o cálculo do centro de gravidade (CG) ou centro de massa (CM) de um corpo. Para um sistema tridimensional pode-se definir um vetor posição desse sistema. Sendo uma partícula com coordenadas  $x_i$ ,  $y_i$  e  $z_i$  seu vetor posição  $\vec{r}$  é definido por

$$\vec{r} = x_{cm} \hat{i} + y_{cm} \hat{j} + z_{cm} \hat{k}$$

no qual os vetores  $\hat{i}$ ,  $\hat{j}$ ,  $\hat{k}$  são unitários e apontam na direção dos eixos  $x$ ,  $y$  e  $z$  respectivamente. E finalmente, o cálculo da posição do centro de massa de um sistema de partículas é definido como

$$\vec{CM} = \frac{1}{M} \sum_{i=1}^n m_i \vec{r}_i \quad (2.4)$$

em que  $M$  é a massa total do sistema.

## 2.3 Linearização

Como já dito, o sistema trabalho aqui neste traalho é de natureza não linear. Assim, para que as técnicas clássicas de controle ou mesmo algumas de controle moderno sejam utilizadas nos mesmos é preciso tratá-los por aproximações linearizantes.

Segundo OGATA (2010), um sistema para ser denominado de não linear é preciso que o princípio da superposição falhe ao ser aplicado nele. Dessa forma, caso tente obter uma resposta do sistema aplicando duas entradas simultâneas considerando as mesmas individualmente e somando-se os resultados, essa resposta não será válida.

Um sistema representado no espaço de estados é definido por HESPANHA (2009) da seguinte maneira

$$\begin{aligned}\dot{x} &= f(x, u) \\ y &= g(x, u)\end{aligned}\tag{2.5}$$

sendo  $x \in \mathbb{R}^n$ ,  $y \in \mathbb{R}^m$  e  $u \in \mathbb{R}^k$ .

### 2.3.1 Ponto de equilíbrio

Para se obter uma aproximação linear de um sistema não linear, é preciso, primeiramente, determinar os pontos de equilíbrio do sistema. A determinação dos pontos de equilíbrio é feito quando se iguala o termo diferencial da Equação (2.5).

Pela definição de HESPANHA (2009), têm-se que um par de pontos  $(x^{eq}, u^{eq}) \in \mathbb{R}^n \times \mathbb{R}^k$  que pode ser chamado de ponto de equilíbrio de (2.5) caso  $f(x^{eq}, u^{eq}) = 0$ . A solução de (2.5) é visto abaixo.

$$u(t) = u^{eq}, \quad x(t) = x^{eq}, \quad y(t) = y^{eq} := g(x^{eq}, u^{eq}), \quad \forall t \geq 0 \tag{2.6}$$

### 2.3.2 Linearização em torno de um ponto de equilíbrio

De acordo com HESPANHA (2009), a linearização do sistema em série de Taylor pode ser representado por matrizes Jacobianas, conforme descrito a seguir.

$$\begin{aligned}A &:= \frac{\partial f(x^{eq}, u^{eq})}{\partial x}, & B &:= \frac{\partial f(x^{eq}, u^{eq})}{\partial u}, \\ C &:= \frac{\partial g(x^{eq}, u^{eq})}{\partial x}, & D &:= \frac{\partial g(x^{eq}, u^{eq})}{\partial u}\end{aligned}\tag{2.7}$$

Assim, o sistema linear invariante no tempo (LTI), no espaço de estados, linearizado em torno de um ponto de equilíbrio  $(x^{eq}, u^{eq})$ , é dado por:

$$\begin{aligned}\dot{\delta x} &= A\delta x + B\delta u \\ \delta y &= C\delta x + D\delta u\end{aligned}\tag{2.8}$$

em que os termos  $\delta x(t)$ ,  $\delta y(t)$  e  $\delta u(t)$  para todo  $t \geq 0$  é descrito como

$$\delta x(t) := x(t) - x^{eq}, \quad \delta y(t) := y(t) - y^{eq}, \quad \delta u(t) := u(t) - u^{eq} \tag{2.9}$$



## 2.4 Sistema em Espaço de Estados

A representação no espaço de estados garante uma maior facilidade de soluções de um modelo matemático e de um sistema físico, já que a mesma pode relacionar um conjunto de variáveis de entradas, saídas e de estados por meio de equações diferenciais de primeira ordem. A representação fornece de maneira prática as soluções do sistema, por adotar notações matriciais quando o sistema dinâmico é linear e invariante no tempo.

### 2.4.1 Caso contínuo no tempo

No caso contínuo no tempo, um sistema representado no espaço de estados de acordo com HESPANHA (2009) pode ser descrito como

$$\begin{aligned}\dot{x}(t) &= A(t)x(t) + B(t)u(t) \\ y(t) &= C(t)x(t) + D(t)u(t)\end{aligned}\tag{2.10}$$

no qual os sinais  $u$ ,  $y$  e  $x$  das Equação (2.10) são chamados de entrada, saída e de estados respectivamente e definidos por

$$u : [0, \infty] \rightarrow \mathbb{R}^k, \quad x : [0, \infty] \rightarrow \mathbb{R}^n, \quad y : [0, \infty] \rightarrow \mathbb{R}^m$$

Sendo o sistema representado pela Equação (2.10) linear e as matrizes  $A(t)$ ,  $B(t)$ ,  $C(t)$  e  $D(t)$  constantes  $\forall t \geq 0$ , o sistema é chamado de linear invariante no tempo (LTI). Assim, pode-se simplificar a representação no caso contínuo

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{2.11}$$

As dimensões das matrizes  $A$ ,  $B$ ,  $C$  e  $D$  das Equações (2.10) e (2.11) são as mesmas e são definidas como sendo

$$\begin{aligned}A &\in \mathbb{R}^{n \times n} & B &\in \mathbb{R}^{k \times m} \\ C &\in \mathbb{R}^{l \times n} & D &\in \mathbb{R}^{l \times m}\end{aligned}$$

### 2.4.2 Caso discreto no tempo

As aplicações de forma geral são feitas utilizando microcontroladores ou algum dispositivo digital. Sendo assim, para aplicações desse tipo, faz-se necessário a discretização do sistema.

CHEN (1999) afirma que a representação discreta do sistema no espaço de estados é descrita como

$$\begin{aligned}x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C_d x_k + D_d u_k\end{aligned}\tag{2.12}$$

em que a variável  $t = kT$  com  $k = 0, 1, 2, \dots, k_f$  e  $T$  o período de amostragem do sistema. O período é escolhido de forma a ser o menor possível, ou seja, o máximo que o microcontrolador consegue trabalhar. As matrizes  $A_d$ ,  $B_d$ ,  $C_d$ ,  $D_d$  são descritas como se segue bem como a solução para  $A_d$  e  $B_d$  (CHEN, 1999, p. 91)

$$A_d = e^{AT} \quad B_d = \left( \int_0^T e^{A\tau} d\tau \right) B \quad C_d = C \quad D_d = D\tag{2.13}$$

## 2.5 Realimentação de Estados

Uma das formas de se implementar controladores por meio da representação no espaço de estados é por meio da realimentação de estados, no qual utiliza-se os próprios estados do sistema com um fator multiplicativo (ganho) para gerar o sinal de controle. Para ter acesso a todos os estados da planta deve-se ter sensores para cada um dos estados do sistema. Como na maioria das aplicações reais isso não ocorre, pois implica em um custo elevado, faz-se necessário a implementação de um sistema que irá estimar os estados (observador) e assim, o sinal de controle passará a ser com relação a esses estados estimados. Contudo, para realizar a construção de um observador ou mesmo fazer o controle do sistema, deve-se, primeiramente, testar o sistema com métodos matemáticos que informam tais capacidades. Dito isso, os critérios são apresentados a seguir para um sistema discreto

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C_d x_k + D_d u_k \end{aligned} \quad (2.14)$$

### 2.5.1 Controlabilidade

De acordo com HESPANHA (2009), para saber se o sistema é controlável ou não, utiliza-se as matrizes  $A_d$  e  $B_d$  da equação de estados de (2.14). O critério para determinar a controlabilidade é demonstrado como sendo

$$\mathbb{C} := \begin{bmatrix} B_d & A_d B_d & A_d^2 B_d & \dots & A_d^{n-1} B_d \end{bmatrix}_{n \times (kn)} \quad (2.15)$$

A partir do critério (2.15) pode-se determinar se o sistema em questão poderá ou não ser controlável. Segundo OGATA (2010), para que o sistema seja controlável é preciso que o **determinante** da matriz (2.15) seja diferente de 0, ou seja, que tenham vetores linearmente independentes (LI) e assim, posto  $n$ .

### 2.5.2 Observabilidade

Como não haverá medições diretas de todos os estados, será necessário a utilização de um observador como já mencionado. O mesmo irá estimar o valor dos estados que faltam e assim a lei de controle da realimentação será baseado neles.

Contudo, nem todos os sistemas são observáveis, dessa forma é preciso garantir que o sistema deste trabalho seja observável. De acordo com HESPANHA (2009), o critério no qual testa-se o sistema será observável ou não é mostrado como sendo

$$\mathbb{O} := \begin{bmatrix} C_d \\ C_d A_d \\ C_d A_d^2 \\ \vdots \\ C_d A_d^{n-1} \end{bmatrix}_{(kn) \times n} \quad (2.16)$$

A análise da matriz (2.16) é dual ao critério de controlabilidade. Para que o sistema seja observável é preciso que os vetores sejam LI, ou seja, que tenham posto (*rank*) igual a  $n$ .

## 2.6 Controlador Linear Gaussiano Quadrático (LQG)

De acordo com BRUNTON *et al.* (2017), o controlador LQG é constituído por um estimador quadrático linear (LQE) ou mais popularmente denominado de Filtro de Kalman, ou seja, um observador no qual os estados estimados são realimentados através de ganhos ótimos projetados por meio do regulador linear quadrático (LQR). A Figura (2.6) apresenta como deve ser o *design* de um controlador LQG.

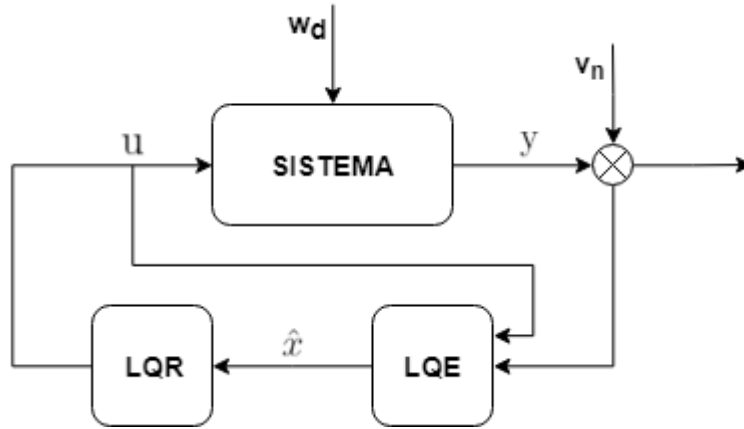


Figura 2.6: Estrutura de um controlador LQG.

### 2.6.1 Regulador Linear Quadrático (LQR)

O LQR possui como principal característica a garantia de estabilidade em malha fechada, um alto grau de robustez do sistema controlado proporcionado ao mesmo margem de ganho infinita e margem de fase  $\pm 60^\circ$  (FERNANDES, 2014). Contudo, se faz necessário conhecer todos os estados do sistema para se trabalhar com este regulador. Dessa maneira, é usual utilizar este controlador juntamente com um estimador tal como o LQE.

Para encontrar o ganho ótimo do LQR, considera-se o seguinte sistema no tempo discreto.

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C_d x_k \end{aligned} \quad (2.17)$$

Assume-se que  $x(0) = x_0$  e  $k = 0, 1, 2, \dots, k_f$ . De acordo com SAGE (1977), a função de custo a ser minimizada é dada por:

$$J = \frac{1}{2} \|x_{k_f}\|_S^2 + \frac{1}{2} \sum_{k=0}^{k_f-1} \{ \|x_k\|_Q^2 + \|u_k\|_R^2 \} \quad (2.18)$$

A matriz  $Q \in \mathbb{R}^{n \times n}$  é uma matriz positiva semi-definida e  $R \in \mathbb{R}^{m \times m}$  é uma matriz positiva definida. Essas matrizes são responsáveis pela ponderação dos estados e o sinal de controle do sistema. De acordo com MATOS (2008), essas matrizes são escolhidas da seguinte forma:

$$Q = C_d^T C_d$$

$$R = \rho \mathbf{I}_{m \times m}, \quad \rho > 0.$$

Conforme SAGE (1977), a equação Hamiltoniana é dada como se segue.

$$H_k = \frac{1}{2} x_k^T Q x_k + \frac{1}{2} u_k^T R u_k + \lambda_{k+1}^T [A_d x_k + B_d u_k] \quad (2.19)$$

Caso o par  $(A_d, B_d)$  seja controlável e o par  $(A_d, C_d)$  observável, a lei de controle ótimo que minimiza a equação de custo (2.18) é dada por:

$$u_k = -K x_k \quad (2.20)$$

em que a matriz  $K \in \mathbb{R}^{m \times n}$  de ganhos ótimos do controlador LQR é definida como se segue

$$K = (R + B_d^T P B_d)^{-1} B_d^T P A_d \quad (2.21)$$

no qual a matriz simétrica  $P \in \mathbb{R}^{n \times n}$  é conseguida após solucionar a equação de Riccati de forma recursiva como segue

$$P_{k+1} = Q + A^T P_k A_d - A_d^T P_k B_d (R + B_d^T P_k B_d)^{-1} B_d^T P_k A_d \quad (2.22)$$

### 2.6.2 Estimador de Estados Quadrático (LQE)

SUNDIN; THORSTENSSON (2012) afirmam que os o filtro capaz de fazer a estimação da derivada do giroscópio é o Filtro de Kalman. O Filtro de Kalman surgiu da teoria de otimização e é um estimador ótimo para sistemas que possuem distúrbios e apresentam características de distribuição normal de ruído branco. O filtro modela o comportamento do sensor e isso inclui o *bias* que possa vir a ter, o que dá uma boa estimativa do ângulo e do *bias* do giroscópio, que melhora a taxa angular uma vez que o mesmo pode ser removido. A Figura (2.7) apresenta a estrutura do filtro no tempo discreto.

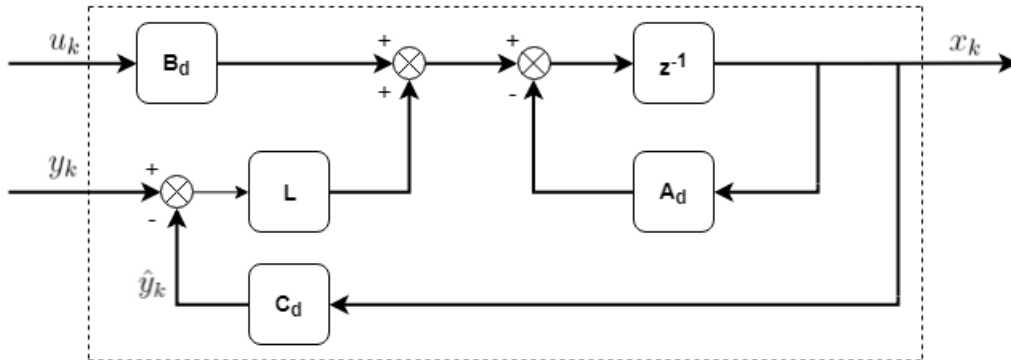


Figura 2.7: Estrutura do Filtro de Kalman discreto em Espaço de Estados. Adaptado de BRUNTON *et al.* (2017).

O ganho do estimador ótimo quadrático linear se baseia no desempenho do observado projetado com a presença de erro de medição e ruído de processo. Logo, a cada tempo que passa o ganho do filtro de Kalman vai sendo atualizado. Primeiramente, para realizar o projeto desse estimador considera-se o seguinte sistema

$$\begin{aligned} x_{k+1} &= A_d x_k + B_d u_k + w_k \\ y_k &= C_d x_k + v_k \end{aligned} \quad (2.23)$$

em que o ruído  $w_k$  e o erro de medição  $v_k$  são sequências aleatórias de ruído branco gaussiano com média zero (SAGE, 1977, p. 231). Sendo assim, temos

$$\mathbb{E}\{w_k\} = 0 \quad \text{e} \quad \mathbb{E}\{v_k\} = 0$$

em que  $\mathbb{E}\{\}$  é conhecido como **expectância** ou valor esperado. A correlação por se tratar de ruído branco é zero

$$\text{cov}\{w_i w_j^T\} = 0 \quad \text{e} \quad \text{cov}\{v_i v_j^T\} = 0 \quad \text{para} \quad i \neq j$$

e a covariância é definida por

$$\text{cov}\{w_k w_k^T\} = Q_o \quad \text{e} \quad \text{cov}\{v_k v_k^T\} = R_o$$

A principal tarefa da otimização é determinar uma matriz de ganhos que minimizará a variância da estimação do erro, que é denotado por  $G_k$

$$G_k = \mathbb{E}\{(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T\} \quad (2.24)$$

Segundo BRUNTON *et al.* (2017), um estimador produz uma estimação dos estados  $x_k$  conhecida como  $\hat{x}_k$  caso conheça a saída medida  $y_k$  provinda de um sensor e o sinal de entrada  $u_k$ . Caso o sistema seja observável, é possível construir o filtro a partir do ganho  $L$  como se segue

$$\begin{aligned} \hat{x}_{k+1} &= A_d \hat{x}_k + B_d u_k + L_k (y_k - \hat{y}_k) \\ \hat{y}_k &= C_d \hat{x}_k \end{aligned} \quad (2.25)$$

sendo que a saída  $\hat{y}_k$  é a predição da expectativa da saída do sensor baseada no estados estimados  $\hat{x}_k$ . Manipulando a Equação (2.25), encontra-se uma expressão para  $\hat{x}_{k+1}$ .

$$\begin{aligned} \hat{x}_{k+1} &= (A_d - LC_d) \hat{x}_k + B_d u_k + L y_k \\ &= (A_d - LC_d) \hat{x}_k + \begin{bmatrix} B_d & L \end{bmatrix} \begin{bmatrix} u_k \\ y_k \end{bmatrix} \end{aligned} \quad (2.26)$$

assim, as matrizes  $A_f$  e  $B_f$  do filtro de Kalman são:

$$A_f = (A_d - LC_d) \quad B_f = \begin{bmatrix} B_d & L \end{bmatrix} \quad (2.27)$$

De acordo com KIM *et al.* (2013), o ganho  $L$  do filtro de Kalman é um vetor pertencente aos números reais e de dimensão  $n \times 1$  e é definido pela seguinte expressão

$$L = A_d S_k C_d^T (C_d S_k C_d^T + R_o)^{-1} \quad (2.28)$$

na qual a matriz  $S$  é obtida quando soluciona-se a equação de Riccati para a equação de diferenças. A expressão que precisa-se solucionar e que fará com que o ganho  $L$  seja ótimo é descrita como

$$S_{k+1} = Q_o + A_d S_k A_d^T - A_d S_k C^T (C_d S_k C_d^T + R_o)^{-1} C_d S_k A_d^T \quad (2.29)$$

em que  $Q_o$  é uma matriz positiva definida e  $R_o > 0$  são especificados como parâmetros de projeto, considerando a condição inicial  $S_0 = 0$ .

## Sistema Mecatrônico

De maneira geral, a estrutura do robô possui forma retangular com duas rodas. As rodas são acopladas de forma que fiquem paralelas uma a outra. A estrutura foi projetada para possui três andares e em cada um deles estão alguns componentes. A Figura (3.1) mostra a vista isométrica da estrutura.

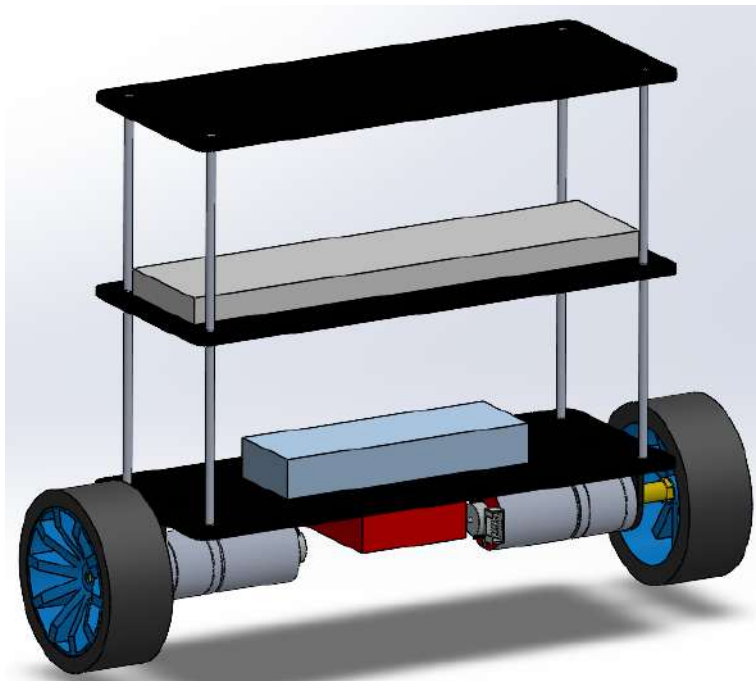


Figura 3.1: Estrutura real do pêndulo na visão frontal e lateral.

### 3.1 Estrutura Adotada

A estrutura do pêndulo consiste em quatro barras rosqueadas que serve para sustentar as prateleiras em seus devidos lugares. O material utilizado para fazer as bases do pêndulo foi o ACM. Foi adotado uma altura de  $200\text{ mm}$  para a planta medindo do chão ao topo, a largura de  $80\text{ mm}$  que é o padrão das placas com uma espessura de  $3\text{ mm}$ . Os componentes do sistema foram distribuídos em cada andar. Na parte inferior da primeira estante ou no ponto mais próximo ao solo temos o *driver* dos motores e os motores propriamente dito.

Contudo, foi necessário a utilização de um suporte que fixasse os motores a base. Já no segundo andar ou na segunda prateleira, estão o microprocessador Arduino Nano e o sensor que capturará o ângulo de inclinação da planta. Por fim, para deixar o centro de massa um pouco mais distante do eixo de referência que passa entre os motores, posicionou-se a bateria na última placa. Em muitas literaturas aconselha-se adotar esse tipo de estratégia para facilitar o controle do pêndulo invertido.

### 3.1.1 Parâmetros Mecânicos da Estrutura

A Tabela (3.1) apresenta os parâmetros mecânicos da estrutura do pêndulo bem como seus respectivos valores. Tais parâmetros foram medidos ou mesmo calculados como mostra o Apêndice ().

Tabela 3.1: Parâmetros mecânicos obtidos através de medições práticas e via cálculos.

Descrição da grandeza	Variável	Valor Nominal	Unidade de medida
Momento de Inércia do Pêndulo	$J_p$	0.00530429	$kg.m^2$
Momento de Inércia das Rodas	$J_w$	0.00005013	$kg.m^2$
Massa total da planta física	$m$	0.7740	$kg$
Massa do pêndulo s/rodas	$M_p$	0.7120	$kg$
Massa de duas rodas	$M_r$	0.0620	$kg$
Altura do centro de massa ao eixo	$z_{cm}$	0.0836	$m$
Raio externo das rodas	$r_{ext}$	0.0315	$m$
Raio interno das rodas	$r_{int}$	0.0250	$m$
Distância eixo até o conjunto-placa 1	$d_{h1}$	0.0150	$m$
Distância eixo até o conjunto-placa 2	$d_{h2}$	0.120	$m$
Distância eixo até o conjunto-placa 3	$d_{h3}$	0.180	$m$
Distância do chão até o topo	$H$	0.200	$m$
Largura dos conjuntos-placa	$W$	0.080	$m$
Resistência de armadura do motor	$R$	0.91	$\Omega$

## 3.2 Instrumentação

No momento em que se inicia o planejamento de um projeto de engenharia, é de suma importância realizar a listagem dos recursos que serão utilizados, bem como a justificativa de cada. Desta forma, segue abaixo a listagem dos *softwares* e materiais que foram ordenados da seguinte maneira: atuadores, transdutor de orientação e eletrônica.

### 3.2.1 Arduino Nano

O Arduino Nano é um microcontrolador de baixo custo e de alta performance. Essa placa será responsável por realizar o controle do sistema, fazer a leitura dos sinais do sensor e enviar os sinais de controle para o atuador. A placa pode ser vista na Figura (3.2).





Figura 3.2: Visão da placa responsável pela leitura dos sinais do sensor e processamento dos sinais de controle.

A decisão da utilização dessa placa ao invés de outro microcontrolador seu deu pelo fato de que a mesma conseguiria suprir os requisitos para o projeto, como a quantidade mínima de pinos PWM e digitais e a capacidade de processamento dos dados. A partir do *datasheet* da placa, é possível conhecer as características principais que a compõe, como visto abaixo.

Tabela 3.2: Dados do fabricante sobre a placa Arduino Nano.

Descrição	Valores Nominais
Chip	ATmega328P 8 bit AVR
Tensão de Operação	5 $V_{cc}$
Tensão de Alimentação ( $V_{in}$ )	7~12 $V_{cc}$
Pinos Digitais	14 (D0 - D13)
Pinos Digitais PWM	D3, D5, D6, D9, D11 (8 bit)
Pinos Analógicos	8 (A0 - A7)
Corrente Máxima p/Pino	40 $mA$
Memória Flash	32 $kB$
Clock	16 $MHz$

### 3.2.2 Driver Motor CC

A Figura (3.3) apresenta o *driver* do motor CC que será utilizado, a Ponte H L298N. Com essa placa será possível realizar o controle de sentido de giro e realizar o controle da velocidade dos dois motores por meio de sinais PWM (*Pulse Width Modulation*).

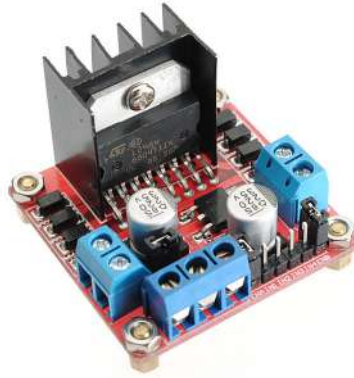


Figura 3.3: Visualização comercial do *driver* Ponte H L298N.

Todas as características dessa placa se encontram na folha de dados do fabricante e algumas das mais importantes a serem consideradas podem ser vistas na Tabela (3.3).

Tabela 3.3: Parâmetros mecânicos da estrutura.	
Descrição	Valores Nominais
Tensão de Operação	$4 \sim 35 V_{cc}$
Tensão Lógica	$5V$
Corrente de Operação Máxima	$2A$ por canal
Corrente Lógica	$0 \sim 36 mA$
Dimensões (C×L×A)	$43 \times 43 \times 27 mm$

Como mencionado, com essa placa realiza-se duas coisas importantes, mudança no sentido de giro e a alteração da velocidade dos motores. O primeiro caso é realizado mudando o sentido da corrente, pois os motores CC giram a partir de um campo magnético que é provocado por uma corrente que passa em suas bobinas. Já no segundo caso temos a variação de velocidade por meio do sinal PWM, na qual a tensão de saída que chegará nos terminais do motor será uma tensão média que sempre dependerá do ciclo de trabalho daquele instante.

Como mencionado, com essa placa realiza-se duas coisas importantes, mudança no sentido de giro e alteração da velocidade dos motores. O primeiro caso é realizado mudando o sentido da corrente, pois os motores CC giram a partir de um campo magnético que é provocado por uma corrente que passa em suas bobinas.

Para controlar a velocidade do motor, é necessário recorrer ao PWM, que é uma técnica na qual se obtém resultados analógicos por meio digital. A técnica consiste em ondas quadradas de alta frequência das quais é possível controlar o percentual de tensão que é aplicado aos motores. Para realizar isto, basta controlar o tempo que deseja manter a saída em 1 (ou ligado) e o em 0 (ou desligado). Esse tempo em nível alto é chamado de *Duty Cycle* e ao utilizar desse recurso, o valor médio de tensão aplicado aos terminais do motor será alterado.

A partir da Figura (3.4), é possível obter a equação do *Duty Cycle* (DC) e da tensão

média que será de fato o valor aplicado aos motores.

$$DC = \frac{DT}{T} \quad (3.1)$$

$$U_{medio} = U_{max}(DC)$$

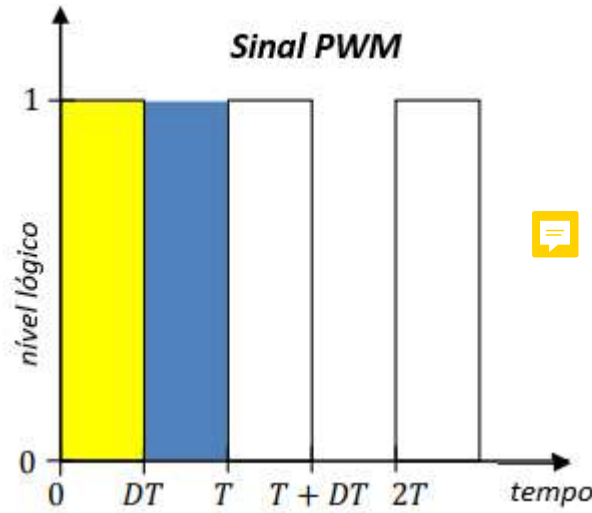


Figura 3.4: Representação de um sinal PWM.

### 3.2.3 Atuador

A Figura 3.5 mostra o atuador da planta escolhido juntamente com seus componentes complementares. O motor é responsável por receber o sinal do controle e transformar em tração, movimentando assim o pêndulo.



Figura 3.5: Conjunto completo do atuador composto por motor, roda, suporte de acoplamento da roda ao motor, suporte de acoplamento do motor à base e parafusos.

### Motores

O motor escolhido em questão opera com uma tensão de 6V contínuo com uma taxa de redução de 34:1 que faz com tenha pouco *backlash*. Possui encoder de quadratura acoplado

na parte anterior de resolução de 11 pulsos por resolução no eixo do motor, que corresponde a 341.2 pulsos por resolução na saída após a caixa de redução. Normalmente, motores pequenos e de baixo preços costumam não possuir uma folha de dados completa e, portanto, as constantes dos mesmos são desconhecidas. Mas nas especificações disponíveis há a tensão nominal ou de operação, velocidade e corrente sem carga, torque, velocidade, potência e corrente em máxima eficiência. Como o motor é de polos é possível encontrar resistência de armadura de forma direta medindo por meio de um multímetro seus terminais. Neste dois casos desconsidera indutância por se tratar de um valor muito pequeno e sua influência será praticamente nula SUNDIN; THORSTENSSON (2012).

Tabela 3.4: Dados do fabricante em relação ao motor.

Condições	Valores Nominais
Tensão Operação	$6 V_{cc}$
Resistência de armadura	$8 \Omega$
Velocidade Sem Carga	$210RPM$ (0.13A)
Eixo Travado	$0.980665Nm$ (3.2A)
Máxima Eficiência	$0.196133Nm / 170RPM / 2.0W / 0.60A$
Máxima Potência	$0.509946Nm / 110RPM / 3.1W / 1.10A$
Taxa de Redução	34:1

## Rodas

O conjunto roda e pneus utilizados possuem dimensão de  $65 \times 25mm$ . As rodas foram escolhidas desse tamanho por considerar que as mesmas **atendem a velocidade angular do motor e a velocidade desejada para o sistema**. Por sua vez, a roda é fabricada por um tipo de plástico azul brilhante e os pneus por borracha. O conjunto é acoplado ao motor através de um suporte e fixado por um parafuso bem no centro da roda. O conjunto será aproximado por um anel para o cálculo do momento de inércia e não por um cilindro maciço já que as rodas é vazada e possuem alguns rasgos.

Tabela 3.5: Parâmetros da roda obtidos experimentalmente.

Parâmetro	Valor Nominal	Unidade	Descrição
$M_r$	0.0310	$kg$	Massa de cada roda
$r_1$	0.0250	$m$	Raio interno da roda
$r_2$	0.0315	$m$	Raio externo da roda

### 3.2.4 Transdutor de Orientação

Essa *shield* possui dois sensores, acelerômetro e giroscópio, dos quais serão fundidos a fim de obter uma melhor precisão da medição da orientação da estrutura do pêndulo. A Figura (3.6) apresenta a placa física e um modelo que representa todos os 6 eixos, 3 para

o acelerômetro e 3 para o giroscópio, o que faz com a mesma possua 6 graus de liberdade (*6DOF*).

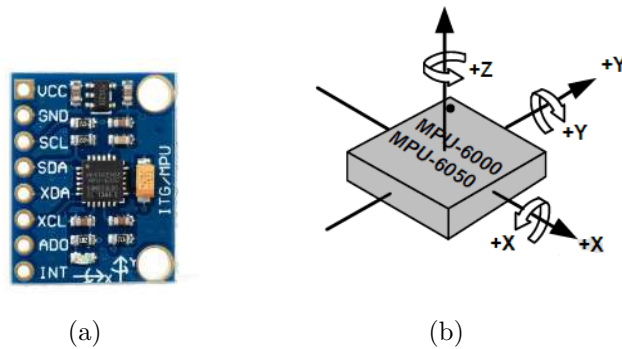


Figura 3.6: A placa comercial é vista em (a) e os detalhes dos eixos do sensor de orientação utilizado para medição da inclinação do robô é visto em (b).

A partir do *datasheet* da placa algumas características podem ser obtidas como visto na Tabela (3.6).

Tabela 3.6: Principais características da placa MPU6050 (INVENSENSE, 2019).

Descrição	Valores
Modelo	GY-521
Tensão de Operação	$3 \sim 5 V_{cc}$
Conversor ADC	16 <i>bits</i>
Comunicação	<i>I<sup>2</sup>C</i>
Tempo de Acomodação	10 <i>ms</i>
Dimensões (C×L×A)	21×17×3 <i>mm</i>

### Acelerômetro

O acelerômetro é um equipamento que é utilizado para mensurar sua aceleração própria. A aceleração própria é diferente da estabelecida através da relação entre velocidade e tempo. Sendo que esta considera a sensação de peso medida em um dado referencial. Quando considera-se o acelerômetro colocado em uma superfície plana, a medição será de aproximadamente  $9.81m/s^2$ . Na maioria dos casos, a aceleração é medida em força g, que é basicamente a aceleração sentida como peso. A Figura (3.7) mostra a inclinação de algum objeto utilizando dois eixos.

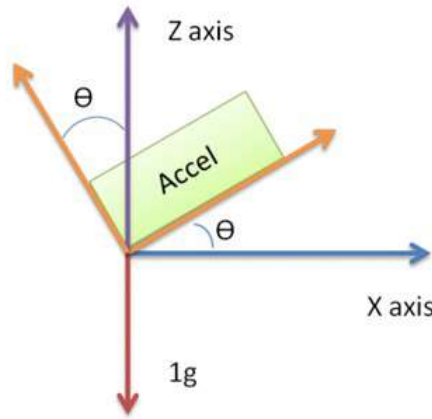


Figura 3.7: Detalhamento de forma gráfica dos 3 eixos do acelerômetro.

A medição da orientação pode ser feita de três maneiras: utilizando apenas um eixo, dois e três eixos. A forma mais simples é a primeira, porém a mais eficiente em termos de precisão é a última. As equações são mostradas logo abaixo:

$$\begin{aligned}\theta &= \sin^{-1}(x) \\ \theta &= \arctan\left(\frac{x}{z}\right) \\ \theta &= \arctan\left(\frac{x}{\sqrt{z^2 + y^2}}\right)\end{aligned}\tag{3.2}$$

### Giroscópio

O giroscópio por sua vez, é um dispositivo utilizado para manter ou medir a orientação. Um giroscópio mecânico normalmente consiste de um disco rotativo, em que os eixos ligados a ele são capazes de se deslocarem livremente em qualquer orientação.

O valor lido da MPU6050 sem nenhum tratamento pode ser chamado de valor bruto já que não representa nenhuma grandeza física e para realizar a conversão para um valor usual, é necessário saber/configurar o quão rápido o sensor está girando em graus por segundo (*dps*). A Tabela (3.8), com conteúdo encontrado no *datasheet* do fabricante, relaciona cada velocidade de giro do sensor com a sua sensibilidade, sendo que essas constantes são chamadas de ganho e devem ser divididas por 1000.

Tabela 3.7: Parâmetros do giroscópio: velocidade de giro e nível de sensibilidade.

Faixa (°/s)	Sensibilidade LSB/(°/s)
±250	131
±500	65.5
±1000	32.8
±2000	16.4

### 3.2.5 Bateria

A escolha da bateria de Lipo de 2200mAh, como visto na Figura (3.8), passou por alguns critérios, como o tempo de funcionamento e capacidade de fornecimento de energia. As características mais importantes encontradas no *datasheet* do fabricante, é visto na Tabela (3.8).

Tabela 3.8: Parâmetros do giroscópio: velocidade de giro e nível de sensibilidade.

Descrição	Valores	Unidade
Tensão	7.4	$V_{cc}$
Capacidade	2.2	$Ah$
Descarga	30	$C$
Comprimento	105	$mm$
Largura	32	$mm$
Altura	16	$mm$
Massa	121.4	$g$

O item mais relevante a considerar é se a mesma consegue fornecer a capacidade necessária de energia para o sistema. O consumo máximo de corrente que a bateria pode fornecer é diretamente proporcional a sua capacidade ampére por hora ( $Ah$ ) e sua taxa de carga e descarga, que nesse caso é de 30  $C$ . O cálculo é simples e é realizado da seguinte forma

$$\text{Corrente}(A) = 2.2Ah \times 20C = 44A \quad (3.3)$$

assim, pela Equação (3.3) a bateria vista na Figura (3.8) suporta fornecer até 44A em teoria quando solicitada. Geralmente, se trabalha com uma margem de 50% para preservá-la.



Figura 3.8: Bateria Lipo de duas células (7.4  $V_{cc}$ ) com capacidade de 2200  $mAh$  e taxa de carga e descarga 20  $C$ .

Com relação ao tempo de funcionamento, basta conhecer o valor de consumo de cada dispositivo que compõe o sistema. Para esse projeto, pode-se considerar apenas o consumo

dos motores, Tabela (3.9), já que a corrente drenada pelos outros eletrônicos são irrelevantes perante aos atuadores. Em teoria e considerando tudo em perfeitas condições, é possível obter 1 (uma) hora de funcionamento utilizando o torque máximo ou 110 min usando a máxima eficiência dos motores em descarga contínua.

Tabela 3.9: Consumo dos Motores CC.

Dispositivo	Máx. Eficiência	Máx. Torque
Motores	1.2 A	2.2 A

### 3.2.6 Representação da Montagem do Circuito Eletrônico

A representação esquemática do circuito eletrônico contendo todos os componentes detalhados aqui neste capítulo, pode ser visto na Figura (3.9). Pode-se observar através da figura como os componentes foram interligados uns aos outros. Essa é a representação final do circuito eletrônico para controle da posição angular da estrutura com a vertical.

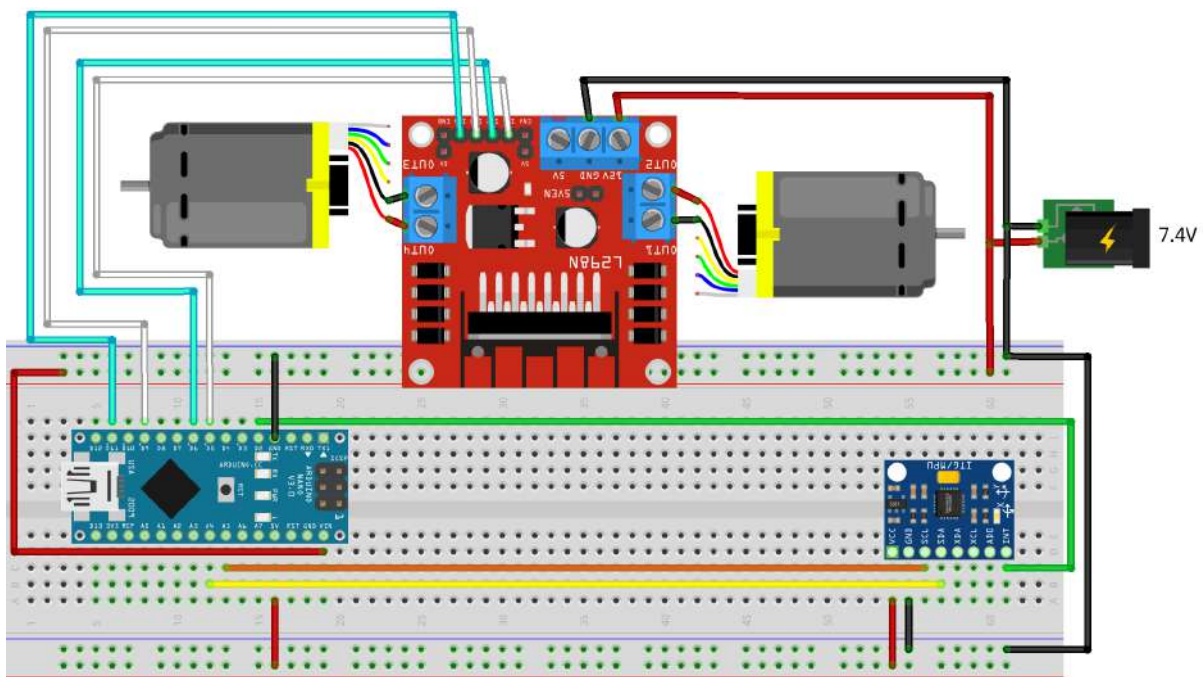


Figura 3.9: Representação do circuito eletrônico montado para controle da posição angular do pêndulo.

## 3.3 *Softwares*

A realização deste trabalho se deu através de uso de alguns *softwares* que ajudaram e facilitaram bastante. Abaixo é feito um breve resumo de cada um e para qual propósito os mesmos serviram.



### 3.3.1 SolidWorks

O SolidWorks é um *software* de CAD 3D, utilizado para modelagem de peças sólidas e para simulações de processos. Foi escolhido como plataforma para o desenvolvimento do projeto mecânico das peças da estrutura devido à experiência do autor com o programa, adquirido ao longo do curso.

### 3.3.2 MATLAB

O MATLAB é um *software* interativo de alta **performance** voltado para o cálculo numérico. Realiza a análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos em ambiente fácil, sendo programado com linguagem matemática ao invés da programação tradicional. O *software* conta com inúmeras extensões, mas com destaque para o Simulink. Essa extensão conta com uma programação baseada em diagrama de blocos e com esse *toolbox* é possível realizar as simulações dos circuitos de uma forma que fique mais simples e de fácil visualização. Foi escolhido para realização dos cálculos de parâmetros mecânicos, dos ganhos do controlador e simulação do sistema.

### 3.3.3 Arduino IDE

É um *software* gratuito, de código aberto, que facilita a gravação dos programas desenvolvidos na placa Arduino. A linguagem de programação é baseada em C. Foi utilizado no desenvolvimento do código responsável por acionar os motores de corrente contínua e também o processamento do controlador.

### 3.3.4 Processing

O Processing por sua vez é uma linguagem de programação de código aberto e possui um ambiente de desenvolvimento integrado (IDE). Contudo, ele é considerado um *sketchbook*, alternativa de organização de projetos sem ser o de um IDE padrão. Por mais que seja baseado na linguagem Java, a forma de programar não é de Java puro, a não ser que o usuário deseje. Foi utilizado neste trabalho em conjunto com o Arduino para validação dos sinais do sensor, lendo dados através da porta serial do computador.

### 3.3.5 Visual Studio Code (VS Code)

O Visual Studio Code, ou popularmente conhecido como VS Code, é um editor de código-fonte desenvolvido pela Microsoft para Windows, Linux e macOS. É customizável, permitindo que os usuários possam mudar o tema do editor, teclas de atalho e preferências. É um *software* livre e de código aberto. Foi utilizado para ler os sinais gravados em arquivo com extensão .txt do Processing utilizando linguagem python.

### 3.3.6 *Fritizing*

O *Fritizing* é uma iniciativa de código aberto para desenvolver um *software* tipo CAD para desenhos de *hardware* eletrônico, com intuito de apoiar a construção de um circuito mais permanente como a confecção de uma placa de circuito impresso (PCB). Utilizou desse *software* para a elaboração de uma visão esquemática do circuito eletrônico da planta.

## 3.4 Custo dos Materiais Adquiridos

A seguir, na Tabela (3.10), estão listados os materiais utilizados para a elaboração mecânica e eletrônica do projeto, seus respectivos preços e a quantidade necessária.

Tabela 3.10: Tabela de custo dos materiais utilizados na construção do protótipo.

Descrição	Un.	Qtd.	Valor Un.	Valor Total
Arduino Nano	pç	1	R\$ 19,50	R\$ 19,50
Kit Motor CC	pç	2	R\$ 119,90	R\$ 239,80
Driver L298 H	pç	1	R\$ 19,90	R\$ 19,90
MPU6050	pç	1	R\$ 16,90	R\$ 16,90
Bateria Lipo 2200mAh	pç	1	R\$ 130,00	R\$ 130,00
Carregador Lipo Imax B6	pç	1	R\$ 110,00	R\$ 110,00
Estrutura	-	1	R\$ 45,00	R\$ 45,00
Miscelaneas	-	1	R\$ 58,00	R\$ 58,00
<b>TOTAL</b>				<b>R\$ 639,10</b>

O item denominado de miscelâneas da lista de materiais remete aos materiais que são difíceis de contar um a um e na maioria da vezes são parafusos, porcas, *jumpers*, etc. Para encontrar seu valor, considerou-se 10% do valor total de toda a lista.

## Modelagem do Sistema e Projeto do Controlador LQG

Este capítulo abordará a descrição do modelo matemático do pêndulo invertido sobre duas rodas. Para a obtenção das equações do modelo, utilizará de Euler-Lagrange para tal fim. O eixo é fixo no centro das rodas, sendo assim, o mesmo só move quando há movimento linear das rodas. O ângulo de inclinação do pêndulo é  $\theta$ , que é a variável controlada, e a velocidade angular é  $\dot{\theta}$ .

### 4.1 Modelagem do Motor de Corrente Contínua

O circuito elétrico simplificado que modele os motores CC do projeto, pode ser visto na Figura (4.1) (SILVA, 2017).

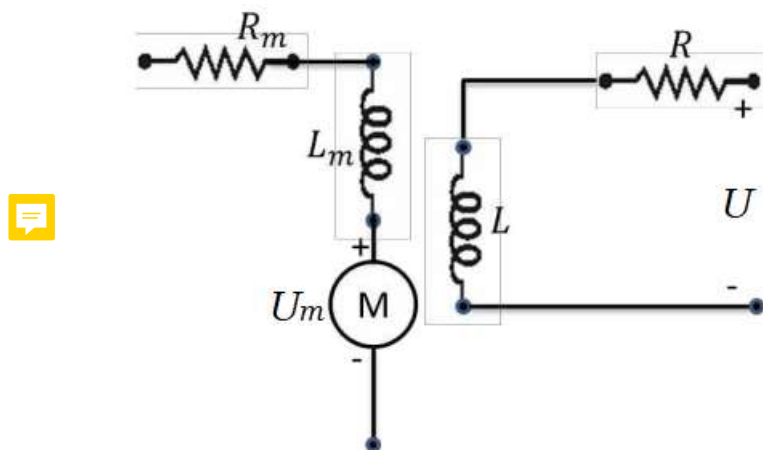


Figura 4.1: Representação esquemática do circuito elétrico do motor CC (SILVA, 2017).

A variável  $U$  representa a tensão de controle aplicada nos motores,  $R$  e  $L$  representam a resistência interna e a indutância dos enrolamentos do estator respectivamente. Aplicando a Lei das Tensões de Kirchoff (KVL), pode-se estabelecer uma relação entre as tensões internas do circuito dos motores CC considerando separadamente cada componente. Para o circuito

da Figura (4.1), obtém-se a seguinte equação.

$$U = Ri + L \frac{di}{dt} + U_m \quad (4.1)$$

Considerando a corrente elétrica do motor constante, desconsiderando reações na armadura, efeitos magnéticos secundários bem como deformidades na geometria do motor, o fluxo magnético que atravessa o entreferro do motor pode ser considerado constante e proporcional a velocidade angular do eixo do motor ( $\dot{\varphi}$ ).

$$U_m = k_e \dot{\varphi} \quad (4.2)$$

A força resultante do eixo do motor é proporcional a corrente, ao número de espiras das bobinas entre outros fatores. O torque resultante que atua sobre o eixo é a multiplicação da força pelo raio dessas bobinas enroladas. Esses valores constantes que dependem da geometria e do material denomina-se de constante mecânica do motor ( $k_m$ ) e o torque que é aplicado no eixo fica proporcional a corrente do motor CC, como mostra a Equação (4.3).

$$\tau_m = k_m i \quad (4.3)$$

Como o sinal a ser controlado da planta será o da tensão, se faz necessário encontrar a expressão do torque em termos da mesma. Assim, o torque do motor em relação a tensão de entrada  $U$  é vista na Equação (4.4) abaixo.

$$\tau_m = \frac{k_m}{R} U - \frac{k_m k_e}{R} \dot{\varphi}_{eixo} \quad (4.4)$$

Para encontrar as equações descritas acima, foram feitas várias simplificações. Um modelo mais fidedigno de um motor real consideraria várias não linearidades como por exemplo folgas mecânicas que aqui foram desconsiderados.

Existe também uma outra propriedade muito importante de um motor que é a sua zona morta (dz- *dead zone*). O conceito dessa propriedade é basicamente uma tensão abaixo da necessária que faça com que o eixo do motor entre em movimento, ou seja, é uma região de tensão entre o valor 0V e o valor mínimo de *input* que faça o motor girar.

#### 4.1.1 Estimando as Constantes Mecânica $k_m$ e Elétrica $k_e$ do Motor

A partir dos dados fornecidos pelo fabricante, é possível encontrar as constantes mecânica  $k_m$  e elétrica  $k_e$  do motor. Com os dados que estão descritos na Tabela (3.4) e com a Equação (4.4) é possível estimar os valores utilizando a Equação (4.5) abaixo.

$$\tau_m = \frac{k_m}{R} U - \frac{k_m k_e}{R} \dot{\varphi}_{eixo} = \alpha + \beta (\dot{\varphi}_{eixo}) \quad (4.5)$$

Realizando uma regressão linear com os dados fornecidos pelo fabricante, encontra-se para  $\alpha$  e  $\beta$  os seguintes valores:  $\alpha = 0.99423$  e  $\beta = -0.04462$ .

De acordo com SILVA (2017), para encontrar a constante elétrica  $k_e$ , considera-se que o torque seja praticamente nulo em velocidade máxima do eixo do motor. Dessa forma, o valor de  $k_e$  pode ser calculado conforme a Equação (4.6).

$$k_e = \frac{U}{\dot{\phi}_{eixo}} = 0.2729 \quad (4.6)$$

Com a Equação (4.3) e com os dados referente a torque vistos na Tabela (3.4), é possível se chegar em uma faixa de valores possíveis para a constante mecânica do motor  $k_m$ . Abaixo em (4.7) é visto os valores encontrados para  $k_m$ .

$$k_m(\tau_m, i) = \begin{cases} k_{eix.trav.} = \frac{\tau_m}{i} = 0.3065 \\ k_{max.efic.} = \frac{\tau_m}{i} = 0.3269 \\ k_{max.pot.} = \frac{\tau_m}{i} = 0.4636 \end{cases} \quad (4.7)$$

De forma a simplificar a escolha do valor de  $k_m$  a ser utilizado no modelo, será realizado uma média ponderada, adotando os seguintes pesos:  $P_{eix.trav.} = 1$ ,  $P_{max.efic.} = 1$  e  $P_{max.pot.} = 2$ . Resolveu-se adotar um peso diferente para o  $k_{max.pot.}$  já que seu valor está bem acima dos demais. Sendo assim, o valor final de  $k_m$  é visto abaixo.

$$k_m = \frac{0.3065 + 0.3269 + 2(0.4636)}{1 + 1 + 2} = 0.3902 \quad (4.8)$$

## 4.2 Modelagem do Sistema baseada na Metodologia de Euler-Lagrange

Na Figura (4.2) é mostrado o diagrama de corpo livre da parte estrutural do pêndulo invertido sobre duas rodas. Sendo que  $z$  a distância que passa no eixo das rodas até o centro de massa representado por  $CM$  e  $\theta$  é a variável (ângulo entre o corpo e o eixo vertical) a ser controlada.

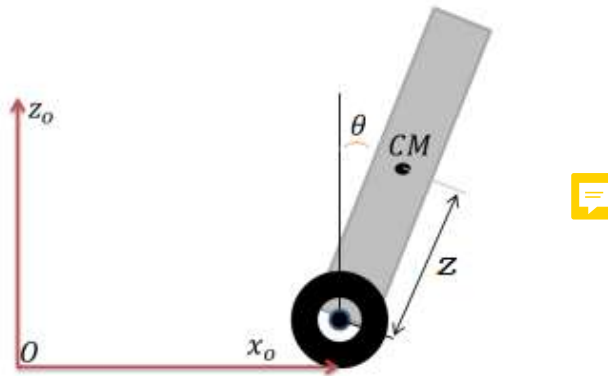


Figura 4.2: Diagrama corpo livre do pêndulo invertido. Adaptado de (SILVA, 2017).

Para a obtenção da modelagem, seguindo o princípio de Euler-Lagrange, é necessário determinar a energia cinética e a energia potencial total do conjunto formado pelo pêndulo e

pelas rodas, conforme FARIA (2015). A energia cinética total é dada pela soma das energias cinéticas lineares e rotacional do corpo do sistema e das rodas, como é mostrado por (4.9) e (4.10).

$$T = T_r^R + T_p^R + T_r^T + T_p^T \quad (4.9)$$

$$T = \frac{1}{2}J_r\dot{\theta}^2 + \frac{1}{2}J_p\dot{\theta}^2 + \frac{1}{2}M_r v_r^2 + \frac{1}{2}m_p v_p^2 \quad (4.10)$$

As velocidades  $v_r$  e  $v_p$  representam, respectivamente as componentes de velocidades das rodas e do corpo do pêndulo. Os módulos de  $v_r$  e  $v_p$  são mostrados a seguir pelas Equações (4.11) e (4.12).

$$\begin{aligned} v_r &= \dot{x} = r\dot{\theta} \\ |v_r|^2 &= \dot{x}^2 = r^2\dot{\theta}^2 \end{aligned} \quad (4.11)$$

$$\begin{aligned} v_p &= \frac{d}{dt}(x + z_{cm} \sin \theta) + \frac{d}{dt}(z_{cm} \cos \theta) \\ |v_p|^2 &= \left[ \frac{d}{dt}(x + z_{cm} \sin \theta) \right]^2 + \left[ \frac{d}{dt}(z_{cm} \cos \theta) \right]^2 \\ |v_p|^2 &= \dot{x}^2 + 2z_{cm}\dot{x}\dot{\theta} \cos \theta + z_{cm}^2\dot{\theta}^2 \end{aligned} \quad (4.12)$$

Substituindo-se os valores encontrados nas equações (4.11) e (4.12) em (4.10), determina-se a equação da energia cinética em função das posições e velocidades angulares, assim como mostra a **Equação (4.13)**.

$$T = \frac{1}{2}J_{eq}\dot{\theta}^2 + \frac{1}{2}m\dot{x}^2 + M_p z_{cm} \dot{x} \dot{\theta} \cos \theta + \frac{1}{2}M_p z_{cm}^2 \dot{\theta}^2 \quad (4.13)$$

Sendo que  $J_{eq}$  é equivalente a soma dos momentos de inércia do pêndulo  $J_p$  e os das rodas  $J_w$ . No sistema temos uma única energia potencia  $V$  que vem da gravidade e afeta somente o corpo do pêndulo. Essa energia é definida pela Equação (4.14).

$$V = M_p g z_{cm} \cos \theta \quad (4.14)$$

A equação de Lagrange de movimento para o PIDR é dada por (4.15). Será trabalhado apenas com a equação de movimento para a variável de interesse  $\theta$ , uma vez que somente esta será possível de ser medida e controlada.

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = \tau_m \quad (4.15)$$

Nas equação descrita, o valor de  $L$  é dado pela Equação (4.16), sendo  $T$  e  $V$  as energia cinética e potencial totais do pêndulo, respectivamente. O torque total aplicado ao pêndulo é representado pelo conjugado de saída  $\tau_m$  do motor.

$$L = T - V \quad (4.16)$$

Realizando a subtração da energia cinética e da energia potencial do sistema que foram determinadas, como mostrado em (4.16), encontra-se a equação para  $L$ . O resultado desta operação é dado por (4.17).

$$L = \frac{1}{2}J_{eq}\dot{\theta}^2 + \frac{1}{2}m\dot{x}^2 + M_p z_{cm}\dot{x}\dot{\theta} \cos \theta + \frac{1}{2}M_p z_{cm}\dot{\theta}^2 - mgz_{cm} \cos \theta \quad (4.17)$$

Como visto na Equação (4.11), há uma correspondência entre  $\dot{x}$  e  $\dot{\theta}$ . Sendo assim, o resultado obtido em (4.17) que está em termos de duas variáveis,  $x$  e  $\theta$ , será modificado para apenas a variável de interesse,  $\theta$ . O resultado é mostrado por (4.18).

$$L = \frac{1}{2}J_{eq}\dot{\theta}^2 + \frac{1}{2}mr^2\dot{\theta}^2 + M_p z_{cm}r\dot{\theta}^2 \cos \theta + \frac{1}{2}M_p z_{cm}\dot{\theta}^2 - mgz_{cm} \cos \theta \quad (4.18)$$

Agora é possível voltar a equação de Lagrange de movimento para o pêndulo sobre duas rodas (4.15), utilizando do resultado conseguido na Equação (4.18). Assim, por meio da Equação (4.19), obtém-se a equação não linear do sistema físico por meio de uma modelagem baseada em Euler-Lagrange.

$$(J_{eq} + mr^2 + 2M_p z_{cm}r \cos \theta + M_p z_{cm})\ddot{\theta} - (M_p z_{cm}r \sin \theta)\dot{\theta}^2 - mgz_{cm} \sin \theta = \tau_m \quad (4.19)$$

O torque de saída do motor  $\tau_m$  em termos da tensão elétrica é definido pela Equação (4.4), na qual a variável  $U$  corresponde à tensão elétrica de entrada controlada. A Equação (4.20) define a equação final do torque do motor para o modelo deste trabalho, na qual considera-se que a velocidade angular do eixo do motor  $\dot{\varphi}$  corresponde a velocidade angular do corpo do pêndulo  $\dot{\theta}$  como forma de aproximação, já que esta é a única variável a ser medida.

$$\tau_m = \frac{k_m}{R}U - \frac{k_m k_e}{R}\dot{\theta} \quad (4.20)$$

Por fim, para a realização da linearização por matrizes Jacobianas conforme descrito na seção (2.3), considera-se que  $\dot{x}_2 = \ddot{\theta}$ ,  $\dot{x}_1 = \dot{\theta}$ ,  $x_1 = \theta$  e  $u = U$ . Dessa forma, temos as seguintes equação de movimento não linear que descreve o sistema físico

$$f_i(x, u) = \begin{cases} \dot{x}_1 = \dot{\theta} \\ \dot{x}_2 = \ddot{\theta} = \frac{(K_m u - K_e x_2) + (M_p z_{cm}r \sin x_1)x_2^2 + mgz_{cm} \sin x_1}{J_{eq} + mr^2 + M_p z_{cm}(1 + 2r \cos x_1)} \end{cases} \quad (4.21)$$

sendo  $K_m = \frac{k_m}{R}$  e  $K_e = \frac{k_m k_e}{R}$ .

Para linearizar as equações encontradas, é preciso escolher um ponto de equilíbrio desejado. Sendo assim, escolheu-se o seguinte valor:  $x_1^{eq} = x_2^{eq} = u^{eq} = 0$ . O resultado da linearização conforme visto em (2.7) é mostrado na Equação (4.22)

$$\begin{aligned}
 \frac{\partial f_1(x^{eq}, u^{eq})}{\partial x_1} &= 0 & \frac{\partial f_1(x^{eq}, u^{eq})}{\partial x_2} &= 1 \\
 \frac{\partial f_2(x^{eq}, u^{eq})}{\partial x_1} &= \frac{mgz_{cm}}{\rho} & \frac{\partial f_2(x^{eq}, u^{eq})}{\partial x_2} &= \frac{-K_e}{\rho} \\
 \frac{\partial f_1(x^{eq}, u^{eq})}{\partial u} &= 0 & \frac{\partial f_2(x^{eq}, u^{eq})}{\partial u} &= \frac{K_m}{\rho}
 \end{aligned} \tag{4.22}$$

sendo que  $\rho = J_{eq} + mr^2 + M_p z_{cm}(1 + 2r)$ .

Feito isso, é possível definir as matrizes  $A$ ,  $B$ ,  $C$  e  $D$  do modelo linearizado. Essas matrizes são mostradas logo abaixo.

$$\begin{aligned}
 A &= \begin{bmatrix} 0 & 1 \\ \frac{mgz_{cm}}{J_{eq} + mr^2 + M_p z_{cm}(1 + 2r)} & \frac{-k_m k_e}{R(J_{eq} + mr^2 + M_p z_{cm}(1 + 2r))} \end{bmatrix} \\
 B &= \begin{bmatrix} 0 \\ \frac{k_m}{R(J_{eq} + mr^2 + M_p z_{cm}(1 + 2r))} \end{bmatrix} \\
 C &= [1 \quad 0] \\
 D &= [0]
 \end{aligned}$$

### 4.3 Projeto do Controlador de Estabilização

O controlador que será projetado nesta seção tem a função de estabilizar e manter o pêndulo na posição de equilíbrio instável e não natural, ou seja, esse controlador busca manter o corpo do pêndulo em  $\theta = 0^\circ$ . Assim como explicado e mostrado em capítulos anteriores, o controlador atuará apenas em uma estreita faixa, a mais próxima do ponto desejada, uma vez que a linearização ocorreu em torno do ponto de equilíbrio.

Na literatura, há uma variada coleção de métodos que podem ser implementados com intuito de definir parâmetros de estratégias de controle. O método “*Linear Quadratic Regulator*”(LQR) em conjunto com o “*Linear Quadratic Estimator*”(LQE) formam o “*Linear Quadratic Gaussian*”(LQG) que é uma técnica robusta e apropriada para encontrar os parâmetros de equilíbrio **do sistema**. Dado que as equações de movimento do sistema podem ser descritas na forma de espaço de estados, conforme a Equação (2.10).

Na seção (2.6) é explicado como funciona e o que é necessário realizar para se obter os parâmetros do controlador LQG. O algoritmo do LQR calcula uma ação de controle  $u$  para minimizar a função custo  $J$  como mostrado pela Equação (4.23)

$$J = \int_0^\infty x(t)^T Q x(t) + u(t)^T R u(t) dt \tag{4.23}$$



sendo que as matrizes  $Q$  e  $R$  definem o peso do vetor de estado e o peso sobre a ação de controle. Percebe-se que se  $Q$  for aumentado, o controlador deve trabalhar mais para minimizar o custo da função e consequentemente, o ganho do controle resultante será maior. O vetor de estado do nosso sistema é definido como sendo:

$$x = [\theta \quad \dot{\theta}]^T$$

Por sua vez, a matriz  $R$  será um escalar uma vez que estamos trabalhando em um sistema SISO. Sendo assim, o único sinal de entrada que a planta terá será a tensão de alimentação do motor  $U$ .

### 4.3.1 Controlador LQR

Na seção (2.6) foram descritos os procedimentos necessários para se determinar os ganhos de realimentação de estados. Para a realização do cálculo deste ganho utilizou-se o pacote **MathWorks** do *software* principal utilizado neste trabalho, o *MATLAB*.

Com os valores dos ganhos definidos pelo projeto, os pólos desejados para o sistema realimentado são obtidos. De forma a mostrar e provar que o sistema é realmente instável, mostra-se primeiramente os pólos de malha aberta. Assim, fica claro e simples de verificar a instabilidade do sistema em espaço de estados, obtido com a linearização exibida na seção (4.2). A Figura (4.3) mostra o mapa de pólos e zeros para o sistema em malha aberta.

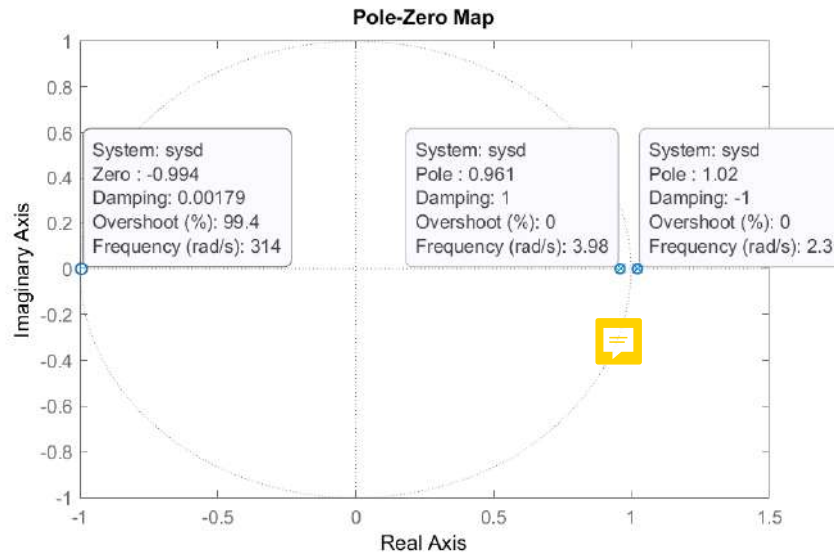


Figura 4.3: Círculo unitário de pólos e zeros do modelo discretizado.

Ao analisar a Figura (4.3), identifica-se um zero em  $-0.994$  e os autovalores de malha aberta do sistema discreto como

$$\lambda_{MA} = [0.9610 \quad 1.0232]$$

sendo que pólo que mais a direita e fora do círculo unitário, confirma a instabilidade do sistema, de acordo com a literatura. Sendo assim, o controlador deverá levar este pólo para dentro do círculo unitário, ou seja, para dentro da região de estabilidade.

Utilizando-se da função *dlqr* do *MATLAB*, o ganho  $K_{lqr}$  do controlador é encontrado. A utilização deste comando depende das matrizes do espaço de estados do sistema discretizado pelo período de amostragem  $T_s = 0.002s$  escolhido -  $A$ ,  $B$ ,  $C$  e  $D$  - e dos pesos das matrizes da função de custo -  $R$  e  $Q$ . A obtenção dos pólos de malha fechada desejados é obtido ao calcular o ganho do controlador com a função *dlqr* ou fazendo  $(A - BK_{lqr})$ . Em relação aos pesos das matrizes da função de custo, definiu-se pesos unitários de modo a conseguir um controlador ótimo entre o custo sobre a ação e a precisão de controle. Abaixo em (4.24) e (4.3.1), é o mostrado o vetor de ganho  $K_{lqr}$  do controlador e o vetor dos pólos de malha fechada  $\lambda_{MF}$ .

$$K_{lqr} = [k_\theta \quad k_{\dot{\theta}}] = [3.2535 \quad 1.1798] \quad (4.24)$$

$$\lambda_{MF} = [0.9855 \quad 0.9272] \quad (4.25)$$

Com intuito de visualizar de forma gráfica o comportamento do modelo não linear com o controlador projetado, elaborou-se uma simulação na plataforma *Simulink* do *MATLAB*. Sabendo-se que a linearização ocorreu em uma região em torno do equilíbrio que é de  $\theta = 0^\circ$ , setou como condições iniciais para o modelo  $\theta_0 = 5^\circ$  e  $\dot{\theta}_0 = 0^\circ/s$ . A Figura (4.4) mostra o conjunto de blocos necessários para a implementação do regulador no *Simulink*.

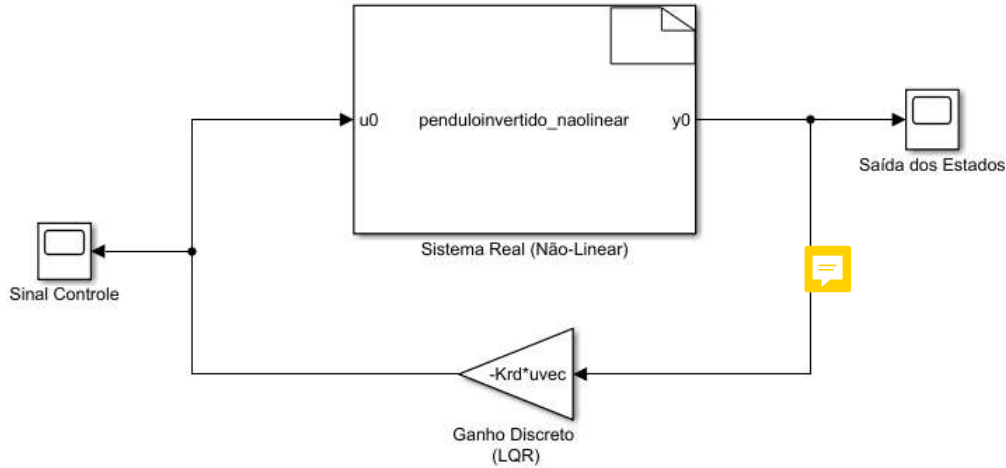


Figura 4.4: Modelo não linear realimentado pelo controlador *LQR*.

As seguintes Figuras (4.5) e (4.6) são as respostas dos estados e o sinal de controle respectivamente para o sistema realimentado apenas com o regulador e sem um observador de estados.

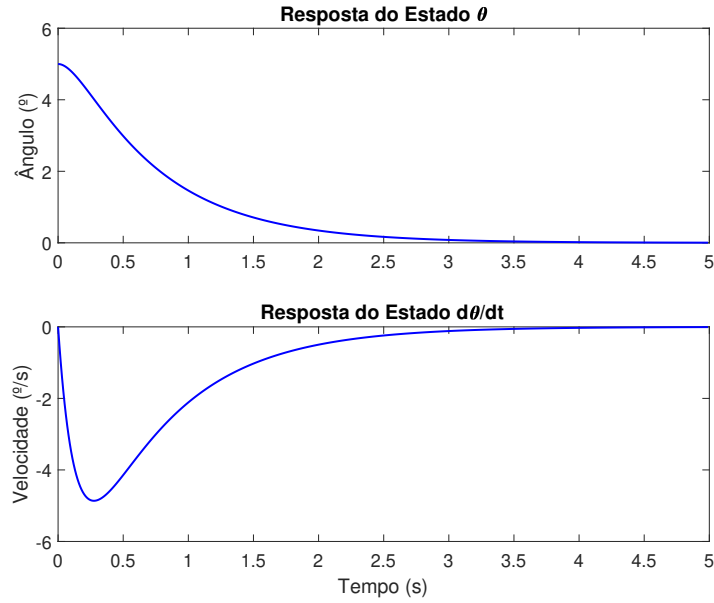


Figura 4.5: Respostas dos estados do sistema para a condição inicial estabelecida.

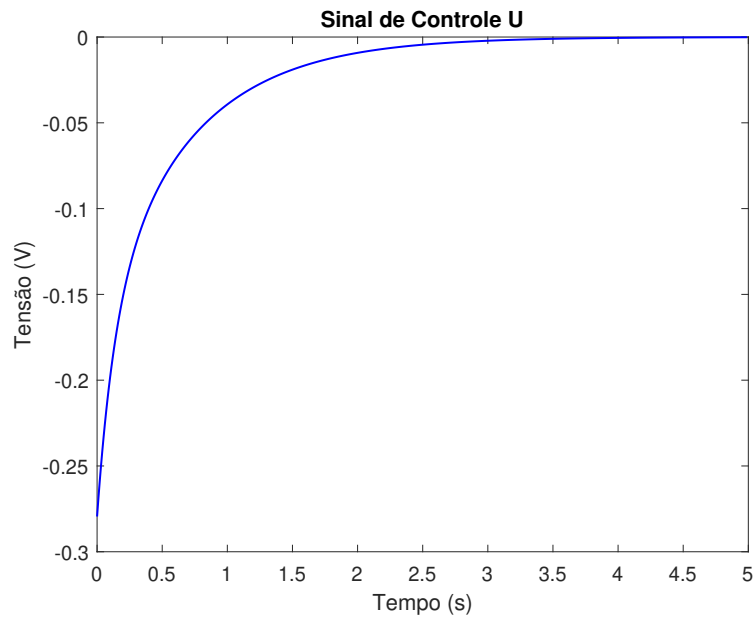


Figura 4.6: Sinal de controle do sistema para a condição inicial estabelecida.

### 4.3.2 Estimador de Estados (Filtro de Kalman)

A subseção (4.3.1) deixa claro que com apenas o ganho do controlador  $LQR$ , temos um resultado satisfatório. Contudo, o intuito do deste trabalho é realizar o projeto de um estimador de estados para que o mesmo consiga melhorar significativamente o sinal de saída dos mesmos. Quando realizamos a junção tanto do  $LQR$  quanto de um estimador de estados, obtemos o  $LQG$  que tende a ser mais robusto e menos sensível a ruídos de medição e distúrbios. A simulação do sistema que agora contém um observador de estados, foi realizada sob as mesmas condições da simulação anterior. Sendo assim, as condições iniciais inseridas

foram  $\theta_0 = 5^\circ$  e  $\dot{\theta}_0 = 0^\circ/s$ . O *design* do sistema com o acréscimo do *LQE* é visto nas Figuras (4.7) e (4.8).

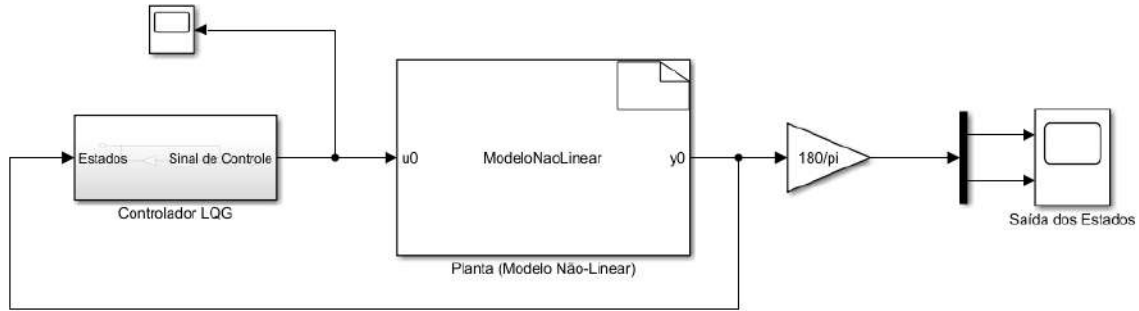


Figura 4.7: Modelo não linear realimentado pelo controlador *LQG*.

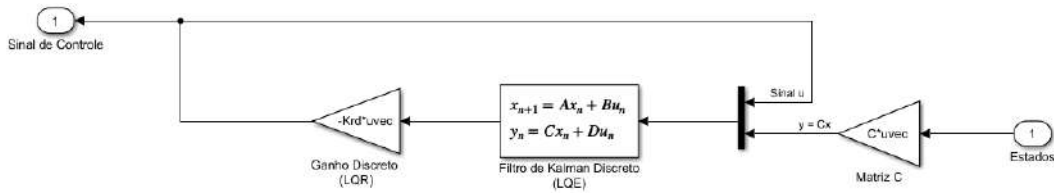


Figura 4.8: Detalhamento do controlador *LQG*.

Com essa nova topografia, realizou-se o mesmo teste que o anteriormente. As condições iniciais para a planta são as mesmas como já mencionado, o que difere agora são as condições iniciais do observador que foram setadas como sendo  $\hat{\theta} = \hat{\dot{\theta}} = 0$ . Abaixo, nas Figuras (4.9) e (4.10) é visto a resposta dos sinais dos estados e do sinal de controle respectivamente.

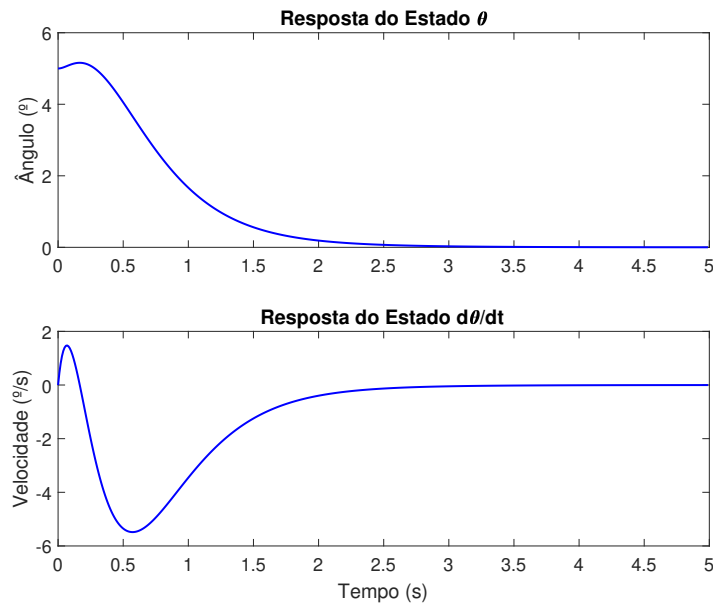


Figura 4.9: Respostas dos estados do sistema para a condição inicial estabelecida.

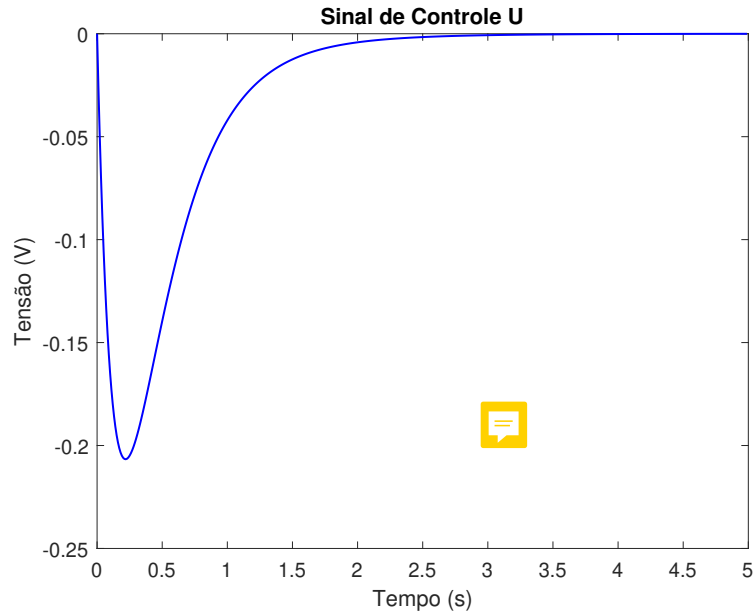


Figura 4.10: Sinal de controle do sistema para a condição inicial estabelecida.

### Inserção de Distúrbio e Ruído de Medição ao Sistema

Como visto, em condições ideais e não reais já que todo sinal de medição haverá algum tipo de ruído, distúrbio e outras perturbações que está sendo até simplificado já no modelo, percebe-se que o controlador atua muito bem e sem dificuldade para estabilizar a planta. Dessa maneira, para testar o modelo de forma mais real e obter um resultado mais fidedigno do que realmente irá acontecer ao embarcar o controlador na planta física, resolveu-se incorporar ao sistema um sinal de ruído branco na resposta do sensor. Esse sinal, como visto na Figura (4.11), possui um *range*  $\pm 1^\circ$ , ou seja, quer dizer o sensor pode estar errando tanto  $1^\circ$  positivo quanto  $1^\circ$  negativo.

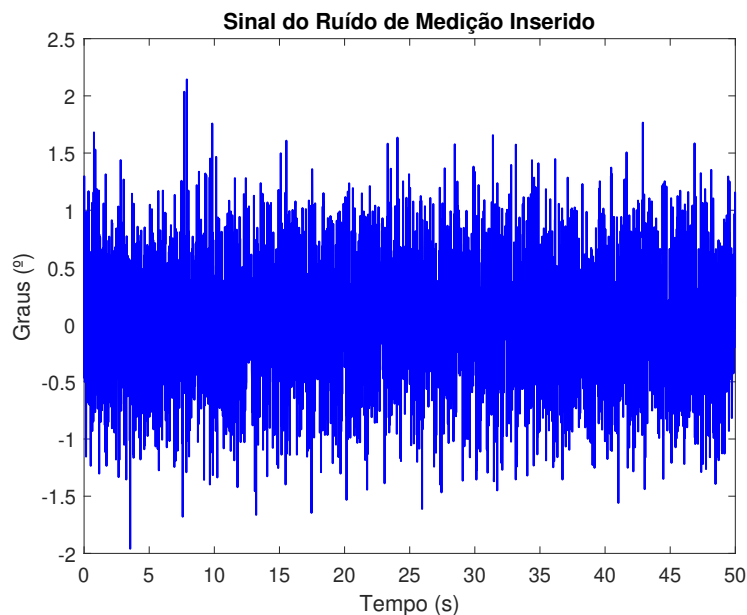


Figura 4.11: Sinal do ruído branco de medição.

Além do ruído inserido, também decidiu-se por aplicar um distúrbio na simulação com intuito de visualizar o comportamento do controlador perante a uma adversidade, uma vez que após um certo tempo de início o sistema tende a ficar estável. Sendo assim, de forma proposital aplicou um sinal quadrático de duração de 0.5s, iniciando em  $t = 20s$  e com amplitude de  $3^\circ$ . O *design* do sistema com essas novas dificuldades impostas ao controlador é visto na Figura (4.12).

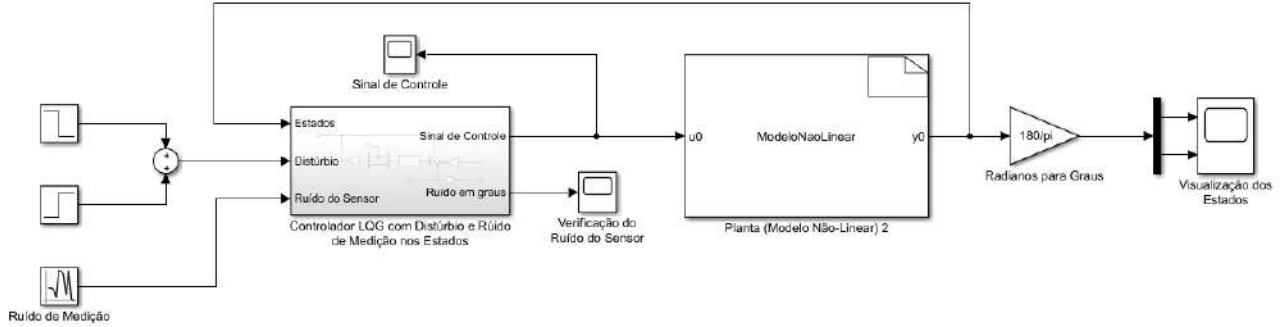


Figura 4.12: Modelo não linear realimentado pelo controlador  $LQG$  com inserção de distúrbio e ruídos.

A Figura (4.13) mostra o detalhamento do controlador  $LQG$  com o distúrbio e o ruído aplicado. Como é visto, o sinal do sinal do distúrbio é somado juntamente ao estado  $\hat{\theta}$ , antes de passar pelo regulador. Por sua vez, o sinal de ruído é somado ao sinal de medição realimentado.

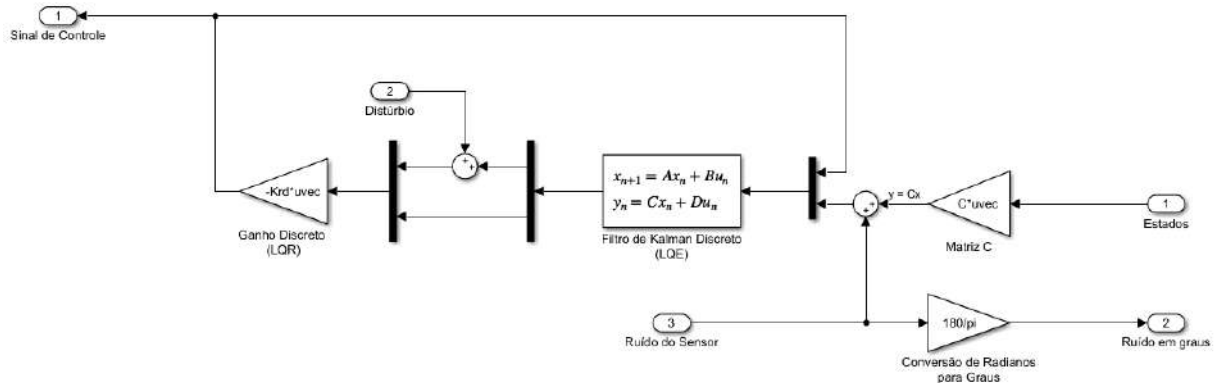


Figura 4.13: Detalhamento do controlador  $LQG$  com a inserção de distúrbio e ruído de medição.

Contudo, para aproximar mais da realidade é inserido ao sistema um sinal quadrático de duração de 0.5s, iniciando em  $t = 20s$  e com amplitude de  $3^\circ$ . O resultado dos sinais dos estados com essa configuração é visto na Figura (4.14). Mais uma vez, mesmo que com variações torno de  $0^\circ$ , percebe-se que em ambiente simulado o controlador atuou de forma satisfatória.



Figura 4.14: Respostas dos estados do sistema para a condição inicial estabelecida.

Para entender o quanto de sinal de controle está sendo aplicado aos motores, observa-se a Figura (4.15).

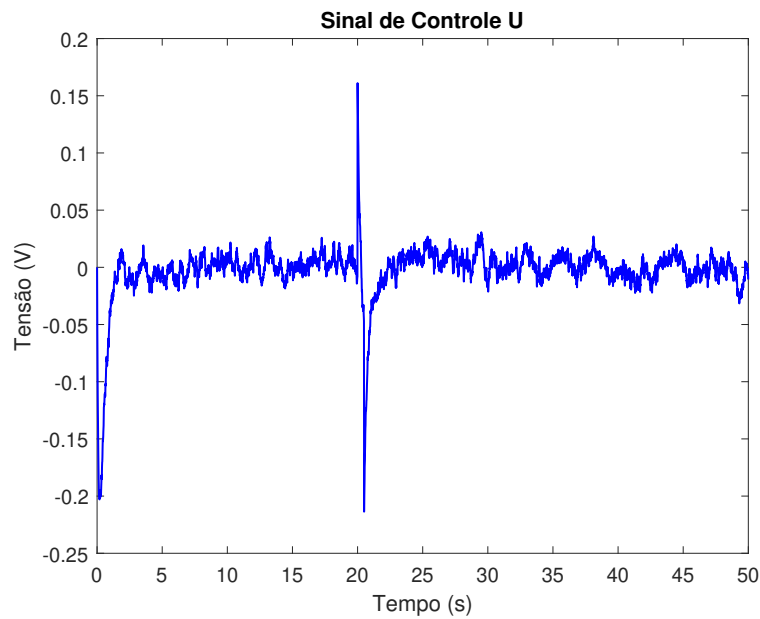


Figura 4.15: Sinal de controle do sistema para a condição inicial estabelecida.



## Resultados Práticos

Este capítulo abordará os resultados práticos obtidos durante o desenvolvimento deste trabalho.

### 5.1 Montagem da Planta

Como relatado na Seção 3.1, ao se desenhar as placas da estrutura em *software* CAD levou-se em consideração os furos para os componentes e passagem das cablagens. Em relação ao material, as mesmas são de ACM preto e foram cortadas por uma fresadora para um melhor acabamento e precisão. A Figura (5.1) mostra duas vistas da montagem, frontal e lateral. Além da estrutura mecânica, é visto também o conjunto de motores e rodas e os componentes eletrônicos.

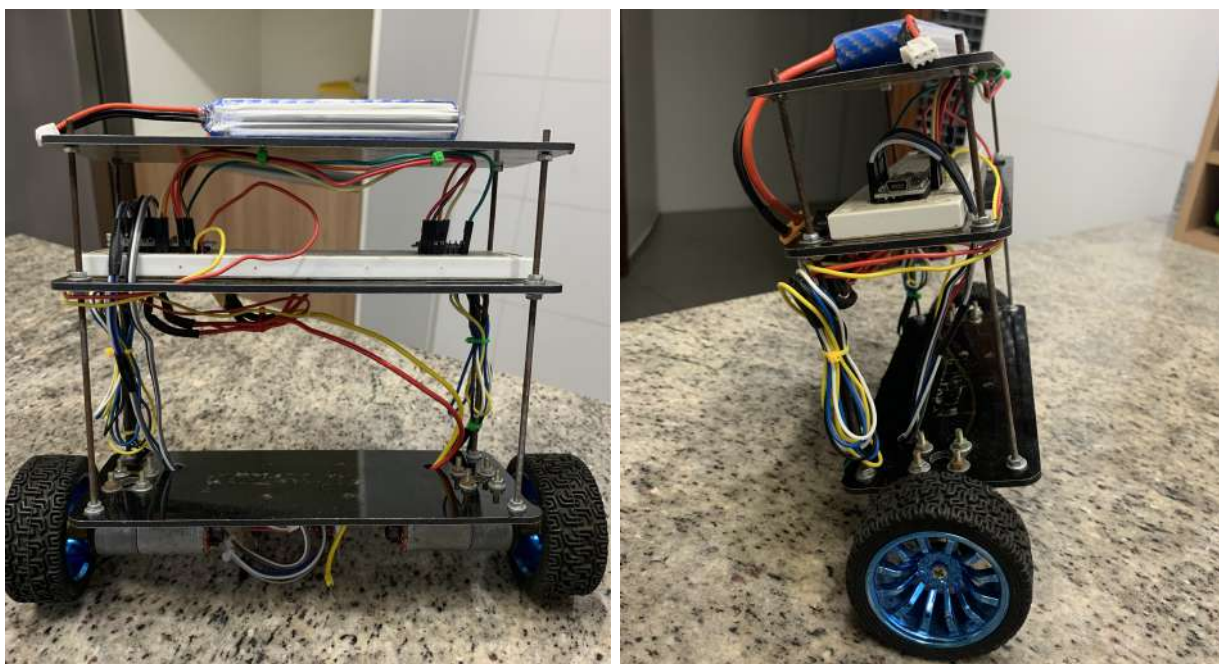


Figura 5.1: Vistas frontal e lateral da montagem da planta física.



## 5.2 Validação do Sensor e Motores

Com a finalização da construção da planta física, projeto do controlador e realização da simulação conforme mostrado no Capítulo 4, foram realizados testes práticos para verificação do funcionamento do sistema montado. Foi desenvolvido um código utilizando a linguagem C, uma vez que a placa controladora será o microcontrolador Arduino. Essa placa tem como principal responsabilidade o processamento do sinal do sensor de posição angular para obtenção do sinal de controle utilizando dos ganhos do controlador LQG e consequentemente, gerando um sinal PWM para a placa L298N acionar os motores de corrente contínua. O código desenvolvido está em meu github, mas a ideia de funcionamento é, basicamente:

- se a estrutura tender a ir para frente, os motores receberão um valor de PWM e um comando para ir para frente. O valor do PWM é dependente de uma multiplicação matricial do valor do sinal do sensor filtrado e do ganho do regulador linear quadrático.
- caso a estrutura for para trás, o comando que a placa L298N receberá será contrário aquele recebido quando a estrutura for para frente. A lógica cálculo do valor do PWM se mantém a mesma.

Após desenvolvimento do código, esse foi carregado para o Arduino para obtenção dos sinais do sensor e acionamento dos motores para testes de movimentação. O circuito eletrônico montado tanto para testes quanto para controle da posição do pêndulo está esquematizado na Figura (3.9). Os resultados são visto abaixo.

### 5.2.1 Validação do Sensor

Após a montagem do circuito eletrônico, como mostrado acima na Figura (3.9), desenvolvimento do código Arduino e carregamento do mesmo na placa, foi possível realizar as validações. Primeiramente, focou-se em validar os sinais do sensor. A ideia é ver a diferença entre o sinal real do sensor e o filtrado por Kalman, afim de corroborar com o que a literatura afirma a respeito deste filtro, de que o mesmo traz uma maior robustez e é menos sensível a ruídos e distúrbios.

Entretanto, para obter os dados do sensor só foi possível através da serial da porta do computador, uma vez que neste trabalho não se utiliza de nenhum componente eletrônico de transmissão. Sendo assim, desenvolveu-se um código no Processing, o qual fez a leitura dos dados em forma de *string* da serial e os salvaram em um arquivo “.txt”. Com esse arquivo, desenvolveu novamente um novo código, só que dessa vez foi utilizando a linguagem Python. Dessa maneira, realizou-se os tratamentos necessários e gerou-se as figuras com os resultados obtidos.

Com intuito de validar bem os sinais e o filtro desenvolvido, gerou-se mais de um arquivo com dados. De forma macro, a Figura (5.2), mostra o resultado guardado de dois arquivos.

O interessante desse resultado visto, é observar que o sinal real (*pitch*) é bastante sensível a ruído e o sinal filtrado, por sua vez, é bem suave mas sempre segue o sinal real.

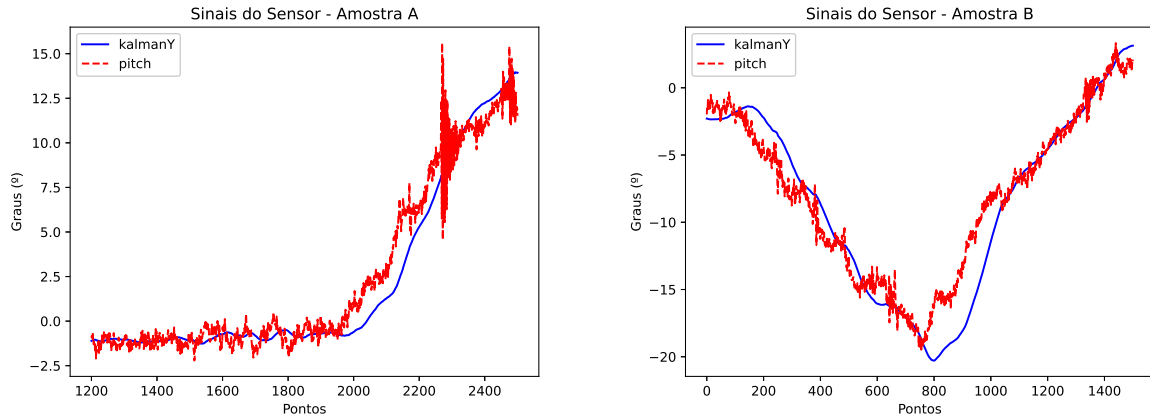


Figura 5.2: Comportamento de forma macro do sinal do sensor filtrado, curva azul e do sinal real, curva vermelha.

Para uma melhor visualização da curva, plotou-se apenas uma pequena fatia dos dados da amostra B, como mostra a Figura (5.3). Aqui, verifica-se e comprova o que foi dito, que a sensibilidade a ruído do sinal real é bem alta e que o sinal filtrado segue a curva real de forma bem suave.

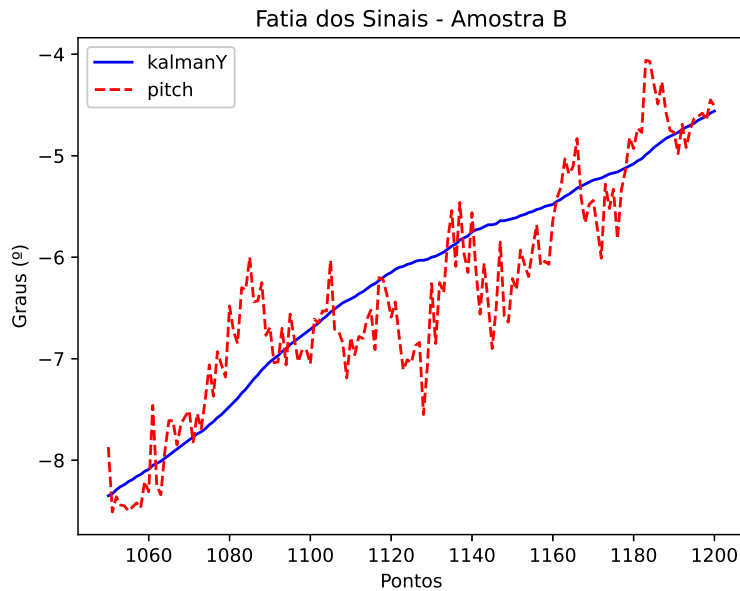


Figura 5.3: Comportamento de forma micro dos dois sinais do sensor.

### 5.2.2 Validação dos Motores

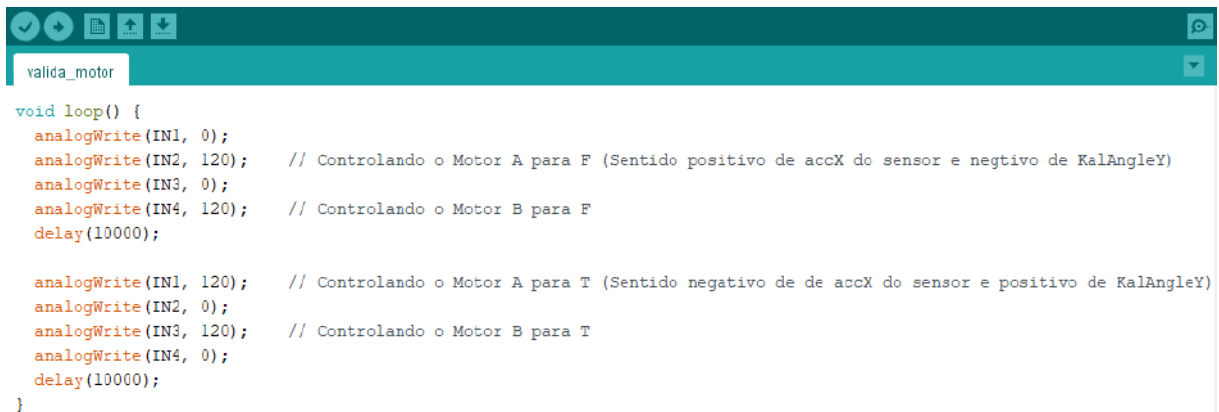
Os motores, por sua vez, irão permitir que aja movimentação da estrutura hora para frente hora para trás, dependendo do sinal enviado do sensor para a placa controladora, Arduino. Inicialmente, escolheu-se 4 pinos do Arduino que tivessem a característica de

serem PWM para controlar o sentido de giro dos motores e os conectaram nos pinos da ponte H L298N, conforme mostrado na Tabela (5.1).

Tabela 5.1: Pinagem entre Arduino e Ponte H L298N.

Motores	Arduino	L298N
Motor A	D5	IN1
Motor A	D6	IN2
Motor B	D9	IN3
Motor B	D11	IN4

Da forma que os conjuntos eletrônicos foram montados e o sensor posicionado, padronizou o sentido para frente sendo o sentido negativo do acelerômetro X (accX) e para trás o sentido positivo. Fazendo dessa maneira, observou-se que quando a estrutura está inclinada no sentido positivo de accX, o ângulo Y (variável desejada a ser controlada) é negativo, sendo assim precisa que o carrinho mova para frente e vice-versa. A Figura (5.4) mostra o código desenvolvido que identificou e validou o sentido dos motores.



```

void loop() {
  analogWrite(IN1, 0);
  analogWrite(IN2, 120); // Controlando o Motor A para F (Sentido positivo de accX do sensor e negativo de KalAngleY)
  analogWrite(IN3, 0);
  analogWrite(IN4, 120); // Controlando o Motor B para F
  delay(10000);

  analogWrite(IN1, 120); // Controlando o Motor A para T (Sentido negativo de de accX do sensor e positivo de KalAngleY)
  analogWrite(IN2, 0);
  analogWrite(IN3, 120); // Controlando o Motor B para T
  analogWrite(IN4, 0);
  delay(10000);
}

```

Figura 5.4: Trecho do código para validar os sentidos de giro dos motores.

Caso queira visualizar o funcionando dos motores através quando esse código é aplicado, basta clicar no link.

## 5.3 Aplicando o Controlador na Planta

Para realizar a tentativa do controle da planta, foi desenvolvido um código em C, utilizando o Arduino IDE, que necessita da realimentação dos estados, ou seja, do sinal de saída da planta que o sensor nos fornece para que seja realizado um cálculo de quanto de PWM é preciso aplicar aos motores. Esse código pode ser visto em github clicando aqui.

Inserindo o ganho  $K_{lqr}$ , visto na Equação (4.24), ao código desenvolvido observa-se a saída da planta, conforme mostrado na Figura (5.5).



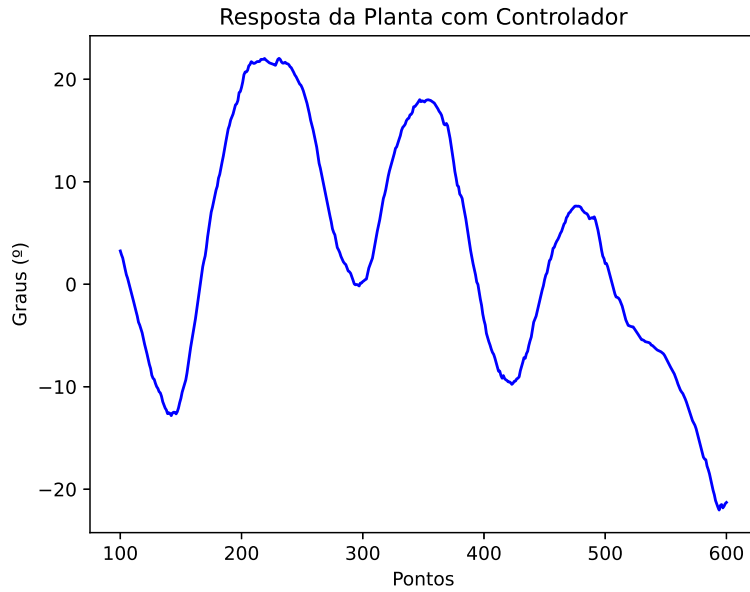


Figura 5.5: Resultado da planta ao aplicar o controlador projetado.

Como mostrado, o sistema fica fora da faixa de linearidade a quase todo momento, algo em torno de  $\pm 10^\circ$ . Sendo assim, fez com que a planta não pudesse ser controlada e sua posição não ficou estável em  $0^\circ$  ou o mais próximo disso.

O sinal de controle também pode ser visto na Figura (5.6). Observa-se que em vários momentos ele atinge os picos de máximos e mínimos permitido. A faixa permitida do PWM é de  $\pm 200$ , já que a tensão de saída da bateria que está sendo utilizada é de  $7.4V$  e o valor máximo permitido de alimentação para os motores é de  $6V$ .

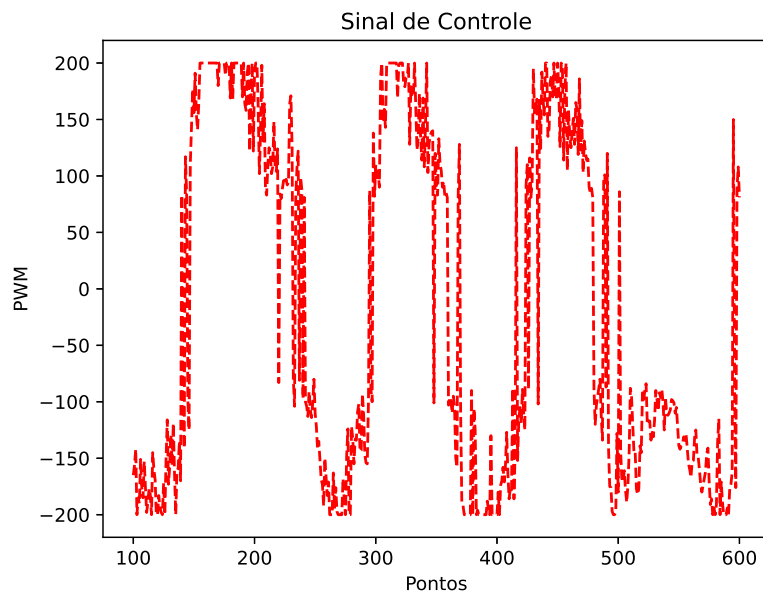


Figura 5.6: Sinal de controle ao aplicar o controlador projetado.

## Considerações Finais

Neste capítulo são apresentadas as considerações decorrentes dos resultados obtidos no desenvolvimento deste trabalho e a proposta de continuidade para o estudo.

### 6.1 Conclusões

O objetivo deste trabalho foi desenvolver uma planta física de controle como parte da disciplina de trabalho de conclusão de curso, utilizando técnica de controle ótimo para sistema não linear. Além disso, utilizou-se de vários outros conceitos abordados ao longo do curso, tais como desenho e cálculos mecânicos, eletrônica e circuitos elétricos e programação para computadores.

Com relação ao tema abordado, conclui-se que é de grande importância na academia e que possui uma variedade de pesquisas já realizadas em relação a este sistema. A técnica de modelagem utilizada é bastante usual na literatura.

Em relação as técnicas de controle ótimo, há vários estudos sobre, o que assegurou uma maior certeza na escolha do controlador implementado. O projeto do controlador LQG por meio de ajuda de *software*, como o MATLAB, mostrou ser fácil de se implementar em simulação e na prática. Em simulação, os resultados obtidos com relação ao sistema não linear, foi bastante eficaz mesmo lidando com distúrbios e ruídos aplicados.

Ao fim da montagem da estrutura mecânica e do circuito eletrônico, realizou-se a validação do sensor e dos motores. Observou-se a sensibilidade para ruído do do sinal real vindo do sensor e a robustez do sinal filtrado. Em relação aos motores, obteve-se os máximos e mínimos de PWM e sentido de giro de cada motor. Ao aplicar o controlador, notou-se a diferença de velocidade em cada roda, fazendo a estrutura girar. Esse acontecimento era esperado uma vez que não há controle dos motores, apenas da posição angular da estrutura. O intuito ao aplicar o controlador era manter a planta estável na vertical, contudo, conforme mostrado no capítulo anterior, os resultados não foram satisfatórios e consequentemente, esse objetivo não foi alcançado de forma eficaz.

Com o presente trabalho foi possível desenvolver habilidades extras da área de controle

que não são abordadas na grade curricular do curso. A experiência adquirida nas disciplinas teóricas bem como as de laboratórios foram de suma importância para atingir os resultados alcançados. Contudo, como já citado, este trabalho é multidisciplinar e que necessitou de utilizar de vários fundamentos adquiridos ao longo do curso.

## 6.2 Propostas de Trabalhos Futuros

A partir da experiência adquirida durante o desenvolvimento deste trabalho, foram elencados os seguintes tópicos como possíveis continuações do estudo:

- Identificação de parâmetros aqui descartados, afim de simplificar o modelo, como a fricção e averiguar a necessidade de adição;
- Identificação mais robusta dos parâmetros mecânicos e elétricos dos motores utilizados;
- Implementação do uso dos *encoders* dos motores, o que acarretaria em uma adição de dois novos estados na modelagem, posição e velocidade linear;
- Um outra abordagem é da utilização de motores de passo ao invés de motores de corrente contínua, uma vez que os motores passo tem um ótimo controle de posição e são muito precisos;
- Implementação de novas técnicas de controle para efeito de comparação com a abordada neste trabalho;
- Utilização de técnicas de controle MIMO (múltiplas entradas e múltiplas saídas), como a realização de um controlador em cascata para o controle da posição linear e da vertical.

# Referências

- ALVES, R. G. *Controle de um pêndulo invertido utilizando técnica de linearização por realimentação*. Ouro Preto, MG, 2018.
- BRUNTON, S. L.; DURIEZ, T.; NOACK, B. R. *Machine Learning Control - Taming Non-linear Dynamics and Turbulence*. [S.l.]: Springer International Publishing, 2017.
- CHEN, C.-T. *Linear System Theory and Design*. 3. ed..ed. New York: Oxford University Press, 1999. 334p.
- ENGİN, M. Embedded LQR controller design for self-balancing robot. In: MEDITERRANEAN CONFERENCE ON EMBEDDED COMPUTING (MECO), 2018., Budva, Montenegro. *Anais. . . IEEE*, 2018. p.1–4.
- FARIA, L. M. *Modelagem e Controle do Pêndulo Invertido Rotacional ou Pêndulo de Furuta*. Belo Horizonte, MG, 2015.
- FERNANDES, P. B. *Controladores Robustos do Tipo LQG/LTR de Ordem Reduzida para Sistemas MIMO com Saídas Independentes de seus Modos Não Dominantes*. 2014. — Universidade Federal do Pará, Belém, PA.
- GROOVER, M. P. *Automação industrial e sistemas de manufatura*. 3. ed..ed. São Paulo: Pearson Prentice Hall: Prentice-Hall, Inc, 2013. 581p.
- HESPAÑA, J. P. *Linear Systems Theory*. 1. ed..ed. Princeton, New Jersey: Princeton University Press, 2009. 280p.
- INVENSENSE. *MPU-6000 and MPU-6050 Product Specification*. Acesso: 2 de Maio de 2019, [https://store.invensense.com/datasheets/invensense/MPU-6050\\_DataSheet\\_V3%204.pdf](https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf).
- JUANG, H.; LURRR, K. Design and control of a two-wheel self-balancing robot using the arduino microcontroller board. In: IEEE INTERNATIONAL CONFERENCE ON CONTROL AND AUTOMATION (ICCA), 2013. *Anais. . . [S.l.: s.n.]*, 2013. p.634–639.

- JUNFENG, W.; WANYING, Z. Research on Control Method of Two-wheeled Self-balancing Robot. In: FOURTH INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTATION TECHNOLOGY AND AUTOMATION, 2011. *Anais...* [S.l.: s.n.], 2011. v.1, p.476–479.
- JUNG, C. R.; OSÓRIO, F. S.; KELBER, C. R.; HEINEN, F. J. Computação Embarcada: projeto e implementação de veículos autônomos inteligentes. , [S.l.], 2005.
- KIM, H. S.; SOHN, J. W.; JEON, J.; CHOI, S.-B. Reduction of the Radiating Sound of a Submerged Finite Cylindrical Shell Structure by Active Vibration Control. *Sensors (Basel, Switzerland)*, [S.l.], p.2132–2147, 2013.
- MATOS, E. J. de. *Projeto de Controladores Robustos do tipo LQG/LTR para Sistema Multivariável Instável com uso de Pré-Compensador Dinâmico*. 2008. — Universidade Federal do Pará, Belém, PA.
- MELO, J. V. *Identificação e Controle de um Veículo Segway para fins Educacionais*. Porto Alegre, RS: SBAI, 2017. 1-6p.
- NGUYEN GIA MINH THAO, D. H. N.; PHUCAND, N. H. A PID backstepping controller for two-wheeled self-balancing robot. In: INTERNATIONAL FORUM ON STRATEGIC TECHNOLOGY 2010. *Anais...* [S.l.: s.n.], 2010. p.76–81.
- OGATA, K. *Engenharia de Controle Moderno*. 5. ed..ed. São Paulo: Pearson Prentice Hall, 2010. 809p.
- OOI, R. C. *Balancing a Two-Wheeled Autonomous Robot*. Western Australia, 2003.
- PAULA, A. R. de. *Modelagem e controle do pêndulo invertido sobre duas rodas*. Fortaleza, CE, 2014.
- PEDROSO, C. C. S.; MODESTO, E. L. P. *Sistema de Controle de Pêndulo Invertido*. Monografia (Bacharel em Mecatrônica Industrial), Universidade Tecnológica Federal do Paraná UTFPR, Curitiba, PR, 2017.
- SAGE, A. P. *Optimum Systems Control*. 2. ed..ed. Englewood Cliffs, New Jersey: Prentice-Hall, Inc, 1977. 427p.
- SEGWAY. Acesso: 20 de Abril de 2019, <http://www.segway.com/products/professional/segway-x2-se>.
- SILVA, G. D. *Projeto, modelagem, simulação, construção e controle de um veículo de duas rodas pêndulo invertido*. Uberlândia, MG, 2017.
- SUNDIN, C.; THORSTENSSON, F. *Design and Construction of a Balancing Robot*. Göteborg, Sweden, 2012.