



Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Engenharia Elétrica



Trabalho de Conclusão de Curso

Desenvolvimento e Controle do sistema “Ball and Beam”

Filipe de Magalhães Costa

João Monlevade, MG
2018

Filipe de Magalhães Costa

Desenvolvimento e Controle do sistema “Ball and Beam”

Trabalho de Conclusão de curso apresentado à Universidade Federal de Ouro Preto como parte dos requisitos para obtenção do Título de Bacharel em Engenharia Elétrica pelo Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto.

Orientador: Prof. Víctor Costa da Silva Campos

Coorientador: Prof. Renan Fernandes Bastos

Universidade Federal de Ouro Preto
João Monlevade
2018

C837d

Costa, Filipe de Magalhães.

Desenvolvimento e controle do sistema Ball and Beam [manuscrito] /
Filipe de Magalhães Costa. - 2018.

44f.: il.: color; grafs; tabs.

Orientador: Prof. Dr. Víctor Costa da Silva Campos.

Coorientador: Prof. Dr. Renan Fernandes Bastos.

Monografia (Graduação). Universidade Federal de Ouro Preto. Instituto de
Ciências Exatas e Aplicadas. Departamento de Engenharia Elétrica.

1. Engenharia elétrica. 2. Máquinas elétricas. 3. Controladores PID. I.
Campos, Víctor Costa da Silva. II. Bastos, Renan Fernandes. III. Universidade
Federal de Ouro Preto. IV. Título.

CDU: 621.313

Catálogo: ficha@sisbin.ufop.br



ANEXO IV - ATA DE DEFESA

Aos 9 dias do mês de julho de 2018, às 13 horas e 30 minutos, no bloco H deste instituto, foi realizada a defesa de monografia pelo (a) formando (a) Filipe de Magalhães Costa, sendo a comissão examinadora constituída pelos professores e convidados: Márcio Feliciano Braga, Gabriel Azevedo Fogli, Víctor Costa da Silva Campos e Renan Fernandes Bastos.

O (a) candidato (a) apresentou a monografia intitulada: Desenvolvimento e Controle do sistema “Ball and Beam”. A comissão examinadora deliberou, por unanimidade, pela aprovação do(a) candidato(a), com a nota média 7,8, de acordo com a tabela 1. Na forma regulamentar foi lavrada a presente ata que é assinada pelos membros da comissão examinadora e pelo (a) formando(a).

Tabela 1 – Notas de avaliação da banca examinadora

Banca Examinadora	Nota
Márcio Feliciano Braga	7,5
Gabriel Azevedo Fogli	8,0
Renan Fernandes Bastos	7,5
Víctor Costa da Silva Campos	8,0
Média	7,8

João Monlevade, 09 de julho de 2018.

Víctor Costa da Silva Campos Filipe de Magalhães Costa
Víctor Costa da Silva Campos Filipe de Magalhães Costa
Professor(a) Orientador(a) Aluno (a)

Renan P. Bastos Márcio Feliciano Braga
Renan Fernandes Bastos Márcio Feliciano Braga
Professor(a) Coorientador(a) Professor(a) Convidado(a)

Gabriel Azevedo Fogli
Gabriel Azevedo Fogli
Convidado(a)



ANEXO X - TERMO DE RESPONSABILIDADE

O texto do trabalho de conclusão de curso intitulado “Desenvolvimento e Controle do sistema “Ball and Beam” é de minha inteira responsabilidade. Declaro que não há utilização indevida de texto, material fotográfico ou qualquer outro material pertencente a terceiros sem a devida citação ou consentimento dos referidos autores.

João Monlevade, 09 de Julho de 2018

Filipe de Magalhães Costa
Filipe de Magalhães Costa

Agradecimentos

Primeiramente, gostaria de agradecer aos meus pais, irmão e à minha família pelo apoio incondicional, pelo suporte financeiro e emocional, e pela confiança dedicada, sem vocês nada disso seria possível. A minha namorada por todo o apoio, companheirismo, confiança, paciência, e principalmente, por sempre acreditar no meus sonhos. Aos meus amigos, que fizeram e fazem com que a caminhada se torne menos árdua. Aos professores que me inspiraram e a UFOP pela oportunidade de crescimento pessoal e profissional. Por fim, gostaria de agradecer ao meu orientador Vítor Costa da Silva Campos e co-orientador Renan Fernandes Bastos, por toda a dedicação, disponibilidade e confiança.

"A persistência é o menor caminho do êxito. "

"Charles Chaplin"

Resumo

Este trabalho visa realizar um estudo detalhado dos componentes, descrevendo o desenvolvimento, instrumentação e controle de um sistema “Ball and Beam”. Dando maior ênfase a estrutura criada pelo autor e os problemas encontrados na criação dessa estrutura, assim como os métodos de controle por síntese direta e por lugar geométrico das raízes. Viu-se a necessidade de empregar a estratégia de controle em cascata e ao analisar essa estratégia, mostrou-se ineficaz pelos materiais utilizados e pelo fato do sistema não ser estável em malha aberta, impossibilitando a realização do controle por síntese direta e por isso, conclui-se o controlador PI do ângulo do suporte sem a cascata, pelo método do lugar das raízes. O suporte é similar a uma “calha”, que possibilita a mobilidade interna da bola de metal, cuja finalidade é o controle da posição dessa bola, caracterizando o sistema “Ball and Beam”.

Palavras-chave: Pêndulo Invertido, PI, Controle por síntese direta, Controle por lugar das raízes, Controle em cascata.

Abstract

This work aims to perform a detailed study of the components, describing the development, instrumentation and control of a “ Ball and Beam ” system. The Emphasis is given to the structure created by the author and the problems found in the creation of this structure, as well as the control methods by direct synthesis and by the root locus. It was seen the need to use the cascade control strategy and when analyzing this strategy, it was ineffective by the materials used and the fact that the system is not stable in open mesh, making it impossible to carry out the control by direct synthesis and therefore, the PI controller of the support angle without the cascade is completed by the root locus method. The support is similar to a “ gutter ”, which allows the internal mobility of the metal ball, whose purpose is to control the position of this ball, characterizing the “ Ball and Beam ” system.

Keywords:Inverted Pendulum,PI,Arduino Uno, Control by direct synthesis, Root locus Cascade control.

Lista de ilustrações

Figura 1 – Protótipo final proposto.	3
Figura 2 – Fonte de Alimentação.	4
Figura 3 – Motor DC 6H2LWSKC.	5
Figura 4 – Motor DC LT25GA98.	5
Figura 5 – Conjunto de engrenagem com diâmetros diferentes.	6
Figura 6 – Conjunto de engrenagem com diâmetros iguais.	7
Figura 7 – Novo conjunto de engrenagem com diâmetros diferentes.	7
Figura 8 – Encoder LPD 3806 400BM-G5-24C.	8
Figura 9 – Sinal em quadratura do Encoder.	9
Figura 10 – Arduino UNO.	10
Figura 11 – IDE do Arduino.	10
Figura 12 – Modo de funcionamento de um circuito Ponte H básico.	11
Figura 13 – Diagrama de blocos do CI L298N.	11
Figura 14 – Entradas e saídas do CI L298N.	12
Figura 15 – CI L298N.	12
Figura 16 – Estrutura para movimentação e localização da bola.	13
Figura 17 – Configurações do INA 122.	14
Figura 18 – Diagrama de blocos do sistema com o controlador.	15
Figura 19 – Sistema Ball and Beam.	16
Figura 20 – Diagrama de blocos do controle em cascata do sistema.	18
Figura 21 – Simulação do modelo e aquisição dos parâmetros para controle.	18
Figura 22 – Pontos da simulação do modelo e aquisição dos parâmetros para controle.	19
Figura 23 – Tensão aplicada ao sistema pelo número de pulsos por rotação.	19
Figura 24 – Curva de resposta ao degrau unitário para um sistema de 1 ^a ordem	20
Figura 25 – Detecção do sistema em malha aberta.	22
Figura 26 – Detecção do sistema em malha aberta.	22
Figura 27 – Lugar geométrico das raízes do sistema.	24
Figura 28 – Protótipo montado.	26
Figura 29 – Resultado do sistema de controle da planta.	27
Figura 30 – Protótipo do sistema “Ball and Beam”.	27
Figura 31 – Resultado do sistema de controle em cascata da planta sem a lata.	28
Figura 32 – Resultado do sistema de controle em cascata da planta com a lata.	29

Sumário

1	INTRODUÇÃO	1
1.1	Objetivos	1
1.1.1	Objetivos Gerais	1
1.1.2	Objetivos Específicos	2
1.2	Estrutura do Trabalho	2
2	PROTÓTIPO PROPOSTO	3
2.1	Fonte de alimentação	4
2.2	Motor DC	4
2.3	Engrenagem	5
2.4	Encoder	8
2.5	Arduino	9
2.6	Ponte H	10
2.7	Estrutura para Movimentação e Localização da Bola	13
3	ESTRATÉGIAS DE CONTROLE	15
3.1	Revisão Bibliográfica	15
3.1.1	Método da Síntese Direta (<i>Internal Model Control</i>)	15
3.1.2	Ball and Beam	16
3.1.3	Método do Lugar das Raízes (LGR)	17
3.1.4	Controle em Cascata	17
3.2	Processo de Identificação do Modelo	17
3.3	Controlador PI pelo Método da Síntese Direta	20
3.4	Controlador de Ângulo	23
4	RESULTADOS	26
4.1	Problemas Encontrados	29
5	CONCLUSÃO	33
5.1	Trabalhos Futuros	33
	ANEXO A – CÓDIGO IMPLEMENTADO NO MATLAB	34
	ANEXO B – CÓDIGO IMPLEMENTADO NO ARDUINO UNO	36
	REFERÊNCIAS	43

1 Introdução

O estudo de pêndulos começou a partir das descobertas que Galileu Galilei desenvolveu, acerca dos estudos da natureza das oscilações e da descoberta da periodicidade em movimento pendular (ALBARELLO; PEREIRA; FAORO, 2013). Nos anos 40, despertou o interesse e fomentou o estudo sobre o tema (NAWAWI; AHMAD; OSMAN, 2008), visto a grande capacidade do pêndulo a ser controlado de ser aproximado em sistemas reais de engenharia. Exemplos desses sistemas reais são o controle de um foguete na fase de lançamento, o equilíbrio mecânico e movimento das rodas, como o *segway*, e sistemas robóticos bípedes com caminhadas dinâmicas, entre outros. Além disso, esses sistemas são muito utilizados com finalidade acadêmica visto que o sistema é naturalmente instável, não linear e de fase não mínima, sendo um ótimo modelo de teste de métodos de controle (WAI; KUO; LEE, 2008; OGATA, 2010).

O avanço da tecnologia computacional e de dispositivos eletrônicos, como microprocessadores e microcontroladores, em paralelo aos avanços de pesquisa, estudo de métodos de controle, programação e códigos mais eficientes e robustos permitiram o advento do controle digital (KUO, 1992). E como o esperado, o controle digital é, nos tempos atuais, o mais utilizado em projetos, visto a quantidade de equipamentos que trabalham a partir de pulsos digitais, além de apresentar as vantagens de ser resistente as perturbações em relação à capacidade de variação dos parâmetros frente a ruídos, mais prático em relação a aquisição de dados e processamento e potencialmente mais barato.

Diante deste contexto, este trabalho descreve e projeta um controlador PI digital pelo método da síntese direta, com o intuito de, posteriormente, realizar o controle de um pêndulo invertido agregando ao projeto o problema do “Ball and Beam”.

1.1 Objetivos

Os objetivos deste trabalho foram divididos em objetivos gerais e específicos, e estão descritos abaixo.

1.1.1 Objetivos Gerais

Conhecer e definir as topologias de pêndulos existentes e com isso, desenvolver, montar e controlar um protótipo de pêndulo invertido agregando o problema do “Ball and Beam”.

1.1.2 Objetivos Específicos

- Levantar um modelo do comportamento dinâmico do protótipo a partir de ensaios experimentais;
- Construir e instrumentar o protótipo a ser controlado;
- Projetar e testar diferentes leis de controle no protótipo.

1.2 Estrutura do Trabalho

No Capítulo 1, foi introduzido o tema proposto, assim como os objetivos e expectativas a serem alcançados neste trabalho. No Capítulo 2 é desenvolvido o modelo proposto, mostrando os componentes necessários ao desenvolvimento do protótipo.

No Capítulo 3, a área de estudo é então, investigada, realizando a revisão dos conceitos, abordando os materiais utilizados e os conhecimentos necessários, a fim de criar uma base sólida ao entendimento do trabalho como um todo. É apresentado como são realizados o levantamento e medições dos parâmetros do protótipo, assim como os controles e no Capítulo 4 são mostrados os resultados obtidos e os principais problemas encontrados são então discutidos. Por fim, o Capítulo 5 é destinado à conclusão do trabalho e considerações finais.

2 Protótipo Proposto

O protótipo final utilizado está ilustrado na Figura 1. No desenvolvimento deste protótipo, utiliza-se uma fonte de alimentação para o acionamento do motor DC. Tal motor é acoplado a um eixo de rotação por meio de um conjunto de engrenagens. O eixo é acoplado a outro eixo por meio de uma polia e pelo contato entre os eixos e uma lata, que é utilizada como suporte para uma calha.

Ao segundo eixo é acoplado um *encoder* ótico, utilizado para medir a rotação dos eixos.

Por fim, a calha é utilizada para mover a posição do bola (que se deseja controlar ao final).

O acionamento do motor DC é feito por meio de uma ponte H, enquanto que o sinal de controle da ponte H e a aquisição de dados dos sensores são realizados por um microcontrolador.

Uma visão mais detalhada dos componentes e seus respectivos funcionamentos são apresentados a seguir.

Figura 1 – Protótipo final proposto.



Fonte: Do autor.

2.1 Fonte de alimentação

É utilizada como fonte de alimentação para prover a tensão necessária ao funcionamento do motor DC de + 12 V, uma fonte de alimentação universal de *notebook*, que consiste em uma fonte de tensão regulada e ajustável CA/CC de +12 V a +24 V.

Figura 2 – Fonte de Alimentação.



Fonte: Do autor.

2.2 Motor DC

O motor DC utilizado está ilustrado na Figura 3. Nesse tipo de componente, a variação de velocidade pode ser controlada variando a tensão aplicada na entrada do motor, cujo modelo tem uma tensão de operação de 12 V e velocidade de 4000 rpm.

Como o motor DC é o atuador do eixo do protótipo, a ponte H e o controle realizado neste trabalho agem diretamente sobre o motor, a fim de diminuir a velocidade de rotação.

Quando agregado ao sistema da lata, a calha com a bola dentro, o torque do motor mostrou-se ineficiente para girar o sistema. Por isso, houve a necessidade de trocar o motor por outro de maior torque. Segundo Fitzgerald e Elétricas (2006), a razão entre a potência mecânica e a velocidade angular mecânica é dada por

$$\frac{P_m}{\omega_m} = T_m. \quad (2.1)$$

Percebe-se então que, o motor terá um torque maior, com uma velocidade angular mecânica menor.

O motor DC escolhido para ser utilizado está ilustrado na Figura 4. Nesse tipo de motor com caixa de redução, como esperado, a velocidade é menor, 100 rpm, e a variação

Figura 3 – Motor DC 6H2LWSKC.



Fonte: Do autor.

de velocidade pode ser controlada variando a tensão aplicada na entrada do motor, cujo modelo tem uma tensão de operação de 12 V, assim como o motor anterior e nesse modelo o torque aplicado é de 2 kgf/cm.

Para o projeto, o motor necessita de um maior torque, independentemente da diminuição da velocidade, visto que o sistema não necessita ser rápido, e sim, conseguir fazer os movimentos necessários ao controle da planta.

Figura 4 – Motor DC LT25GA98.



Fonte: Do autor.

2.3 Engrenagem

O primeiro conjunto de engrenagens, chamados de trem de engrenagem está ilustrado na Figura 5. Segundo Leite et al. (2012), a razão de transmissão da velocidade e torque é dado por

$$\frac{N_2}{N_1} = \frac{R_2}{R_1} = \frac{\omega_1}{\omega_2}. \quad (2.2)$$

Em que N_2 e N_1 são os números de dentes da engrenagem motora e movida respectivamente e R_2 e R_1 são os raios das engrenagens. ω_1 e ω_2 são velocidades angulares em (rad/s), que podem também ser descritas em rpm (G.A.IRVING; M.W.ANDRÉS, 2004).

A engrenagem motora utilizada contém 10 dentes, a engrenagem movida contém 36 dentes e a velocidade angular de rotação do eixo do motor é 4000 rpm. Substituindo os valores, obtém-se:

$$\frac{36}{10} = \frac{R_2}{R_1} = \frac{4000}{\omega_2} \quad (2.3)$$

A equação mostra que a engrenagem condutora necessita girar 3,6 vezes para que a engrenagem conduzida complete uma volta em relação eixo, e sua velocidade angular é menor que a engrenagem condutora.

Figura 5 – Conjunto de engrenagem com diâmetros diferentes.



Fonte: Do autor.

O segundo conjunto de engrenagens possui a mesma quantidade de dentes e o mesmo raio, então sua relação de transmissão de velocidade é linear. Toda a velocidade captada no eixo é transferida para o *encoder*.

Contudo, as engrenagens não são realmente iguais, e isto interfere no controle do projeto, como será mostrado no Capítulo 4.

Como mostrado na seção anterior, houve a necessidade de mudar o motor, para outro com um eixo de diâmetro maior, e esse fato acarretou na mudança da engrenagem do primeiro conjunto de engrenagens, ilustrado na Figura 7.

Com essa mudança a engrenagem motora utilizada contém 70 dentes, a engrenagem movida continua com 36 dentes e a velocidade angular de rotação do eixo do motor em 100 rpm. Substituindo os valores, obtém-se:

Figura 6 – Conjunto de engrenagem com diâmetros iguais.



Fonte: Do autor.

$$\frac{36}{70} = \frac{R_2}{R_1} = \frac{100}{\omega_2} \quad (2.4)$$

Pela equação, a engrenagem condutora necessita girar aproximadamente 0,5 vezes para que a engrenagem conduzida complete uma volta em relação eixo, e sua velocidade angular é maior que a engrenagem condutora, porém menor que o antigo conjunto de engrenagens.

Figura 7 – Novo conjunto de engrenagem com diâmetros diferentes.



Fonte: Do autor.

2.4 Encoder

Sensores do tipo encoder são normalmente utilizados para determinar a posição e velocidade do motor ou equipamento acoplado ao eixo do sensor. É comumente usado um sensor com dois pinos, que fornecem dois trens de pulso, em quadratura e sua frequência é proporcional à velocidade de rotação do eixo (BELLINI; BIFARETTI; COSTANTINI, 2003).

Esse trem de pulso ou *clock* do pulso são então processados pelo microcontrolador Arduino, “que conta” o número de pulsos de subida ou de descida dos dois pinos, já que o sensor tem a capacidade de movimentar para ambas as direções e com essa informação, fornece velocidade e posição.

Figura 8 – Encoder LPD 3806 400BM-G5-24C.



Fonte: <https://lista.mercadolivre.com.br/encoder-lpd-3806>.

O sensor encoder LPD 3806 400BM-G5-24C utilizado no protótipo contém dois canais de aquisição de dados. Cada canal consegue contar até 400 pulsos por rotação completa do eixo. Então consegue-se analisar até 800 pulsos.

É possível mensurar a velocidade em radiano por segundo de acordo com (BELLINI; BIFARETTI; COSTANTINI, 2003)

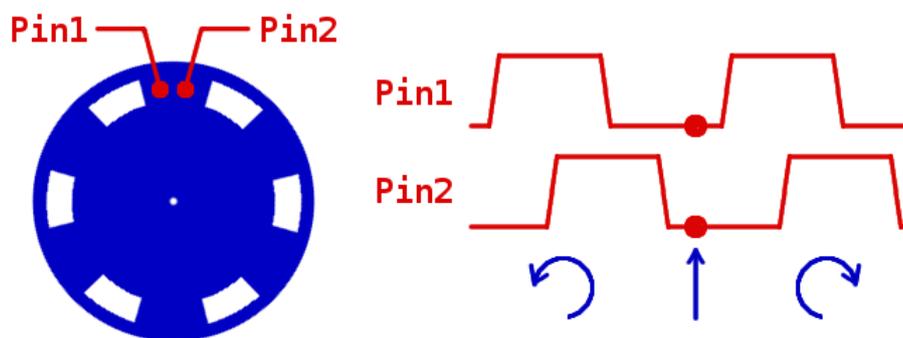
$$\omega = \frac{2\pi n}{NT}, \quad (2.5)$$

Em que $|n|$ = número de pulsos do encoder LPD 3806 400BM-G5-24C em um período de tempo, N = número de pulsos a cada volta, como visto acima é 800, e T é o período de amostragem.

A Figura 9 mostra como é o funcionamento do encoder e a forma de onda em quadratura, visto que as formas de onda estão 90 graus defasadas para os dois pinos.

O sensor detecta os “buracos” à medida que é movimentado o eixo. Quando o círculo abaixo gira primeiro no sentido horário, é detectado primeiro pelo pino 1, e depois pelo pino 2 e quando gira no sentido anti-horário, acontece o inverso.(UNDERSTANDING QUADRATURE ENCODED SIGNALS, 2017).

Figura 9 – Sinal em quadratura do Encoder.



Fonte: <https://www.pjrc.com/teensy/td_libs_Encoder.html>.

2.5 Arduino

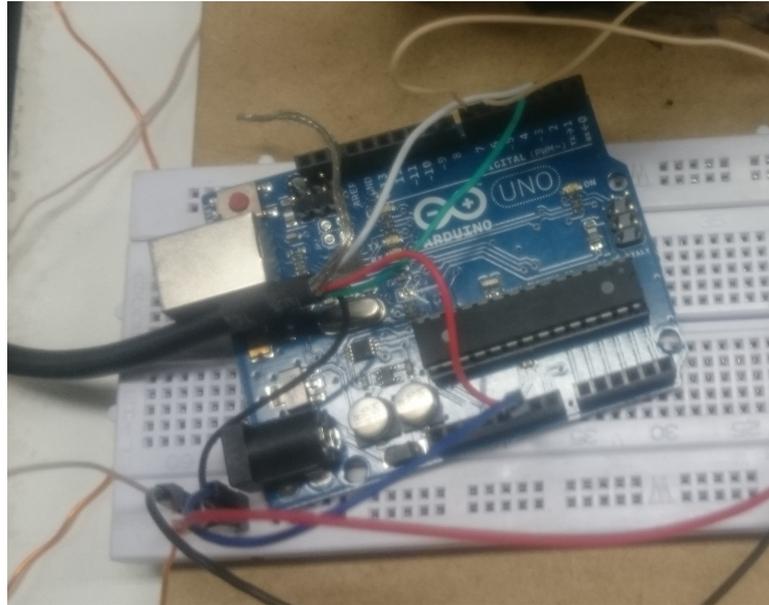
Para este projeto é necessário um equipamento que processe o algoritmo de maneira eficiente, fazendo a comunicação entre as partes físicas e o algoritmo do projeto, a fim de atuar e processar os dados dos sensores e, com isso realizar o controle do protótipo. Optou-se pelo uso do Arduino, uma placa de código livre, usada em vários países, com um custo baixo de aquisição do equipamento, facilitando a montagem do protótipo.

Facilidade somada ao fato da placa de código livre ser usada com finalidade acadêmica, visto que sua utilização é mais simples e existe muito mais documentação de projetos do que outros microcontroladores encontrados para compra.

Como o código do projeto não necessita de muita memória, pode-se utilizar o modelo UNO da Arduino, uma placa que contém o microcontrolador ATmega328P. Possui 14 pinos de entrada/saída digitais, dos quais 6 podem ser utilizados como saídas PWM, 6 entradas analógicas, um cristal de quartzo de 16 MHz, conexão USB, entrada para alimentação e botão de reinicialização (ARDUINOCC, 2017).

Outra facilidade do uso da placa da fabricante é a linguagem de programação, muito próxima do C. Possui um *software* com uma *Integrated Development Environment* (IDE) própria para a programação do microcontrolador mostrada na Figura 11, e por possuir um *bootloader* exclusivo para carregar o código ao microcontrolador da placa, não existe necessidade de retirar o componente para carregar o código.

Figura 10 – Arduino UNO.



Fonte: Do autor.

Figura 11 – IDE do Arduino.

```
encodeste03.cpp | Arduino 1.8.1
encodeste03.cpp
encodeste03.cpp

volatile unsigned int counter = 32000; |
int counter=0;
//motor B
int IN3 = 4; // codigo novo
int IN4 = 7;
int contador = 0;
int sensor=1;
double erro=0;
double erroanci=0;
double wmg;
double r=50;

void setup() {
  Serial.begin(9600);
}

pinMode(IN3,OUTPUT); //codigo novo
pinMode(IN4,OUTPUT); // codigo novo

pinMode(2, INPUT); // set pin to input
pinMode(3, INPUT); // set pin to input
digitalWrite(2, HIGH); // turn on pullup resistors
digitalWrite(3, HIGH); // turn on pullup resistors

Arduino|Banana Uno em C/C++
```

Fonte: Do autor.

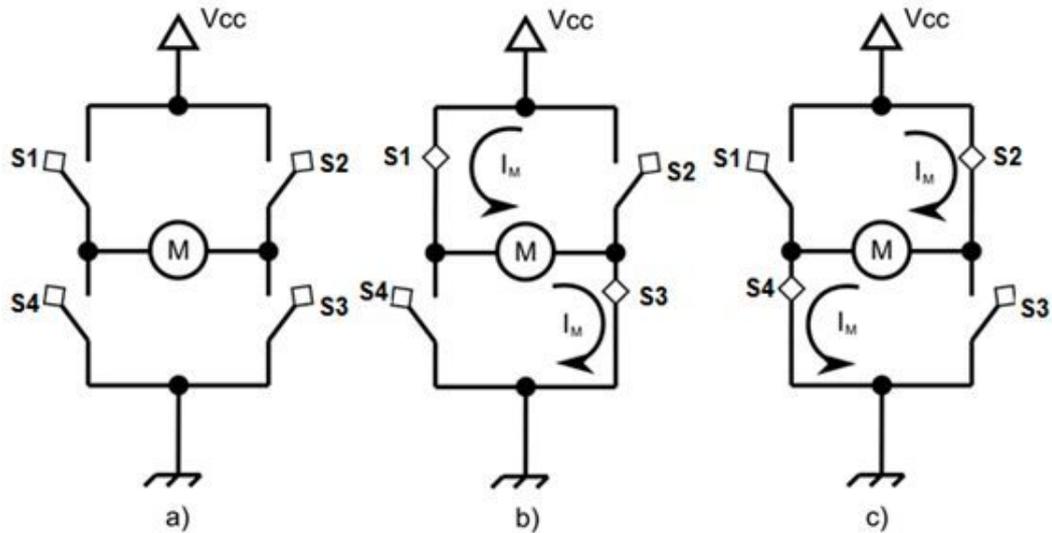
2.6 Ponte H

Utiliza-se o circuito eletrônico ponte H neste projeto para alternar a rotação do motor DC. Em conjunto com o microcontrolador da placa Arduino, consegue-se controlar a direção de rotação do eixo do motor e atuar as engrenagens.

A ponte H tem esse nome por causa do formato em que é montado o circuito, e é constituída basicamente por quatro chaves, como mostrado na figura abaixo.

As chaves S1, S2 e S3, S4 não podem ser acionadas ao mesmo tempo pois geraria um curto circuito dentro do componente utilizado e, como visto na Figura 12, mudando o chaveamento da ponte H, o sentido de rotação do motor é modificado. Porém, ao invés

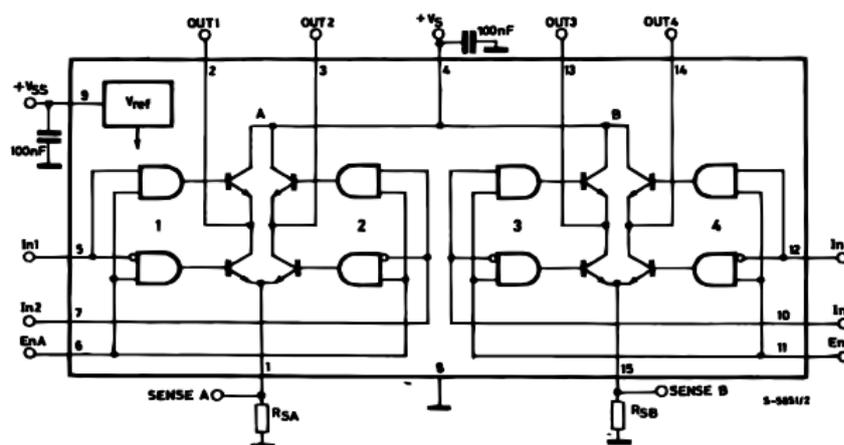
Figura 12 – Modo de funcionamento de um circuito Ponte H básico.



Fonte: <<http://blog.vidadesilicio.com.br/arduino/modulo-ponte-h-l298n-arduino/>>.

de montar uma ponte H com componentes eletrônicos como os transistores, optou-se por utilizar o circuito integrado CI L298N composto por duas pontes H, que possibilita o controle de mais um motor caso necessário, como pode-se analisar pelo diagrama de blocos da Figura 13.

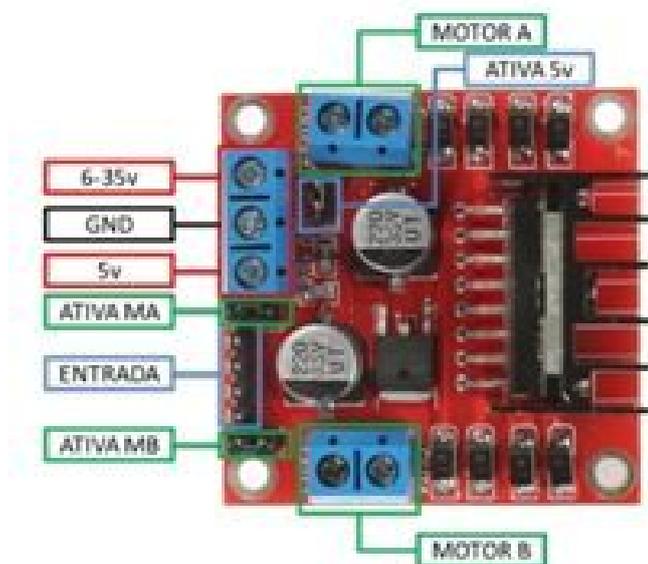
Figura 13 – Diagrama de blocos do CI L298N.



Fonte:<https://www.sparkfun.com/datasheets/Robotics/L298_HB_ridge.pdf>.

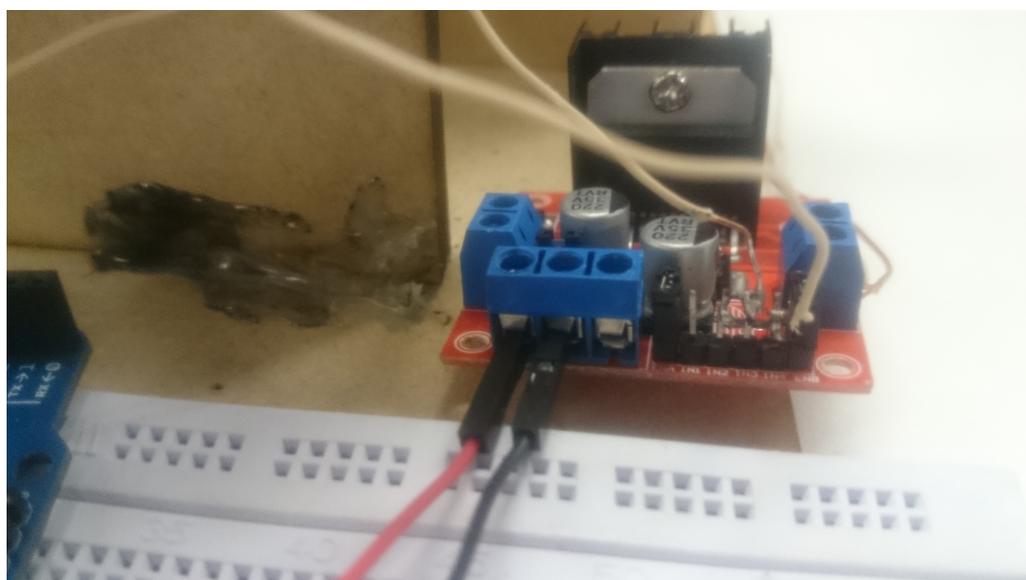
Analisando as Figuras 14 e 15, percebe-se as conexões do motor B, conectando o motor CC utilizado, além das conexões da alimentação da placa com os 12 V provenientes da fonte de alimentação e do ground (GND). No circuito, todos os terras são conectados

Figura 14 – Entradas e saídas do CI L298N.



Fonte: <<http://blog.vidadesilicio.com.br/arduino/modulo-ponte-h-l298n-arduino/>>.

Figura 15 – CI L298N.



Fonte: Do autor.

juntos. Utilizam-se também, as entradas IN3 e IN4, compatíveis com sinais PWM que são utilizados para o controle do motor CC.

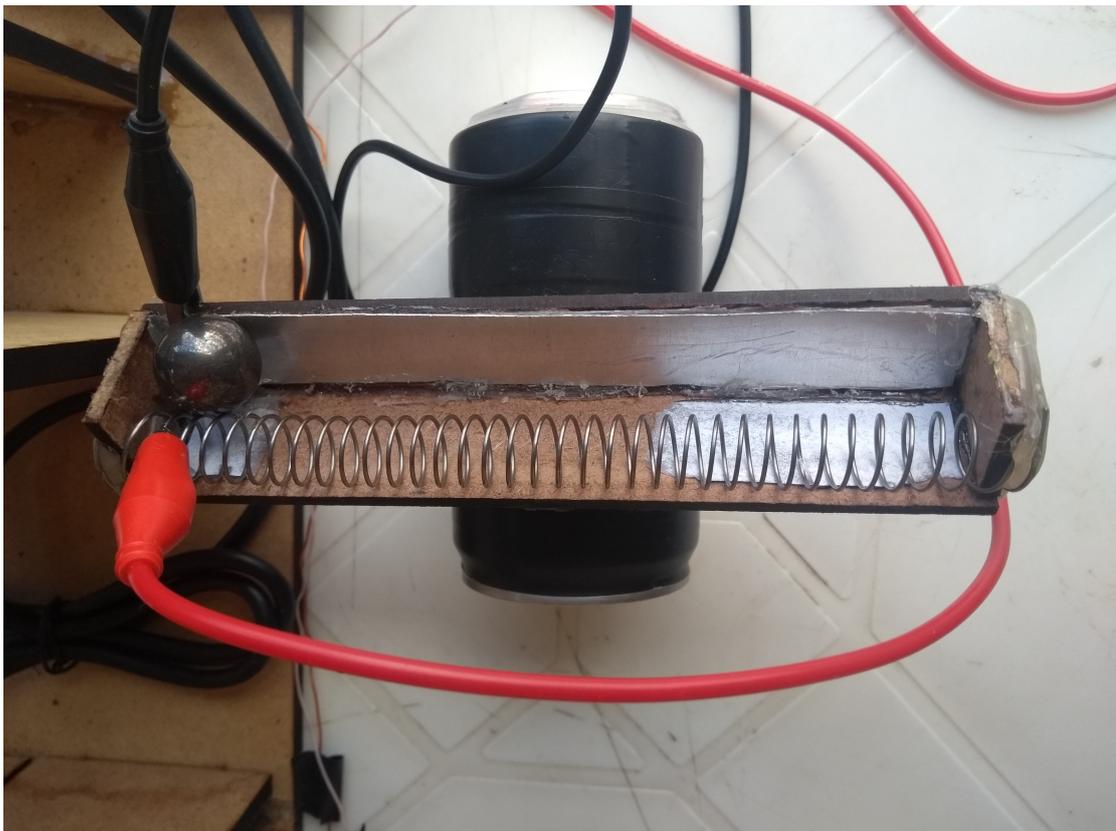
2.7 Estrutura para Movimentação e Localização da Bola

Para movimentar a bola, necessitou-se criar uma calha em V , e para a localização da bola, utiliza-se resistência de chuveiro e chapa de alumínio. O processo de “teste” dos materiais para realizar a movimentação e localização do sistema é detalhado na Seção 4.1.

Para realizar a medição da bola na calha, montou-se um resistor variável, a partir dos materiais condutores citados acima. Como a bola utilizada é feita de ferro, então é ela que realiza o fechamento do contato entre os dois materiais que estão nas laterais da calha, como mostrado na Figura 16. Desta maneira, consegue-se uma variação de valores de tensão, medida a partir da variação de resistência com o fechamento da bola.

A escala de valores medidos não tinham nenhuma variação facilmente perceptível, visto que os valores estavam muito próximos uns dos outros quando possível medir a diferença, impossibilitando a realização do controle da movimentação da bola, então, necessitou-se do uso do amplificador *INA 122*, para dar um ganho no sinal medido, a fim de aumentar a margem de folga da variação da tensão da calha.

Figura 16 – Estrutura para movimentação e localização da bola.

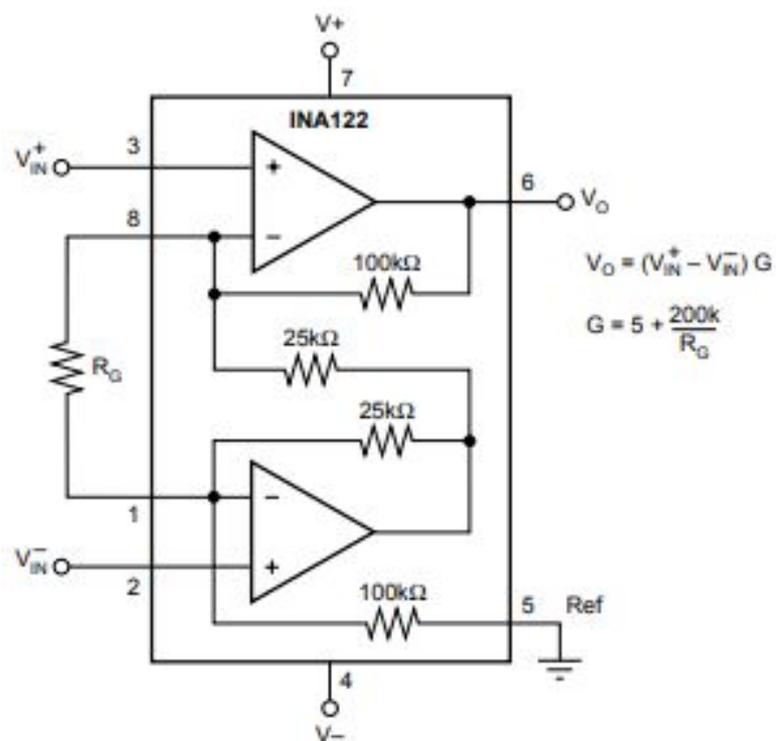


Fonte: Do autor

Opta-se pelo uso desse amplificador por não necessitar de fonte simétrica para a alimentação, além da alimentação ser de 2,2 V até 36 V e pela robustez (DATASHEET

INA122-TEXAS INSTRUMENTS, 2018). A Figura 17 mostra as configurações do INA 122, assim como a equação de ganho do amplificador.

Figura 17 – Configurações do INA 122.



Fonte: <<http://www.ti.com/lit/ds/symlink/ina122.pdf>>.

3 Estratégias de Controle

3.1 Revisão Bibliográfica

Antes de relatar a estrutura aplicada na construção e controle do protótipo, faz-se necessário ter o conhecimento da teoria do problema no qual é desenvolvida. É fundamental para a total absorção dos conhecimentos desenvolvidos neste trabalho, conhecer o método da síntese direta, utilizado para desenvolver o PI, e o problema do “Ball and Beam”.

3.1.1 Método da Síntese Direta (*Internal Model Control*)

O método por síntese direta é utilizado para a sintonia do controlador PI do protótipo. Segundo Campos e Teixeira (2006), o método foi proposto por Garcia e Morari em 1982.

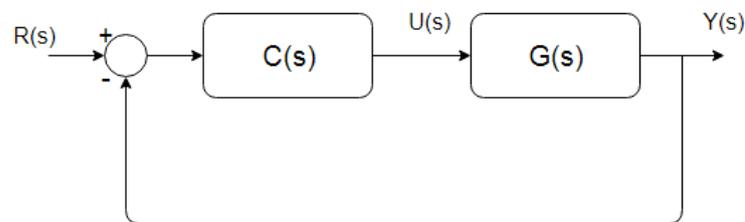
O modelo consiste em selecionar e regular um único parâmetro de desempenho da planta, que é a constante de tempo do filtro τ . Essa sintonia é também conhecida como o método do modelo interno (IMC), usando o método da síntese direta (CAMPOS; TEIXEIRA, 2006).

Esse parâmetro é então, definido como o tempo de acomodação do sistema e pode ser interpretado como a “velocidade” que o sistema controlado possui em malha fechada.

Após identificada a planta, é realizada a sintonia do controlador PI pelo método da síntese direta.

O diagrama de blocos do sistema é mostrado na Figura 18.

Figura 18 – Diagrama de blocos do sistema com o controlador.



Fonte: Do autor

A equação em malha fechada do sistema é dada por:

$$G_F(s) = \frac{G(s)C(s)}{1 + G(s)C(s)} \quad (3.1)$$

Sendo $G_F(s)$ o sistema em malha fechada. Podemos isolar o controlador em função das outras funções de transferência em (3.1), obtendo assim (CAMPOS; TEIXEIRA, 2006)

$$C(s) = \frac{1}{G(s)} \frac{G_F}{(1 - G_F)} \quad (3.2)$$

Considerando uma malha fechada desejada de primeira ordem e sem atraso de transporte, podemos escrever uma $G_F(s)$ desejada como (CAMPOS; TEIXEIRA, 2006)

$$G_F^d = \frac{1}{\tau_d s + 1}. \quad (3.3)$$

Substituindo em (3.3), (3.12), (3.2), e após manipulações algébricas, obtém-se:

$$C(s) = \frac{\tau s + 1}{k\tau_d s} = \frac{\tau}{k\tau_d} + \frac{1}{k\tau_d} * \frac{1}{s} \quad (3.4)$$

Em que k_P é apresentado em (3.5) e k_i é apresentado em (3.6).

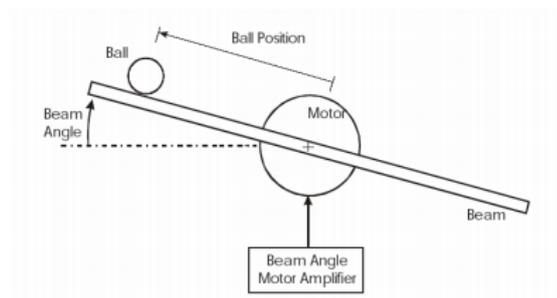
$$k_P = \frac{\tau}{k\tau_d} \quad (3.5)$$

$$k_i = \frac{1}{k\tau_d} \quad (3.6)$$

3.1.2 Ball and Beam

“Ball and Beam” é um sistema “simples” de se fabricar o protótipo, como mostrado pela Figura 19.

Figura 19 – Sistema Ball and Beam.



Fonte: (VIRSEDA, 2004)

Consistindo basicamente em um suporte, que proporciona a mobilidade interna de uma bola ao longo do mesmo. O eixo do suporte é montado de maneira a permitir a inclinação em torno do seu próprio eixo (VIRSEDA, 2004).

O controle do sistema consiste em regular a posição da bola, alterando a velocidade e o ângulo do suporte. Segundo Virseda (2004), a dificuldade do controle consiste exatamente no fato da bola não permanecer no mesmo lugar, mas se mover ao longo do suporte com

uma aceleração proporcional à inclinação do suporte. O intuito é gerar um controlador e introduzi-lo no sistema para estabilizá-lo, mantendo a bola na posição desejada do suporte. Assim como o pêndulo invertido, esse sistema também é muito utilizado para o ensino de controle de sistemas na engenharia (BALL AND BEAN, 2017).

3.1.3 Método do Lugar das Raízes (LGR)

O método do lugar das raízes foi desenvolvido por Evans em 1948, e juntamente com o métodos baseado na resposta em frequência, correspondem aos alicerces da teoria de clássica de controle (DORF; BISHOP, 2001).

A técnica consiste em um método gráfico no qual as raízes da equação característica são localizadas em função dos parâmetros, que no caso deste trabalho são os ganhos do sistema, porém, pode ser qualquer parâmetro da função transferência do sistema em malha aberta. O método do lugar das raízes mostra como as mudanças na características do sistema em malha aberta influenciam as características do sistema em malha fechada, no plano S (FRANKLIN; POWELL; WORKMAN, 1998).

Utilizou-se neste trabalho o toolbox *Sisotool* do *Matlab* para auxiliar no projeto do controlador PID do ângulo.

3.1.4 Controle em Cascata

Este trabalho propõe o uso da estratégia de controle em cascata, visto a necessidade da implementação de dois controladores em conjunto para superar o atrito, folgas e outras perturbações agregados no sistema.

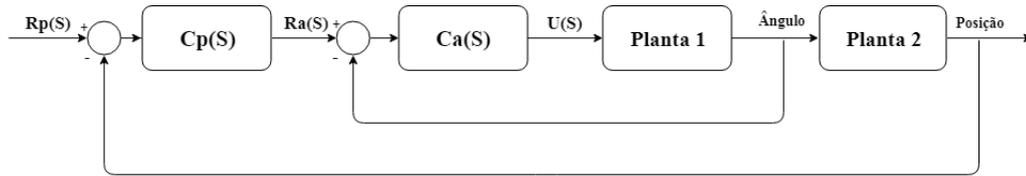
A estratégia de controle em cascata é uma estratégia muito utilizada para simplificar o projeto de controladores para sistemas que podem ser subdivididos em subsistemas. A ideia é dividir o sistema em uma malha interna e uma malha externa. Dessa forma, a ordem dos subsistemas é menor do que a do sistema completo e o projeto de controladores se torna mais simples. De modo que a dinâmica da malha interna possa ser desconsiderada durante o projeto da malha externa, é comum supor que a dinâmica da malha interna é muito mais rápida do que a dinâmica da malha externa e por isso, pode-se considerar a função transferência dela unitária para sintonizar a malha externa.

Com esta estratégia, consegue-se realizar o controlador de ângulo e, a partir da referência gerada para este controlador, implementar o controlador de posição.

3.2 Processo de Identificação do Modelo

O processo de identificação do modelo a ser controlado foi realizado pelo *software* Matlab, por meio dos parâmetros obtidos no arduino Uno. Para realizar a identificação,

Figura 20 – Diagrama de blocos do controle em cascata do sistema.

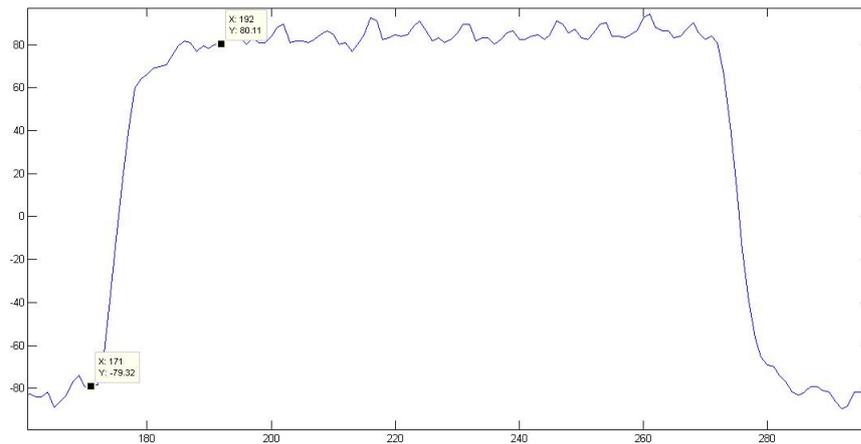


Fonte: Do autor

aplicou-se um degrau de tensão (correspondente a um degrau no valor de pwm utilizado para atuar o motor), e foi medido o comportamento da velocidade de rotação.

Pela Figura 21 gerada no *software* Matlab, pode-se aproximar o sistema por um modelo de primeira ordem. Na mesma figura, consegue-se obter os parâmetros necessários a implementação do controle.

Figura 21 – Simulação do modelo e aquisição dos parâmetros para controle.



Fonte: Do autor.

A partir das Figuras 21 e 23, podemos encontrar os parâmetros do modelo de primeira ordem, como ganho e tempo de acomodação.

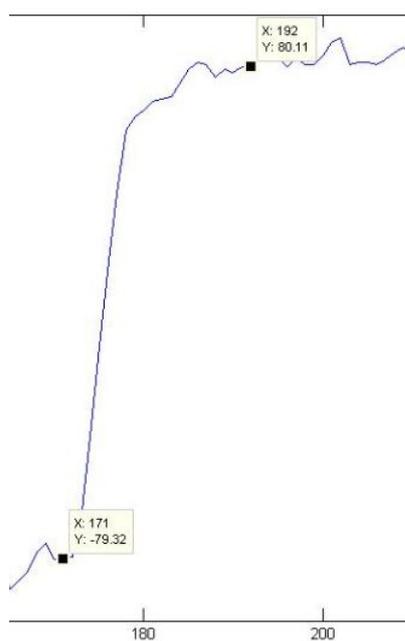
Para a simulação e implementação do modelo, optou-se por utilizar um tempo de acomodação $T = 4 \tau$, segundo Ogata (2010). Para o sistema de primeira ordem, a relação entre entrada e saída é dada por

$$T_{98\%} = 4\tau \tag{3.7}$$

$$G_s = \frac{C_S}{R_S} = \frac{K}{\tau s + 1}. \tag{3.8}$$

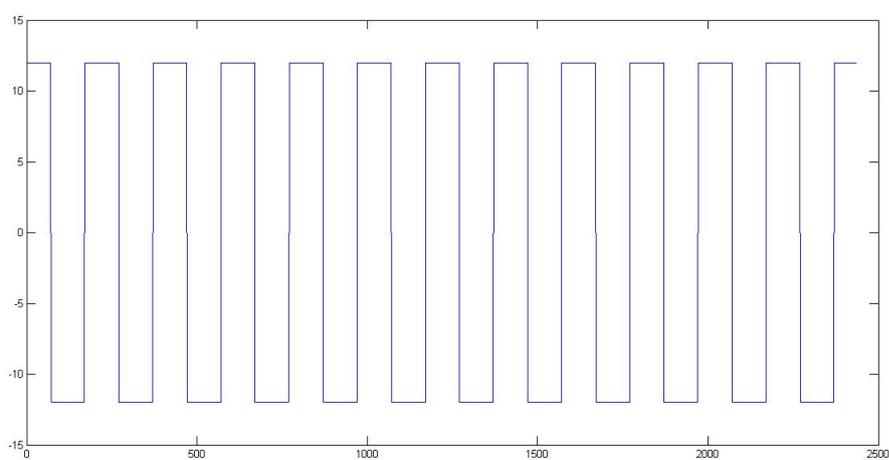
Em que C_S e R_S são saída e entrada do sistema, respectivamente. Como aplicou-se uma entrada em degrau e o atraso do sistema foi desprezado, consegue-se obter os

Figura 22 – Pontos da simulação do modelo e aquisição dos parâmetros para controle.



Fonte: Do autor.

Figura 23 – Tensão aplicada ao sistema pelo número de pulsos por rotação.



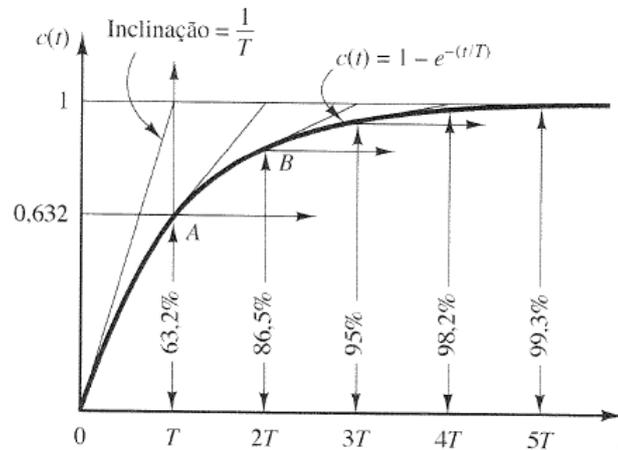
Fonte: Do autor.

parâmetros, como visto na Figura 21, e calcular o tempo de acomodação e o ganho $|K|$ do sistema, dado por

$$4\tau = (192 - 171) * (0,01) \quad (3.9)$$

$$\tau = 0,0525 \quad (3.10)$$

Figura 24 – Curva de resposta ao degrau unitário para um sistema de 1ª ordem



Fonte: Ogata (2010)

$$K = \frac{160}{24} = 6,667 \quad (3.11)$$

Portanto, a função transferência em malha aberta do sistema a ser controlado é

$$G_s = \frac{127}{s + 19,05}. \quad (3.12)$$

3.3 Controlador PI pelo Método da Síntese Direta

Utilizando as equações da Seção 1.2.1. Fazendo

$$\tau_d = \tau = 0.0525 \quad (3.13)$$

$$k = 6.667 \quad (3.14)$$

e substituindo os valores em (3.5) e (3.6), obtêm-se $k_p = 0.1500$ e $k_i = 2.8570$.

O Controlador PI encontrado é

$$G_c(s) = \frac{0,15s + 2,857}{s}. \quad (3.15)$$

Para gerar a ação de controle no arduino, optou-se por discretizar o controlador projetado utilizando o comando `c2d` com a transformada de Tustin (transformada bilinear) no *software* Matlab, com um tempo de amostragem de 0,01 segundos. É empregado um tempo de amostragem pequeno visto que, quanto menor o tempo de amostragem, mais

próximo o sinal discreto gerado se torna do sinal analógico (FRANKLIN; POWELL; WORKMAN, 1998).

O controlador é representado pela transformada z, como

$$\frac{U(z)}{E(z)} = G_{cd}(z) = \frac{0,1643z - 0,1357}{z - 1}. \quad (3.16)$$

Sabendo que multiplicar por z no domínio z é avançar no tempo discreto.

$$U(z)(z - 1) = E(z)(0,1643z - 0,1357) \quad (3.17)$$

E, com o auxílio da tabela de transformada z, é extraído o sinal discreto de (3.17), a fim de ser implementado o controlador discreto no arduino, resultando em

$$u[n + 1] - u[n] = 0,1643e[n + 1] - 0,1357e[n]. \quad (3.18)$$

Deslocando de volta ao presente a (3.18) tem-se que a equação de diferenças aplicado ao microcontrolador para a realização do controle é dado por

$$u[n] = u[n - 1] + 0,1643e[n] - 0,1357e[n - 1]. \quad (3.19)$$

Como foi necessário a troca do motor, também foi necessário encontrar outra função transferência para o sistema, pois, com a mudança do motor, muda-se também as características do sistema. Desenvolve-se o controlador de ângulo diretamente por meio da síntese direta e também, fazendo cascata usando uma malha interna de velocidade.

Realizando o procedimento semelhante ao anterior, cuja as figuras para o cálculo do parâmetro K são mostrados nas figuras 25 e 26.

Fazendo

$$K = \frac{\text{Pulsos}}{\text{tensão}} = \frac{65}{16} \quad (3.20)$$

e

$$\tau = 10ms \quad (3.21)$$

Portanto, a função transferência em malha aberta do sistema a ser controlado é:

$$G_s = \frac{4,006}{0,01s + 1} \quad (3.22)$$

Utilizando as equações da seção 1.2.1.

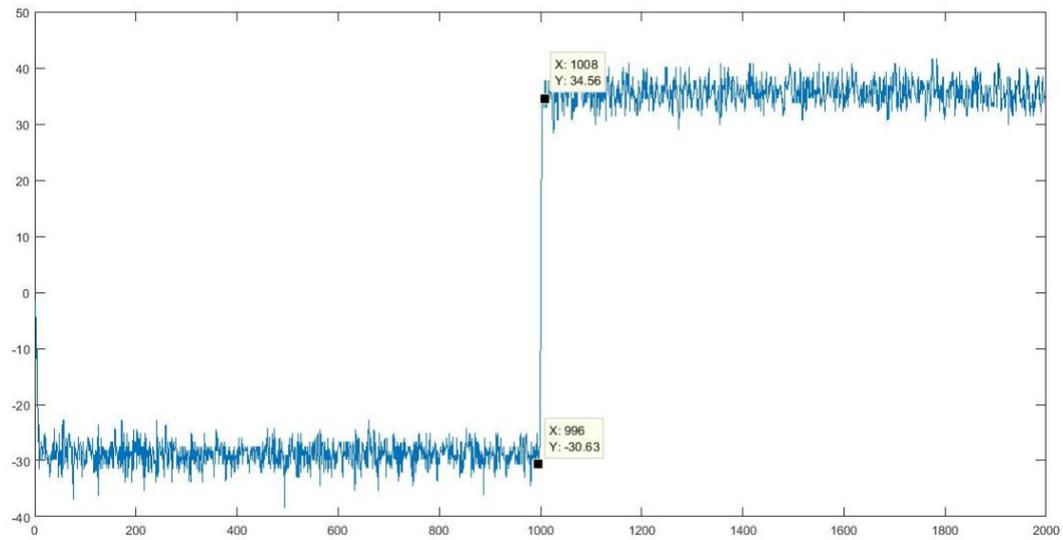
Fazendo

$$\tau_d = \tau = 0.01 \quad (3.23)$$

$$k = 4,006, \quad (3.24)$$

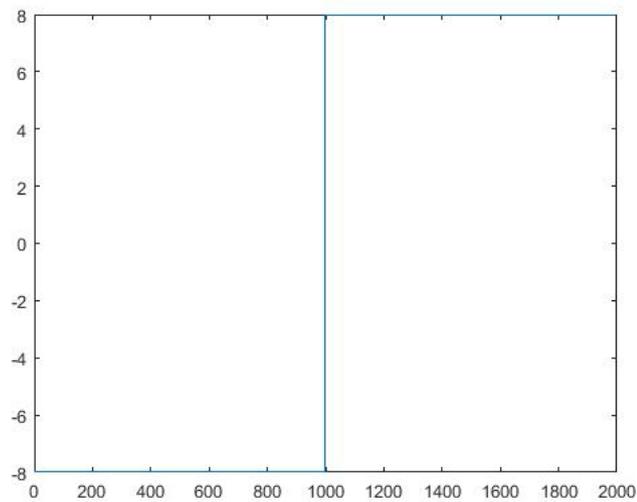
e substituindo os valores em (3.5) e (3.6), obtêm-se $k_p = 0.2496$ e $k_i = 24.9626$.

Figura 25 – Detecção do sistema em malha aberta.



Fonte: Do autor.

Figura 26 – Detecção do sistema em malha aberta.



Fonte: Do autor.

O Controlador PI encontrado é

$$G_c(s) = \frac{0,2496s + 24,96}{s}. \quad (3.25)$$

Utilizando o mesmo método de discretização para o controlador, e o de obtenção da equação de diferenças que é implementado no Arduino, obtêm-se

$$\frac{U(z)}{E(z)} = G_{cd}(z) = \frac{0,3744z - 0,1248}{z - 1} \quad (3.26)$$

$$u[n] = u[n - 1] + 0,3744e[n] - 0,1248e[n - 1]. \quad (3.27)$$

Primeiro, a função transferência da velocidade foi gerada por meio da síntese direta, a partir do gráfico da resposta ao degrau. Como o controlador mostrou-se muito agressivo, optou-se pelo uso do controle em cascata com a implementação do controlador de ângulo.

3.4 Controlador de Ângulo

Para o controle em cascata, supõe-se que a malha interna tem ganho unitário e como o ângulo é a integral da velocidade, é necessário então, adicionar um polo na origem para obter o sistema, isto é, a dinâmica do ângulo em função da velocidade é $1/s$. Gera-se então um PI que é o controlador de ângulo do sistema vezes $1/s$ que é a dinâmica considerada do ângulo em função da velocidade.

Considerando o projeto do controlador de ângulo como:

$$C(s) = \frac{K_{\theta} \cdot s + Ki_{\theta}}{s} \cdot \frac{1}{s} \quad (3.28)$$

E a equação em malha fechada é dada por:

$$G(s) = \frac{K_{\theta} \cdot s + Ki_{\theta}}{s^2 + K_{\theta} \cdot s + Ki_{\theta}} \quad (3.29)$$

Onde, K_{θ} é apresentado em (3.30) e Ki_{θ} é apresentado em (3.31).

$$K_{\theta} = 2\zeta\omega_n \quad (3.30)$$

$$Ki_{\theta} = \omega_n^2 \quad (3.31)$$

É escolhido para o projeto o tempo de assentamento igual a 5 segundos e $\zeta = \frac{\sqrt{2}}{2}$, visto que, com esse valor de zeta obtêm-se bons resultados.

Como $T_s = \frac{4}{\zeta\omega_n}$, obtém $\omega_n = 1,13$.

E substituindo estes valores nas equações (3.30) e (3.31), obtêm-se $k_p = 1.16$ e $k_i = 1.281$.

O Controlador PI encontrado é respectivamente:

$$G_c(s) = \frac{1,6s + 2,857}{s} \quad (3.32)$$

E representado pela transformada z, como

$$\frac{U(z)}{E(z)} = G_c(z) = \frac{1,606z - 1,594}{z - 1}. \quad (3.33)$$

Realizando o mesmo procedimento pra a obtenção da equação de diferenças a ser aplicado ao microcontrolador para a realização do controle, resulta-se em

$$u[n] = u[n - 1] + 1.606e[n] - 1.594e[n - 1]. \quad (3.34)$$

O resultado desta técnica é não satisfatória. Não é possível realizar o controle por síntese direta porque o sistema não é estável em malha aberta.

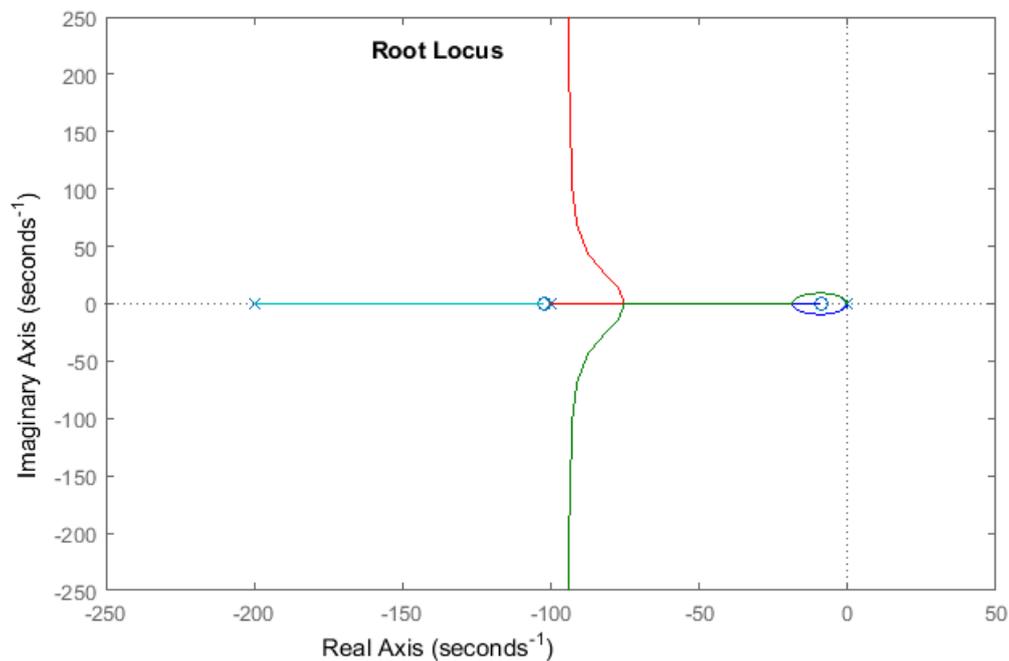
Diferentemente do controle em cascata no qual o controlador de ângulo produzido é utilizado como a referência do controlador da velocidade, para o LGR é suposto que o controle da velocidade e o controle do ângulo de maneira conjunta. É a transformada da velocidade junto com um integrador.

Consegue-se obter a função de transferência em malha aberta do sistema multiplicando por $\frac{1}{s}$ a função transferência da velocidade (3.22), cuja a equação resultante é dada por

$$G_s = \frac{4,006}{0.01s^2 + s}. \quad (3.35)$$

E o lugar geométrico das raízes geradas a partir de (3.35) é ilustrado na Figura 27.

Figura 27 – Lugar geométrico das raízes do sistema.



Fonte: Do autor

Quando é necessário acrescentar um zero ao sistema, tem-se que acrescentar um polo também, visto que, normalmente quer-se acrescentar zeros, pois, pode ter mais polo que zero no controlador, mas o posto não é válido. Para a aquisição do controlador, posicionou-se o zero o mais próximo da zona desejada no intuito de atrair o lugar das raízes.

O Controlador desenvolvido por esse método é dado por:

$$G_c(s) = \frac{13.993(s + 8.734).(s + 102.3)}{s(s + 200)} \quad (3.36)$$

Para gerar a ação de controle no arduino, optou-se por discretizar o controlador projetado da mesma maneira que os controladores anteriores, isso é, utilizando o comando `c2d` com a transformada de Tustin (transformada bilinear) no *software* Matlab, com um tempo de amostragem de 0,01 segundos. O controlador é representado pela transformada z , como

$$G_c(z) = \frac{11.039(z - 0.9163).(z - 0.323)}{z(z - 1)}. \quad (3.37)$$

Realizando o mesmo método pra a obtenção da equação de diferenças a ser aplicado ao microcontrolador para a realização do controle, obtém-se

$$u[n] = u + 11.039u[n - 1] + 13.6806e[n] + 3.26752e[n - 1]. \quad (3.38)$$

4 Resultados

O circuito do modelo para implementar e processar os dados foram agrupados na mesma plataforma de madeira da estrutura. A figura 28 mostra o protótipo final proposto.

Figura 28 – Protótipo montado.



Fonte: Do autor

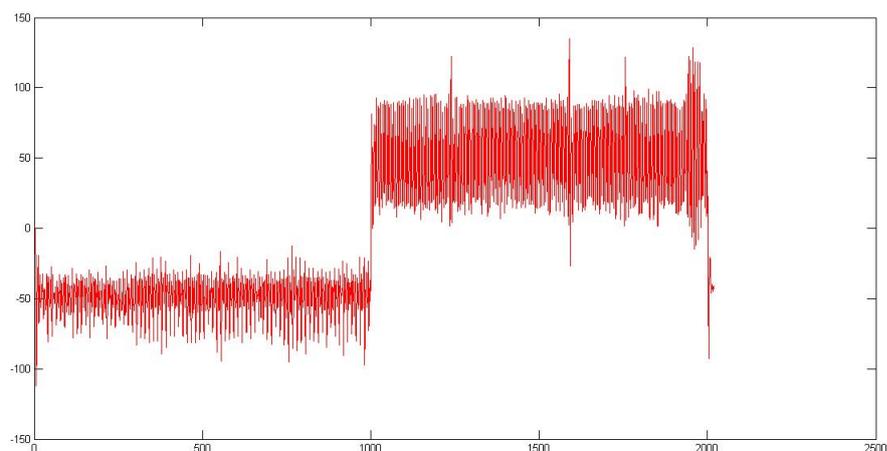
Primeiro, realiza-se a programação do microcontrolador, calculando a velocidade (v) a partir de (2.5), com os parâmetros anteriormente citados e com o tempo de amostragem igual a 0,01 segundos.

Após isso, o controlador PI calculado é codificado no Arduino. Considerando que a referência da planta é 50 rad/s, obtém-se então, como resposta ao controlador desenvolvido a Figura 29.

A velocidade manteve a média de ± 50 , porém, mudando o sentido de rotação do protótipo, também muda o comportamento do sistema, acrescentando mais perturbações, como mostrado na Figura 29.

Este comportamento da planta provavelmente é devido à folga e colocação das engrenagens, com isso aumentam os números de ruídos e distúrbios no modelo controlado.

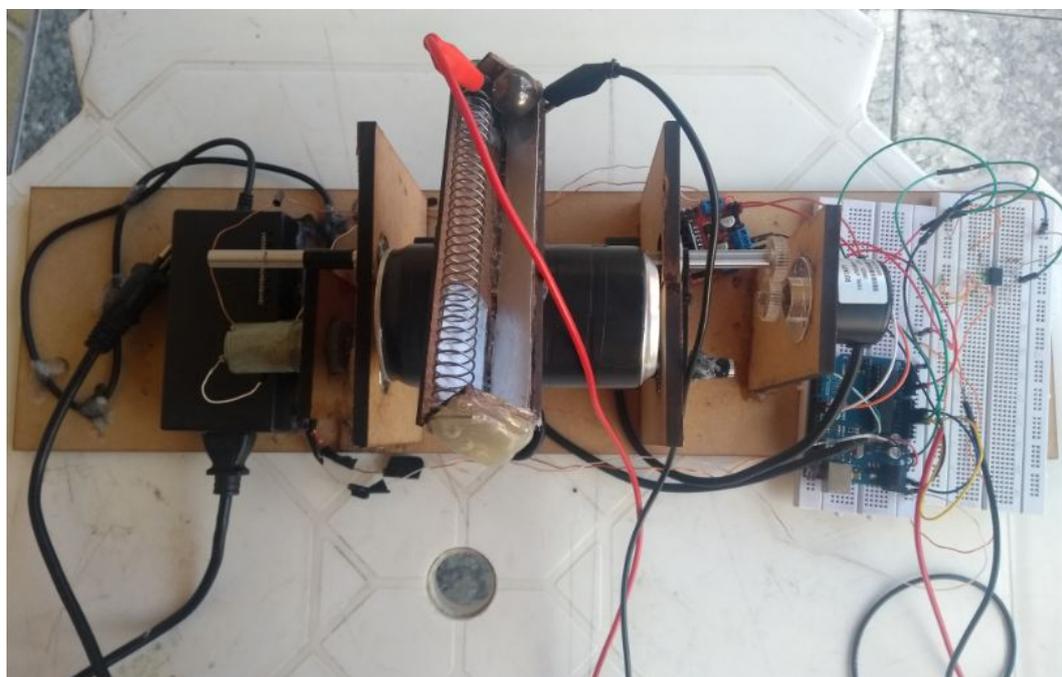
Figura 29 – Resultado do sistema de controle da planta.



Fonte: Do autor

A figura 30 mostra o protótipo final proposto para implementar e processar os dados do “Ball and Beam”.

Figura 30 – Protótipo do sistema “Ball and Beam”.



Fonte: Do autor

Para chegar ao objetivo final, que é controlar a bola na calha pensou-se primeiramente, em desenvolver um controlador em cascata, usando a velocidade para controlar a posição. Como não é usado um sensor próprio para a medição de velocidade, mas sim o encoder para estimar a velocidade, fazendo com que o controlador amplifique os ruídos do

encoder e por conta disso, obteve-se um controlador de ângulo muito oscilatório. A solução foi desenvolver um controlador de ângulo diretamente, sem a cascata com o controlador de velocidade.

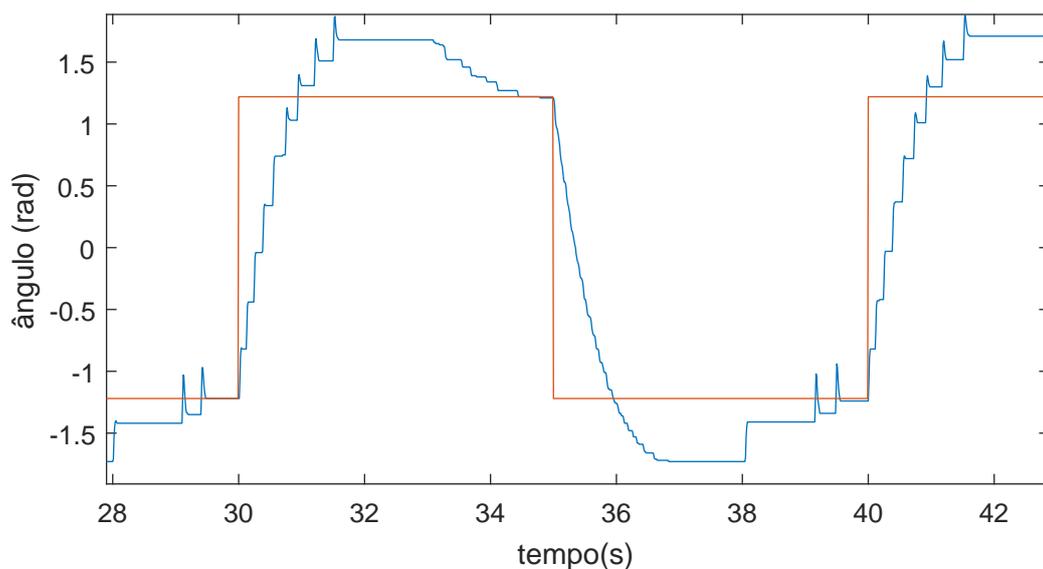
A amplificação ocorria porque a velocidade era calculada a partir da derivada do ângulo, o que implicava na amplificação dos ruídos.

O controlador PI do ângulo calculado (3.37), é codificado no Arduino como apresentado no Anexo B. Considerando que a referência de ângulo é 1,22 rad, obtém-se então, como resposta ao controlador desenvolvido as Figuras 31 e 32, que representam o controlador sem a calha e lata, e com a lata com a calha respectivamente, visto a maior capacidade de entendimento da planta desenvolvida, após esta comparação.

O ângulo manteve a média de $\pm 1,22$ rad, em ambas as figuras, porém, conclui-se pela Figura 31, que o controlador têm um *overshoot* de 15 %. A causa desse *overshoot* é principalmente o atrito da planta, porque o eixo agarra. Percebe-se pela Figura 32, um *overshoot* maior que somente o eixo rodando sem a lata e calha.

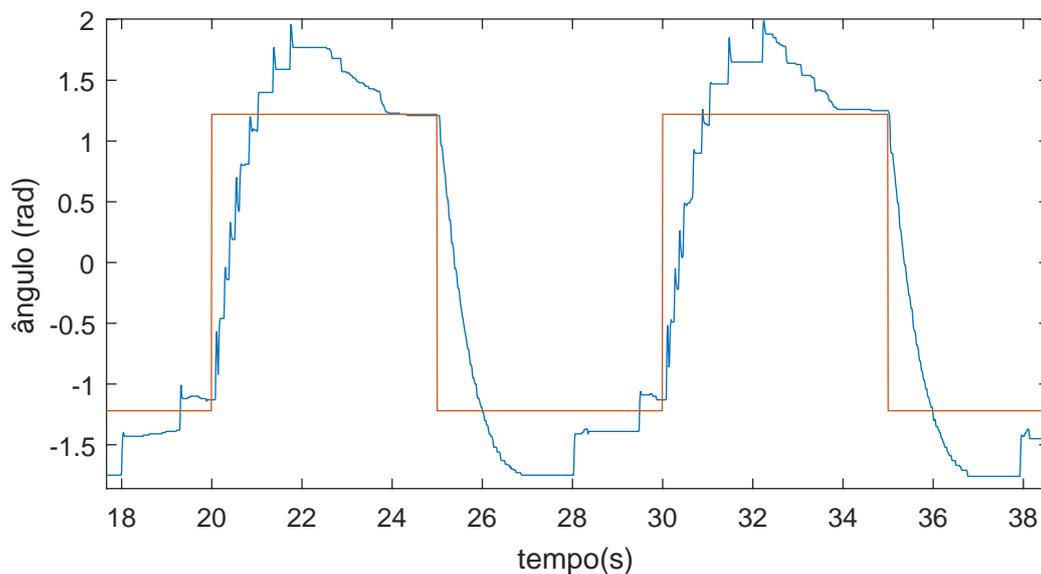
A explicação para este comportamento da planta se encontra no fato que, o sistema acaba tendo um comportamento com saltos porque existe um atrito e folga nas engrenagens que levam a um valor maior de tensão pra mudar a posição do eixo. Por conta desse atrito, o integrador vai acumular os valores de forma a vencer o atrito. Além disso, o problema é que o atrito dinâmico e cinemático é diferente, fazendo com que acumule erros e dificulte a solução.

Figura 31 – Resultado do sistema de controle em cascata da planta sem a lata.



Fonte: Do autor

Figura 32 – Resultado do sistema de controle em cascata da planta com a lata.



Fonte: Do autor

4.1 Problemas Encontrados

Além do problema da engrenagem já citado acima, ocorreram outros problemas que serão discursados nesta secção, em ordem cronológica. O primeiro problema encontrado foi conseguir fazer com que a lata rodasse no eixo sem que deslizasse, isto é, resolver o problema da falta de atrito entre a lata de alumínio e os eixos, visto que até nesta etapa do projeto, pensou-se que não seria necessário a troca por um motor com caixa de redução, que aumenta o torque do motor. Primeiro, testou-se a lata com e sem o conteúdo, e ficou bastante claro que a lata precisa estar cheia para conseguir rodar, mas para resolver o problema, utilizou-se fita isolante, envolvendo a lata de alumínio, e os eixos. O resultado obtido até então, foi satisfatório.

Resolvida a etapa da montagem, o próximo desafio era colar a calha que foi montada como uma caixa retangular de $25 \times 4 \times 4 \text{ cm}$, sem a parte de cima, para conseguir colocar a bola de metal de aproximadamente $1,5 \text{ cm}$ de diâmetro e testar se o problema anteriormente explicado tinha mesmo sido resolvido. Porém, após os testes, ficou claro que somente a fita isolante não seria a solução. Colocou-se então, envolto nos eixos pedaços de borracha. Com isso melhorou consideravelmente o projeto mas, ficou claro que o problema não seria resolvido assim, então pensou-se em duas opções para o solucionar do problema. Trocar o motor por outro, com o torque maior ou então trocar a calha por outra com as dimensões menores, a fim de diminuir o peso do conjunto calha-bola.

Como a troca do motor era mais complexa, pela necessidade de mudar toda a parte estrutural visto que, o outro motor é menor e tem um eixo do motor mais grosso, que acarretaria a mudança também da engrenagem acoplada ao eixo, optou-se então pela

mudança da calha, diminuindo para $25 \times 4 \times 4$ cm, mantendo o formato retangular.

Após a montagem do sistema com a nova calha e o teste, o motor ainda não tinha torque suficiente para movimentar a lata e a calha, sendo necessário trocar o motor. Com a troca do motor e da necessidade da troca da engrenagem, conseguiu-se solucionar o problema da movimentação da lata para a movimentação da bola.

A próxima etapa do processo de conclusão do projeto é a medição da bola dentro da calha. Para conseguir esse objetivo, pensou-se em colocar materiais com determinada resistências na base e em uma das laterais da calha, e por meio da bola de metal, fechar os contatos entre estas duas resistências, fazendo então uma resistência variável. Com isso, aplicando uma determinada tensão em um dos lados destes materiais, consegue-se fazer uma medição de valores e assim realizar o ajuste e controle desta bola dentro da calha.

O primeiro material analisado foram chapas de alumínio finas. Testado esse material primeiramente com o multímetro, percebeu-se dois problemas com relação a este tipo de medição da posição da bola. O primeiro é com relação a condutividade do material, que não pode ser muito boa pois, senão não retorna valores precisos. E o outro problema é exatamente com relação a precisão do material, que não varia de forma linear, impossibilitando o controle preciso da bola na calha. O alumínio tem uma resistência pequena, então a escala de medição foi muito ruim, e para contornar esse imprevisto, pensou-se em realizar um divisor de tensão para aumentar o valor aferido e fazer o controle da corrente que passa sob o sistema. Neste caso o método pensado não foi eficiente, utilizando então um amplificador de instrumentação para dar um ganho no sinal aferido. Neste ponto do trabalho, conseguiu-se fazer a medição da bola, corrigir o problema do ganho do sinal, porém com as placas de alumínio não conseguia ter uma medição com variação linear. Além disto, percebeu-se que o fechamento dos contatos com a bola de metal não estava sendo realizado de maneira correta, com a movimentação da bola dentro da calha, a mesma não fechava os contatos entre as placas em todos os pontos.

Para corrigir este imprevisto, colocou-se papel entre as chapas de alumínio e o a madeira, para diminuir o espaço entre as chapas mas, ao invés de solucionar o imprevisto, resultou em outro. Com o espaço reduzido dentro da calha, a bola agarrava dentro da calha em determinados pontos, necessitando retirar um pouco do papel utilizado, e após esta retirada do papel a bola continuou não fechando o contato entre as placas em determinados pontos, ficando claro que o formato da calha não poderia ser mais retangular. Montou-se então a calha em formato V , para sanar este imprevisto já que, desta forma a bola se movimenta pela calha sem agarrar em alguns pontos e sempre mantendo o contato entre as placas.

Outro forma de obtenção de uma resistência viável ao projeto foi a preparação de uma tinta condutiva a partir do pó de grafite e de esmalte base para unha transparente. Misturando os dois produtos consegue-se uma tinta que, após a aplicação no papelão para teste, mostrou-se eficaz. Com isso, aplicou-se essa tinta nas placas de alumínio para

aumentar a condução do material, e após a medição, percebeu-se que apesar da melhora, não foi suficiente para o uso no trabalho, além de não melhorar a medição de forma linear do material. Aplicou-se então esta tinta no papel cartão, moldado no mesmo comprimento da placa de alumínio, fazendo duas “*placas*” de papel com a tinta condutiva para a aplicação na calha. Por meio deste processo, conseguiu-se gerar as duas placas necessárias para a continuação do projeto.

Com estas placas, gerou-se os controladores de ângulo e posição mostrados neste trabalho, todavia, após a utilização destas placas, o material começou a mudar suas propriedades condutivas, sendo necessário a aplicação da tinta condutiva novamente para conseguir continuar o processo de realização do controle do projeto e, em seguida a aplicação, percebeu-se que o material não atendia mais o propósito do projeto visto que a variação do material não era mais de maneira linear, com isso, aplicou-se a tinta novamente em outro papel cartão só que, não conseguiu-se reproduzir os mesmos resultados satisfatórios da primeira aplicação, sendo necessário mudar novamente o material condutivo colocado na calha.

Utilizou-se então *eletrofitas*, uma fita auto-adesiva que é colocada sobre superfícies e é condutora elétrica no projeto (FITA CONDUTORA-ELETROFITAS, 2018). Ao analisar o material percebeu que sua propriedade resistiva era muito baixa, não servindo para a aplicação no projeto. Então começou-se a fazer uma resistência por meio do uso de arame envolto em papelão, produzindo várias voltas sobre o mesmo, também sem sucesso. Com isso lembrou-se da resistência de chuva e após testes considerou-se a aplicação deste material no projeto todavia, não conseguiu-se conectar duas resistências de chuva nas laterais da calha, porque o espaço entre os mesmos era demasiadamente pequeno e se aumentasse novamente a calha, além do risco de ocorrer o mesmo problema do atrito descrito anteriormente, faria com que a bola tivesse mais espaço do que o necessário para movimentar-se, e com isso, cairia das resistências. Pensando nisso, optou-se novamente pelo uso da placa de alumínio, porém, somente em um dos lados da calha, utilizando a resistência de chuva do outro lado.

Com um lado da calha com a placa de alumínio e o outro lado com a resistência de chuva, conseguiu-se um sistema que variasse a resistência o mais próximo da linearidade até então testada, e com valores que conseguiriam ser utilizadas no projeto, através do amplificador de instrumentação que daria o ganho do sinal. Contudo ao testar o projeto em movimento constatou-se que a bola não movimentava-se de maneira contínua sobre a resistência, a mesma dá saltos entre as voltas da resistência, fazendo com que a aplicação deste material já se tornasse inviável no projeto da maneira que estava. Para tentar solucionar este problema, utilizava-se sempre o último valor lido no programa. Porém, além desse empecilho, mesmo com o ganho do sinal, constatou-se que a variação apesar de ser quase linear, era muito próximo entre os valores obtidos e qualquer variação dos mesmos agregaria muito erro ao sistema, impossibilitando o controle da bola na calha de

maneira eficiente.

Analisando as opções testadas e a viabilidade de comprar e testar outro material visando a solução dos problemas acima relatados, notou-se que os materiais que poderiam ser a solução são demasiadamente caros e pouco viáveis para serem adquiridos, como por exemplo, placa de grafite, cujo preço é extremamente caro para a aplicação. Por isso, foram realizados até o controle de ângulo da calha de maneira eficiente, o controle de posição da bola junto com o anterior, realizado no trabalho não foram aplicados e testados de maneira conjunta, sendo que esta e outras estratégias para medir sua posição são deixados para trabalhos futuros.

5 Conclusão

Este trabalho é de grande importância porque possibilita posteriormente, realizar a implementação e controle de um pêndulo invertido, um sistema instável e não linear, propiciando análises de diferentes técnicas de controle.

No começo deste trabalho, foi proposto realizar o desenvolvimento de um pêndulo invertido agregando o problema “Ball and Beam”, da parte estrutural até o teste de diferentes métodos de controle deste protótipo. Contudo, a parte estrutural do trabalho se mostrou um problema logo no começo, pela dificuldade de encontrar as peças necessárias a montagem do primeiro protótipo.

Foram superados problemas como falta de engrenagem e / ou quebra da mesma por ter sido feita de acrílico, troca de motor, a aquisição da bola de metal, a falta de atrito entre o eixo e a lata, só resolvida depois da colocação uma “espuma” no eixo, o formato e tamanho da calha escolhida, entre outros. Foi possível realizar neste protótipo os métodos de controle por síntese direta e por LGR porém, não foi realizável a implementação e teste de um controle efetivo da posição na calha, impossibilitando a conclusão do “Ball and Beam”.

5.1 Trabalhos Futuros

A partir deste trabalho, é possível sugerir:

- Utilização de um material condutor inteiriço, como a placa de grafite para a obtenção correta dos valores aplicados a mesma;
- Montagem de uma nova estrutura para movimentação e localização da bola;
- Medir a posição da bola por meio de uma da câmera, usando visão;
- Usar um sensor de velocidade que diminui o ruído da planta e, com isso, consegue-se usar o controlador de velocidade.

ANEXO A – Código implementado no Matlab

Algorithm 1 discretização

```

1  clc; close all;
2
3  % num=4.006;
4  % den=[0.01 1];
5  % sys=tf(num ,den)
6  % figure;
7  % step(sys);
8
9  % Sintonia por sntese direta ou IMC
10
11 % td= 0.01;
12 % k=num;
13 % kp=(t/(k*td))%
14 % ki=(1/(k*td))
15
16 % Gc=tf([kp ki],[1 0 ])
17 % mf=feedback (Gc*sys,1)
18 % figure
19 % step(mf)
20
21 % Gcd = c2d(Gc,0.01,'tustin')% discretizando o controlador
    para jogar no arduino, com tempo de amostragem de 0.01s
22 %
23 % ts=(4/zetai*wn)
24 ts=5;% 1 segundo
25 zeta=0.707;
26 %
27 wn=(4/(ts*zeta))
28 %
29 kp0=2*zeta*wn
30 ki0=wn^2
31 %

```

Algorithm 2 discretização-continuação

```
1
2  Gcontrolador=tf([kp0 ki0],[1 0 ])
3  %
4  %  num=[kp ki];
5  %  den=[1 kp ki];
6  %  sys=tf(num ,den)
7  %
8  sys2=feedback(Gcontrolador*mf,1)
9  %  figure
10 %  step(sys2)
11 %
12 mfu2 = (1-sys2)*Gcontrolador
13 %
14 %  figure
15 %  step(mfu2)
16 %
17 Gcd2 = c2d(Gcontrolador,0.01,'tustin')% discretizando o
    controlador para jogar no arduino, com tempo de amostragem
    de 0.01s
```

ANEXO B – Código implementado no arduino Uno

Algorithm 3 Controle em Cascata

```

1 // Encoder connect to digitalpin 2 and 3 on the Arduino.
2
3 volatile unsigned int counter = 32000; //Aumenta ou diminui a
   variavel, dependendo do numero de pulsos
4 int counterant=0;
5 int contador = 0;
6 //motor B
7 int IN3 = 9 ;// codigo novo
8 int IN4 = 10 ;
9 int sentido=-1;
10 double erro=0;
11 double erroanti=0;
12 double u=0;
13 double r=0;// altera a velocidade
14 double P=0;
15 double angulolata=0;
16 double angulo=0;
17 double refang=((10*7)/180.0)*3.1415;
18 double erroantiang=0;
19 double tensao=0;
20
21 void setup() {
22   Serial.begin (9600);
23
24
25
26   pinMode(IN3,OUTPUT);//codigo novo
27   pinMode(IN4,OUTPUT);// codigo novo
28
29   pinMode(2, INPUT);           // set pin to input
30   pinMode(3, INPUT);           // set pin to input
31
32   digitalWrite(2, HIGH);       // turn on pullup resistors
33   digitalWrite(3, HIGH);       // turn on pullup resistors

```

Algorithm 4 Controle em Cascata-continuação

```
1
2 //Setting up interrupt
3 //A rising pulse from encodenren activated ai0().
4   AttachInterrupt 0 is DigitalPin nr 2 on moust Arduino.
5 attachInterrupt(0, ai0, RISING);
6
7 //B rising pulse from encodenren activated ai1().
8   AttachInterrupt 1 is DigitalPin nr 3 on moust Arduino.
9 attachInterrupt(1, ai1, RISING);
10
11 }
12
13 void loop() {
14   contador++;
15   if (contador == 500) {
16     refang = -refang;
17     contador = 0;
18   }
19
20
21
22   double d=counter-counterant;
23   if (d > 20000)
24     d-= 65536;
25   else if (d<-20000)
26     d+=65536;
27   double v = (100.0)*(2*3.1415/800.0)*d;// calculo da velocidade
28     do encoder
29   counterant=counter;
30
31   P= ((double) analogRead(A1))*(5.0/1023.0)-2.0;
32
33
34   angulo=((int)(counter-32000)/400.0)*3.1415;
35   angulolata=angulo/7;
36
37
38   double erroangulo= refang-angulo;
39
40
41
42
43   r=r+ 1.606 *erroangulo-1.594*erroantiang; // controlador PI
44     angulo
45   erroantiang=erroangulo;
```

Algorithm 5 Controle em Cascata-continuação

```
1
2     if (r > 40)
3     r=40;
4
5     if (r <-40)
6     r=-40;
7
8     erro= r -v;
9     u=u+ 0.3744*erro-0.1248*erroanti; // implementação do
10    controlador PI
11    erroanti=erro;
12    if (u>12)
13    u=12;
14    if(u<-12)
15    u=-12;
16
17    tensao=u;
18
19
20    if (tensao<0){
21    int PWM=map(-tensao,0,12,0,255); // PWM tem faixa de 0-255
22
23    analogWrite(IN3,PWM);
24    digitalWrite(IN4,LOW);
25    }
26    else{
27    int PWM=map(tensao,0,12,0,255);
28
29    digitalWrite(IN3,LOW);
30    analogWrite(IN4,PWM);
31    }
32
33    Serial.print(refang);
34    Serial.print ("□");
35    Serial.print(erroangulo);
36    Serial.print ("□");
37    Serial.print(u);
38    Serial.print ("□");
39    Serial.println(angulo);
40
41
42    delay(10);
43
44 }
```

Algorithm 6 Controle em Cascata-continuação

```
1   void ai0() {
2   // ai0 is activated if DigitalPin nr 2 is going from LOW to
   HIGH
3   // Check pin 3 to determine the direction
4   if(digitalRead(3)==LOW) {
5   counter++;
6   }
7
8
9
10
11  else{
12  counter--;
13  }
14  }
15
16  void ai1() {
17  // ai0 is activated if DigitalPin nr 3 is going from LOW to
   HIGH
18  // Check with pin 2 to determine the direction
19  if(digitalRead(2)==LOW) {
20  counter--;
21  }else{
22  counter++;
23  }
24  }
```

Algorithm 7 Controlador angulo-LGR

```

1
2
3
4
5 // Encoder connect to digitalpin 2 and 3 on the Arduino.
6
7 volatile unsigned int counter = 32000; //Aumenta ou diminui a
   variavel, dependendo do numero de pulsos
8 int counterant=0;
9 int contador = 0;
10 //motor B
11 int IN3 = 6 ;// codigo novo
12 int IN4 = 7 ;
13 int sentido=-1;
14 double erro=0;
15 double erroanti=0;
16 double u=0;
17 double r=0;// altera a velocidade
18 double P=0;
19 double angulolata=0;
20 double angulo=0;
21 double refang=((10*7)/180.0)*3.1415;
22 double erroantiang=0;
23 double tensao=0;
24 double erroantiang2=0;
25 void setup() {
26 Serial.begin (9600);
27
28
29
30 pinMode(IN3,OUTPUT);//codigo novo
31 pinMode(IN4,OUTPUT);// codigo novo
32
33 pinMode(2, INPUT);           // set pin to input
34 pinMode(3, INPUT);           // set pin to input
35
36 digitalWrite(2, HIGH);       // turn on pullup resistors
37 digitalWrite(3, HIGH);       // turn on pullup resistors
38
39 //Setting up interrupt
40 //A rising pulse from encodenren activated ai0().
   AttachInterrupt 0 is DigitalPin nr 2 on moust Arduino.
41 attachInterrupt(0, ai0, RISING);
42
43 //B rising pulse from encodenren activated ai1().
   AttachInterrupt 1 is DigitalPin nr 3 on moust Arduino.
44 attachInterrupt(1, ai1, RISING);
45
46 }

```

Algorithm 8 Controlador angulo-LGR-continuação

```
1
2
3
4   void loop() {
5   contador++;
6   if (contador == 300) {
7   refang = -refang;
8   contador = 0;
9   }
10
11  double d=counter-counterant;
12  if (d > 20000)
13  d-= 65536;
14  else if (d<-20000)
15  d+=65536;
16  double v = (100.0)*(2*3.1415/800.0)*d;// calculo da velocidade
17  counterant=counter;
18
19
20  P= ((double)analogRead(A1))*(5.0/1023.0)-2.0;
21
22
23  angulo=((int)(counter-32000)/400.0)*3.1415;
24  angulolata=angulo/7;
25
26
27  double erroangulo= refang-angulo;
28
29  u=u+11.039*erroangulo- 13.6806*erroantiang
30    +3.26752*erroantiang2; // controlador PID  ângulo
31  if(abs(erroangulo)<0.05){
32  erroangulo=0;
33  u=0;
34  }
35
36  erroantiang2=erroantiang;
37  erroantiang=erroangulo;
38  if (u>12)
39  u=12;
40  if(u<-12)
41  u=-12;
42  tensao=u;
```

Algorithm 9 Controlador angulo-LGR-continuação

```
1
2  if (tensao<0){
3  int PWM=map(-tensao,0,12,0,255); // PWM tem faixa de 0-255
4
5  analogWrite(IN3,PWM);
6  digitalWrite(IN4,LOW);
7  }
8  else{
9  int PWM=map(tensao,0,12,0,255);
10
11  digitalWrite(IN3,LOW);
12  analogWrite(IN4,PWM);
13  }
14
15  Serial.print(refang);
16  Serial.print ("␣");
17  Serial.print(erroangulo);
18  Serial.print ("␣");
19  Serial.print(u);
20  Serial.print("␣");
21  Serial.print(v);
22  Serial.print ("␣");
23  Serial.print(angulo);
24  Serial.print ("␣");
25  Serial.println(counter);
26
27  delay(10);
28  }
29
30  void ai0() {
31  // ai0 is activated if DigitalPin nr 2 is going from LOW to
32  // HIGH
33  // Check pin 3 to determine the direction
34  if(digitalRead(3)==LOW) {
35  counter++;
36  }
37  else{
38  counter--;
39  }
40
41  void ai1() {
42  // ai0 is activated if DigitalPin nr 3 is going from LOW to
43  // HIGH
44  // Check with pin 2 to determine the direction
45  if(digitalRead(2)==LOW) {
46  counter--;
47  }else{
48  counter++;
49  }
}
```

Referências

- ALBARELLO, J. da R.; PEREIRA, K.; FAORO, V. A digital speed filter for motion control drives with a low resolution position encoder. *Vivências:Revista eletrônica de Extensão da URI*, v. 9, 2013. 1
- ARDUINOCC. 2017. Disponível em: <<https://store.arduino.cc/usa/arduino-uno-rev3>>. Acesso em: 26/08/2017. 9
- BALL AND BEAN. 2017. Disponível em: <http://www.quanser.com/Products/ball_beam>. Acesso em: 08/08/2017. 17
- BELLINI, A.; BIFARETTI, S.; COSTANTINI, S. A digital speed filter for motion control drives with a low resolution position encoder. *Automatica*, v. 44, p. 67–74, 2003. 8
- CAMPOS, M. C. M. M. de; TEIXEIRA, H. C. G. *Controles típicos de equipamentos e processos industriais*. [S.l.]: Editora Edgard Blucher, 1ª edição., 2006. 15, 16
- DATASHEET INA122-TEXAS INSTRUMENTS. 2018. Disponível em: <<http://www.ti.com/lit/ds/symlink/ina122.pdf>>. Acesso em: 25/06/2018. 14
- DORF, R. C.; BISHOP, R. H. *Sistemas de controle modernos*, 2001. 8ª, 2001. 17
- FITA CONDUTORA-ELETROFITAS. 2018. Disponível em: <<http://www.eletrofitas.com.br/>>. Acesso em: 29/06/2018. 31
- FITZGERALD, A.; ELÉTRICAS, M. 6ª Edição. [S.l.]: Editora Bookman, 2006. 4
- FRANKLIN, G. F.; POWELL, J. D.; WORKMAN, M. L. *Digital control of dynamic systems*. [S.l.]: Addison-wesley Menlo Park, CA, 1998. v. 3. 17, 21
- G.A.IRVING; M.W.ANDRÉS. “DISEÑO CONCEPTUAL PARA LA CAJA REDUCTORA DE UN ROTOR DE COLA DE UN HELCIOPTERO BIPLAZA”. 12 p. Tese (Doutorado) — Instituto politécnico nacional, 2004. 6
- KUO, B. C. *Digital control systems*. [S.l.]: New York Oxford University Press, 1992. 1
- LEITE, F. et al. *ESTUDO DO FUNCIONAMENTO DE UMA CAIXA DE TRANSMISSÃO E SEUS COMPONENTES* Fernanda Leite, José Carlos Teles, Douglas Ferreira, Israel A. M. Lima e Carla T. M. Anflor. 2012. 5
- NAWAWI, S. W.; AHMAD, M. N.; OSMAN, J. H. S. Real-Time Control of a Two-Wheeled Inverted Pendulum Mobile Robot. *Engineering and Technology*, v. 29, n. 1, p. 214–220, 2008. 1
- OGATA, K. *Engenharia de Controle Moderno*. [S.l.]: 5ª edição, Pearson Prentice Hall,São Paulo, 2010. 1, 18, 20
- UNDERSTANDING QUADRATURE ENCODED SIGNALS. 2017. Disponível em: <https://www.pjrc.com/teensy/td_libs_Encoder.html>. Acesso em: 08/08/2017. 9
- VIRSEDA, M. *Modeling and Control of the Ball and Beam Process* . 2004. 52 p. 16

WAI, R. J.; KUO, M. A.; LEE, J. D. Design of Cascade Adaptive Fuzzy Sliding Mode Control for Nonlinear Two axis Inverted Pendulum Servomechanism. *IEEE Transactions on Fuzzy Systems*, v. 16, 2008. 1