

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE MECÂNICA
CURSO DE ENGENHARIA INDUSTRIAL MECÂNICA

FABRIZIO ALMEIDA DE CARVALHO

**CONSTRUÇÃO E IMPLEMENTAÇÃO DE UM SISTEMA DE
CONTROLE EM UM *BALL AND BEAM* MODIFICADO**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO

2013

FABRIZZIO ALMEIDA DE CARVALHO

**CONSTRUÇÃO E IMPLEMENTAÇÃO DE UM SISTEMA DE
CONTROLE EM UM *BALL AND BEAM* MODIFICADO**

Trabalho de Conclusão de Curso de graduação apresentado à Coordenação de Mecânica da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do título de Engenheiro Industrial Mecânico.

Orientador: Prof. Dr. Marcio Aurelio Furtado Montezuma

CORNÉLIO PROCÓPIO

2013

FOLHA DE APROVAÇÃO

FABRIZZIO ALMEIDA DE CARVALHO

CONSTRUÇÃO E IMPLEMENTAÇÃO DE UM SISTEMA DE CONTROLE EM UM *BALL AND BEAM* MODIFICADO

Trabalho de Conclusão de Curso de graduação apresentado como requisito parcial para a obtenção do título de Engenheiro Industrial Mecânico a Universidade Tecnológica Federal do Paraná – Campus Cornélio Procópio.

Prof. Dr. Marcio Aurélio Furtado Montezuma
Professor Orientador
UTFPR/ Campus Cornélio Procópio

Prof. Ms. Conrado Di Raimo
UTFPR/ Campus Cornélio Procópio

Prof. Esp. Marcio Sadao Hirata
UTFPR/ Campus Cornélio Procópio

Cornélio Procópio, 07 de agosto de 2013

Dedico este trabalho a minha família que sempre me apoiou e me incentivou para que eu concluísse o curso de Engenharia Industrial Mecânica.

AGRADECIMENTOS

Ao Professor Doutor Marcio Aurelio Furtado Montezuma, que primeiro aceitou me orientar, e escolheu o tema a ser abordado, além de ajudar sempre que possível e necessário, fornecendo seus conhecimentos e proporcionando que eu aprendesse cada vez mais.

Aos meus amigos de Cornélio Procópio, em especial ao pessoal do laboratório: Bruno Shimada, Carlos Matheus, Lucas niro e Vitor Takahashi, que ajudaram nas dificuldades e duvidas na construção do mecanismo.

A todos os professores que contribuíram e colaboraram para a realização desta graduação com excelência e qualidade.

Aos amigos e amigas de Ourinhos que estiveram ao meu lado nesses cinco anos de graduação, nas horas boas e ruins.

"Você pode encarar um erro como uma besteira a ser esquecida, ou como um resultado que aponta uma nova direção".
(Steve Jobs)

RESUMO

CARVALHO, Fabrizzio Almeida. **Construção e implementação de um sistema de controle em um *ball and beam* modificado**. 2013. 86 p. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Industrial Mecânica) – Universidade Tecnológica Federal do Paraná – Unidade Cornélio Procópio. Cornélio Procópio, 2013.

A automação está mais presente nas indústrias, na produção de produtos com tolerâncias menores, com altíssima qualidade demandada pelo consumidor, além disso, devido à alta concorrência do mercado atual a produção dos mesmos tem ocorrido em grande escala e em tempo cada vez menor. Dentro de casa a automação também tem se tornado cada vez mais comum: a procura de produtos que sejam quase autossuficientes e que substituam o homem em trabalhos domésticos e que facilitem a sua vida, é consequência de um mundo em que não há tempo para se desperdiçar com tarefas simples. Ligados diretamente a automação encontram-se os sistemas de controles, os quais são parte fundamental para algum mecanismo desempenhar seu controle automático. É neste ponto que entra o trabalho conclusão de curso em questão, que trata da construção de um mecanismo no qual é possível aplicar alguns métodos de controle para poder, então, estudá-los. No caso, o mecanismo escolhido foi um clássico que é o *Ball and Beam*, um dos mais utilizados para realizar este tipo de estudo. Cabe ressaltar que em sua construção foram utilizados materiais de fácil acesso e alguns deles até mesmo considerados sucata. Além da construção, foram também aplicados dois métodos de controle: um clássico através de aplicação em planta e outro através de simulação em software.

Palavras - Chave: *Ball and Beam*. Sistema de controle.PID. Comunicação *Wireless*.

ABSTRACT

CARVALHO, Fabrizzio Almeida. **Construction and implementation of a control system in a modified *ball and beam***. 2013. 86 p. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Industrial Mecânica) – Universidade Tecnológica Federal do Paraná – Unidade Cornélio Procópio. Cornélio Procópio, 2013.

The automation shows up more in the industry, in the production of less geometric tolerance products and with a very high quality demanded by the consumer, furthermore, due to the competition of the current market, the production has been made in large scale and lower time. Indoors the automation has become common too, the demand for products which are almost self-sufficient and replaces the man in housework making his life easier, is consequence of a world where there is no time to waste with simple tasks. Connected directly with automation are the control systems, which are a fundamental part to some mechanism to make its automatic control. That is where fits this final paper, it talks about the construction of a mechanism in which is possible to apply some control methods for then, studying them. In case, the chosen mechanism was the classic Ball And Beam, one of the most used for this type of study. Remembering that in its construction were used materials that are easy to find and some of them even are considered waste. In addition the construction, there were also applied two methods of control: a classic one, through a plant application, and another one through a software simulation.

Keywords: *Ball and Beam*. Control System. PID. Wireless Communication.

LISTA DE FIGURAS

Figura 1 - Mecanismo <i>Ball and Beam</i> da empresa Quanser	16
Figura 2 - Representação esquemática do mecanismo <i>Ball and Beam</i> da Quanser	17
Figura 3 - Esquemático <i>Ball and Beam</i>	17
Figura 4 - Mecanismo <i>Ball and Beam</i> construído	18
Figura 5 - Modelo Físico.....	19
Figura 6 - Funcionamento do sensor de posição por fio resistivo	20
Figura 7 - Mecanismo com seus materiais.....	20
Figura 8 - Vista de trás do mecanismo <i>Ball and Beam</i>	21
Figura 9 - Vista frontal do Mecanismo feito em Solidworks.....	22
Figura 10 - Vista isométrica do Mecanismo feito em Solidworks.....	23
Figura 11 - Vista isométrica do carrinho feito em Solidworks	23
Figura 12 - Balanceador de hélice de helicóptero	24
Figura 13 - Trilha Ferromodelo.....	25
Figura 14 - Carrinho Obtido de Ferromodelo.....	25
Figura 15 - Calha de alumínio	25
Figura 16 - Modelo Físico da estrutura do <i>Ball and Beam</i>	26
Figura 17 - Modelo Físico da estrutura do <i>Ball and Beam</i>	26
Figura 18 - Modelo Físico do carrinho do <i>Ball and Beam</i>	27
Figura 19 - Modelo Físico da estrutura e carrinho do <i>Ball and Beam</i>	27
Figura 20 - Modelo Físico da estrutura e carrinho do <i>Ball and Beam</i>	27
Figura 21 - Funcionamento do mecanismo <i>Ball and Beam</i>	28
Figura 22 - Princípio de funcionamento de um encoder rotativo	30
Figura 23 - Princípio de funcionamento de um encoder linear	30
Figura 24 - Representação gráfica do funcionamento de um encoder.....	31
Figura 25 - Representação gráfica do funcionamento de um encoder em relação a sentido positivo e negativo.	31
Figura 26- Diagrama de Pinos dsPIC30F2010.....	32
Figura 27 - <i>Wireless Serial RS232 Port Adapter Module Freescale</i>	33
Figura 28 - Bateria 7,6 Volts.....	34
Figura 29 - Placa dsPIC30F2010	34
Figura 30 - Representação esquemática do circuito feito no <i>software</i> Eagle PCB....	35
Figura 31 - Desenho esquemático do <i>encoder</i> da família H97X0	36
Figura 32 - Ligação Lm7805.....	37
Figura 33 - LM7805	37
Figura 34-Layout da placa feita em <i>software</i> Eagle PCB	38
Figura 35 -Layout da placa feita em <i>software</i> Eagle PCB	39
Figura 36- Placa já com impressa e corroída parte superior	39
Figura 37- Placa já com impressa e corroída parte inferior.....	40
Figura 38 - Placa corroída e com todos os componentes soldados	41

Figura 39 - Imagem ilustrativa de uma placa de aquisição da NI série M	42
Figura 40 - Pinagem PCI 6251	43
Figura 41 - Sinal Analógico x Sinal digital	44
Figura 42 - Esquemático de um circuito com uma carga e uma chave	45
Figura 43 - Representação do sinal PWM de acordo com o <i>Duty cycle</i>	46
Figura 44 - Funcionamento de um servo com Pwm RC	47
Figura 45 - Servomotor	48
Figura 46 - Ligação Receptor wireless x computador	49
Figura 47 - Foto da Ligação Receptor wireless x computador	49
Figura 48 - Ligação cabo serial com porta serial do computador	50
Figura 49 - Ligação NI PCI 6251 x dsPIC30F2010	51
Figura 50 - Servomotor x dsPIC30F2010.....	52
Figura 51 - Ligação completa do mecanismo.....	52
Figura 52 - Foto da ligação completa do mecanismo.....	53
Figura 53 - Funcionamento de um controle de malha aberta	56
Figura 54 - Funcionamento de um controle em malha fechada	56
Figura 55 - Controle de malha fechada com PID	57
Figura 56 - Sistema de controle seguidor.....	61
Figura 57 - Diagrama de blocos (visão geral).....	63
Figura 58 - Diagrama de blocos dividida em dois (parte esquerda)	64
Figura 59 - Diagrama de blocos dividida em dois (parte direita)	65
Figura 60 - Painel frontal (visão geral)	66
Figura 61 - Painel Frontal (Aquisição de dados)	67
Figura 62 - Painel frontal(PID).....	67
Figura 63 - Painel Frontal (<i>Setpoint</i> , Posição, <i>reset</i> do zero, gravação de dados)....	68
Figura 64 - Painel frontal(Gráfico de posição e ação de controle).....	68
Figura 65 - Parâmetros encontrados do PID	69
Figura 66 - Modelo Simplificado feito no <i>software</i> Adams.....	71
Figura 67 - Modelo simplificado feito em <i>software</i> Adams	72

SUMÁRIO

1 INTRODUÇÃO	13
1.1 OBJETIVOS	13
1.1.1 Objetivo Geral	13
1.1.2 Objetivos Específicos	13
1.2 REVISÃO BIBLIOGRÁFICA	15
2 CONSTRUÇÃO MECÂNICA DO <i>BALL AND BEAM</i> MODIFICADO	22
2.1 MATERIAIS UTILIZADOS:	24
3 CONSTRUÇÃO ELÉTRICA E COMPUTACIONAL DO MECANISMO <i>BALL AND BEAM</i> MODIFICADO	28
3.1 INTRODUÇÃO	28
3.2 PLACA DO CARRINHO	28
3.2.1 Encoder	29
3.2.2 Microcontrolador dsPIC30F2010	32
3.2.3 Eagle PCB	32
3.2.4 Lista de Materiais.....	33
3.2.5 Esquemático.....	344
3.2.5.1 Encoder x dsPIC30F2010.....	345
3.2.5.2 Wireless x dsPIC30F2010.....	346
3.2.5.3 Bateria x LM7805 x dsPIC30F2010.....	346
3.2.5.4 Programador	347
3.2.6 Outros.....	41
3.2.6.1 Labview	41
3.2.6.2 Placa de Aquisição	412
3.2.6.3 Sinal analógico x Sinal digital	413
3.2.6.4 Pwm RC	414
3.2.6.5 Servomotor.....	417
3.2.6.6 Wireless 2 x Computador	418
3.2.6.7 Placa de aquisição x Microcontrolador dsPIC30F2010	41
3.2.6.8 Servo motor x dsPIC30F2010	51
4 SINCRONIZAÇÃO COM COMPUTADOR E APLICAÇÃO DE SISTEMA DE CONTROLE	54
4.1 INTRODUÇÃO	54

4.1.1 Software Matlab/Simulink	54
4.1.2 Software Adams	54
4.1.3 Sistema de controle	55
4.1.4 PID (Proporcional-Integral-Derivativo)	56
4.1.5 Sistema Seguidor	60
4.2 APLICAÇÃO DO MÉTODO DE CONTROLE PID	63
4.3 MODELAGEM ADAMS E SIMULAÇÃO SISTEMA SEGUIDOR	71
5 CONCLUSÃO	74
REFERENCIAS.....	75
ANEXOS	79
ANEXO A - Programa rs232 dspic	79
ANEXO B - Programa que geral sinal pwm rc.....	7981
ANEXO C - Sistema seguidor matlab.....	83
ANEXO D - Matrizes de estado.....	86

1 INTRODUÇÃO

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Primeiro construir um mecanismo *Ball and Beam* modificado com peças de baixo custo e até mesmo sucatas. E posteriormente aplicar um método de controle no mesmo.

1.1.2 Objetivos Específicos

- Construir a parte mecânica e eletrônica de um mecanismo *Ball and Beam modificado*;
- Modelar o mecanismo em software de análise Dinâmica para facilitar posteriormente a aplicação do sistema de controle escolhido;
- Definir o sistema de controle a ser adotado;
- Escolher um programa para realizar a aquisição de dados e aplicação do sistema de controle;
- Comunicar o mecanismo com um computador;
- Criar o sistema de controle no software escolhido para fazer a aquisição de dados e o controle do mecanismo através da técnica *hardware-in-the-loop*.

Este trabalho transcorreu da seguinte forma: primeiro foi realizado um estudo sobre os projetos similares existentes, então iniciou-se a fase de projeto do mecanismo, dando início a parte mecânica e por conseguinte a parte elétrica. Para melhor verificação do projeto mecânico, o mesmo foi desenhado em *Software Solidworks* para ter uma ideia como ficaria aproximadamente depois de construído.

Terminada a parte de estudo e projeto do mecanismo, deu-se início a parte de construção, tanto elétrica como mecânica, e, por conseguinte, foi dado prosseguimento a partir da comunicação e sincronização com o computador utilizando *software Labview*. Com o mecanismo sincronizando e se comunicando com o computador, foi então realizada a etapa de aplicação de um método de

controle. O primeiro aplicado foi o PID utilizando o *software Labview* e o segundo, para realizar uma comparação, foi uma simulação de controle moderno que usa sistema seguidor, utilizando *software* de simulação dinâmica *Adams* e posteriormente *Matlab/Simulink*.

Após serem aplicados os dois métodos de controle, tanto por simulação de um deles e real aplicação do outro, os resultados foram comparados e então o projeto se deu como finalizado. Cabe ressaltar que todas etapas serão explicadas com maiores detalhes posteriormente.

O *Ball and Beam* é um projeto do ramo da engenharia muito conhecido, pois é um modelo onde se pode aplicar a maioria dos métodos de controle (ex.: PID, Fuzzy). Existe até um modelo feito pela empresa *Quanser*, a qual é muito conhecida pelos seus mecanismos para uso didático. Ele é composto por uma haste que pode ser inclinada por um servomotor. Sobre tal haste encontra-se uma esfera, que pode rolar tanto para frente quanto para trás. O intuito do mecanismo é fazer com que a esfera se desloque até a posição previamente programada, através da rotação do servo motor e conseqüentemente gerando a inclinação da haste. Os sistemas de controle, que podem ser vários, comandam tal deslocamento e posicionamento da esfera automaticamente.

No século XVIII, James Watt construiu um controlador centrífugo que tinha como objetivo controlar a velocidade de uma máquina a vapor, tal construção foi o primeiro controle automático significativo na história. Em 1922, Minorsk trabalhou em controladores automáticos para pilotar navios e mostrou como poderia controlar a estabilidade do mesmo através de equações diferenciais.

Logo depois em 1940, graças ao método de resposta de frequência foi possível projetar sistemas de controle em malha fechada satisfazendo requisitos de desempenho. Durante o final da década de 40 e o início da de 50 graças a Evan foi desenvolvido o método do lugar das raízes. Usando como princípio esses dois métodos que constituem o núcleo da teoria de controle clássica (método de resposta de frequência e o método do lugar das raízes) foi possível realizar sistemas estáveis e que satisfazem um conjunto de requisitos de desempenho mais ou menos arbitrários.

Porém a teoria de controle clássica serve para sistemas que possuem uma entrada e uma saída, então surge a teoria de controle moderno que abrange sistemas com varias entradas e saídas, mais complexos e que envolvem maior

número de equações. Tal teoria obteve melhoras a partir de 1960, mais ou menos, com o surgimento dos computadores digitais, os quais possibilitaram a análise e domínio do tempo e de sistemas complexos, dando origem a moderna teoria de controle baseada nas técnicas de análise e síntese através de variáveis de estado. Tal teoria foi criada com a função de lidar com a maior complexidade dos problemas modernos e atender às rigorosas exigências quanto a peso, exatidão e custos de projetos relativos a aplicações militares, espaciais e industriais.

Com a diminuição do preço e do tamanho dos computadores digitais o controle moderno tem cada vez mais sendo aplicado em todos os ramos, não só nas áreas de engenharia, mais também na medicina, biomedicina, biologia, entre outras áreas. Pode-se então observar que cada vez mais o controle automático tem se tornado vital em nossas vidas, começando primeiro com sua importância para projetos com uma avançada tecnologia como é o caso de estações espaciais, controle de guiamento de mísseis entre outros, e passando para o ambiente industrial onde tem sido usado como, por exemplo, em comandos numéricos para máquina-ferramenta, projeto de sistemas de pilotagem automática da indústria aeroespacial e no projeto de automóveis e caminhões da indústria automobilística, também em operações industrial como controle de vazão, temperatura, pressão entre outros.

Tal controle veio para indústria para diminuir possíveis erros causados por operários, além de aumentar a produtividade, a precisão e diminuição de trabalhos repetitivos manuais antes realizados por operários.

Além do ambiente industrial pode ser também encontrado hoje em dia, cada vez mais, presente no ambiente doméstico para poder facilitar a vida das pessoas e aumentar seu conforto, como por exemplo: controle de iluminação através de sensor de presença, microondas, máquina de lavar, entre outros equipamentos.

1.2 REVISÃO BIBLIOGRÁFICA

O mecanismo *Ball and Beam* tem sido utilizado já há algum tempo por ser uma maneira mais simplificada de entender alguns conceitos de controle, além do que aprender a lidar com as dificuldades que podem ser encontradas, as quais são semelhantes nas indústrias quando se quer controlar algum processo ou parâmetro.

Por já ser aplicado há algum tempo, podemos encontrar na literatura diversos modelos que foram feitos, com diversas formas de construção, controle, aquisição de dados, entre outros, é um mecanismo que permite uma ampla variabilidade dependendo da criatividade do projetista. A seguir será apresentado alguns dos projetos feitos com algumas peculiaridade de cada um.

O primeiro exemplo citado é um dos mais encontrados, é o que a empresa *Quanser* desenvolveu, trata-se de um mecanismo didático que é vendido comercialmente, com instruções e módulos, para facilitar sua montagem, é de fácil manuseio, e praticamente só é preciso montar as partes, a estrutura, posicionar o servo, sensores e demais peças, e então aplicar o método de controle desejado, normalmente é aplicado o PID e é utilizado o software *Matlab/Simulink*. O bom deste modelo além desta facilidade é que ele é robusto, outro fator também é que como é bastante usado, é fácil encontrar exemplos para comparar e tomar como base para montar seu próprio. Estruturalmente falando ele é feito com um servomotor ligado a um conjunto de engrenagens em uma junta que é acoplada a uma haste e na outra ponta da haste existe uma espécie de mancal para a mesma poder movimentar-se para cima e para baixo, para melhor exemplificar o este exemplo segue as figura 1 e 2. A desvantagem dele é o fato de seu valor para aquisição ser um pouco elevado, e como ele já tem sua estrutura pronta não possui muitas mudanças.

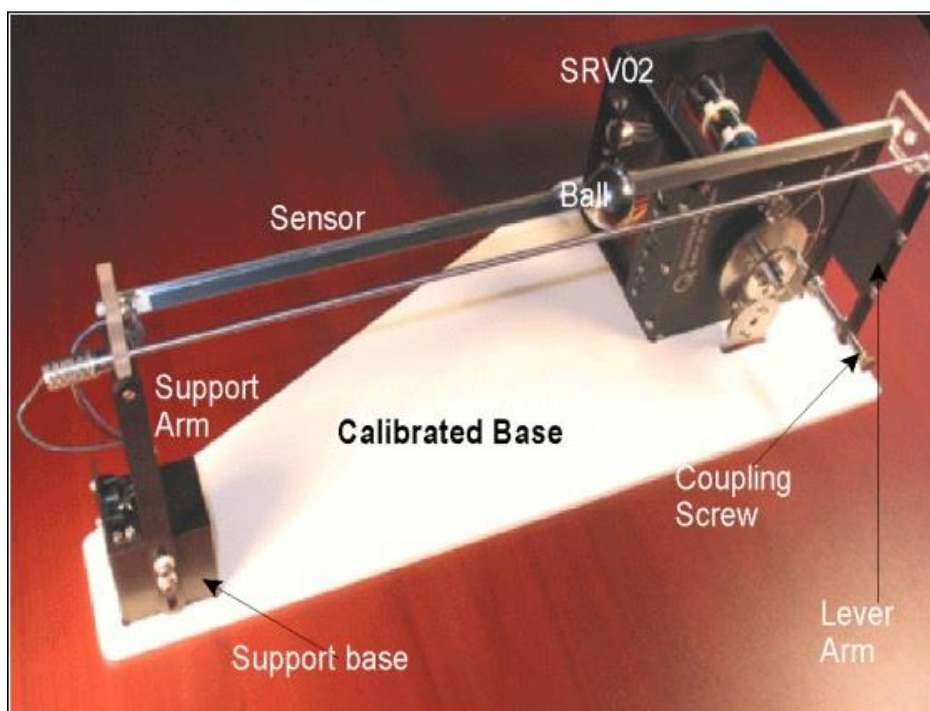


Figura 1 – Mecanismo *Ball and Beam* da empresa Quanser
Fonte: Quanser (2013, P. 01)

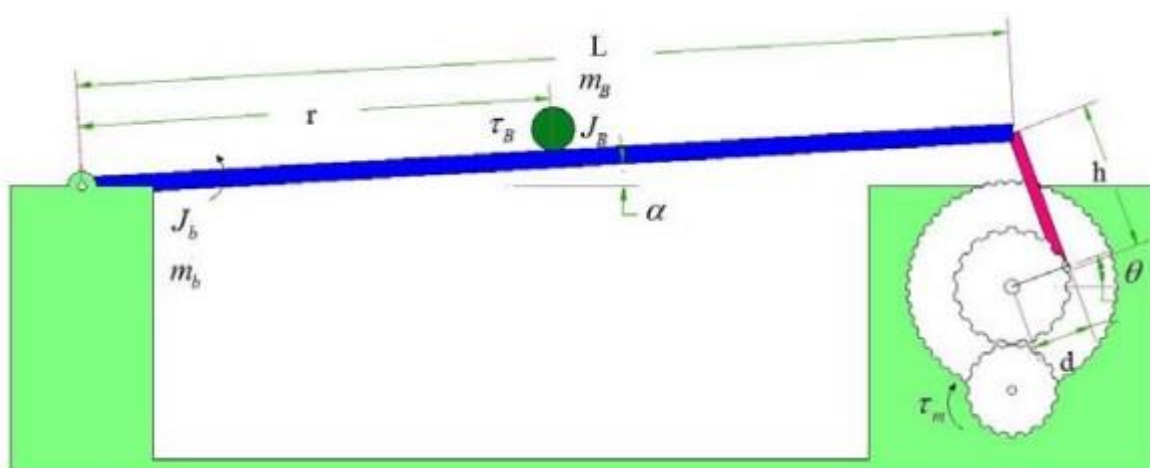


Figura 2 – Representação esquemática do mecanismo Ball and Beam da Quanser
Fonte: Jahromi (2012,P.17)

Choi, Dorsett e Evanko (2013) fizeram um modelo o qual a haste é de alumínio, é utilizado como sensor de posição para a bolinha dois sensores infra vermelho, a bolinha é de ping pong e tem como objetivo apenas estabilizar a bolinha no centro e não em qualquer posição pré definidas como outros projetos, para entender melhor como foi feito, segue abaixo a figura 3 deste modelo.

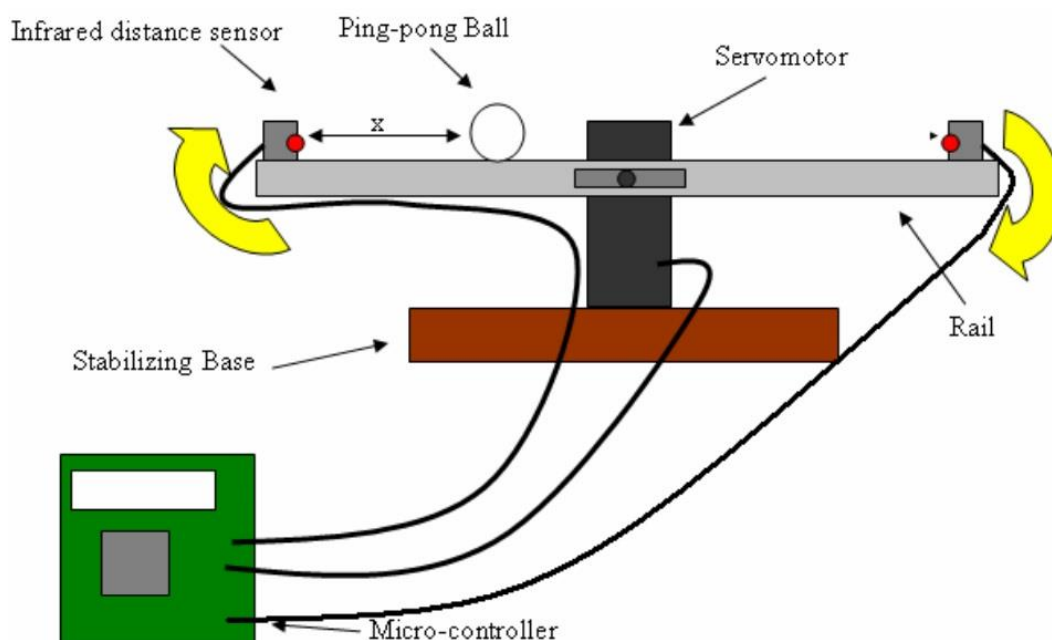


Figura 3 - Esquemático *Ball and Beam*
Fonte: Choi, Dorsett, Evanko (2013, P. 02)

Trata-se de um projeto interessante por utilizar materiais de baixo custo e de fácil acesso, assim sendo mais fácil sua construção. Porém tem algumas desvantagens, como, por exemplo, o sensor infra vermelho não é tão preciso, tornando assim o controle não tão eficaz, outro fator também é que como foi dito acima, ele só posiciona no centro, então não tem como o usuário escolher setar uma posição, ele pode apenas deslocar a bolinha e então ela retornará para o centro. Abaixo na figura 4 pode ser verificado como foi montado, não apenas o modelo como na figura 3 anterior.

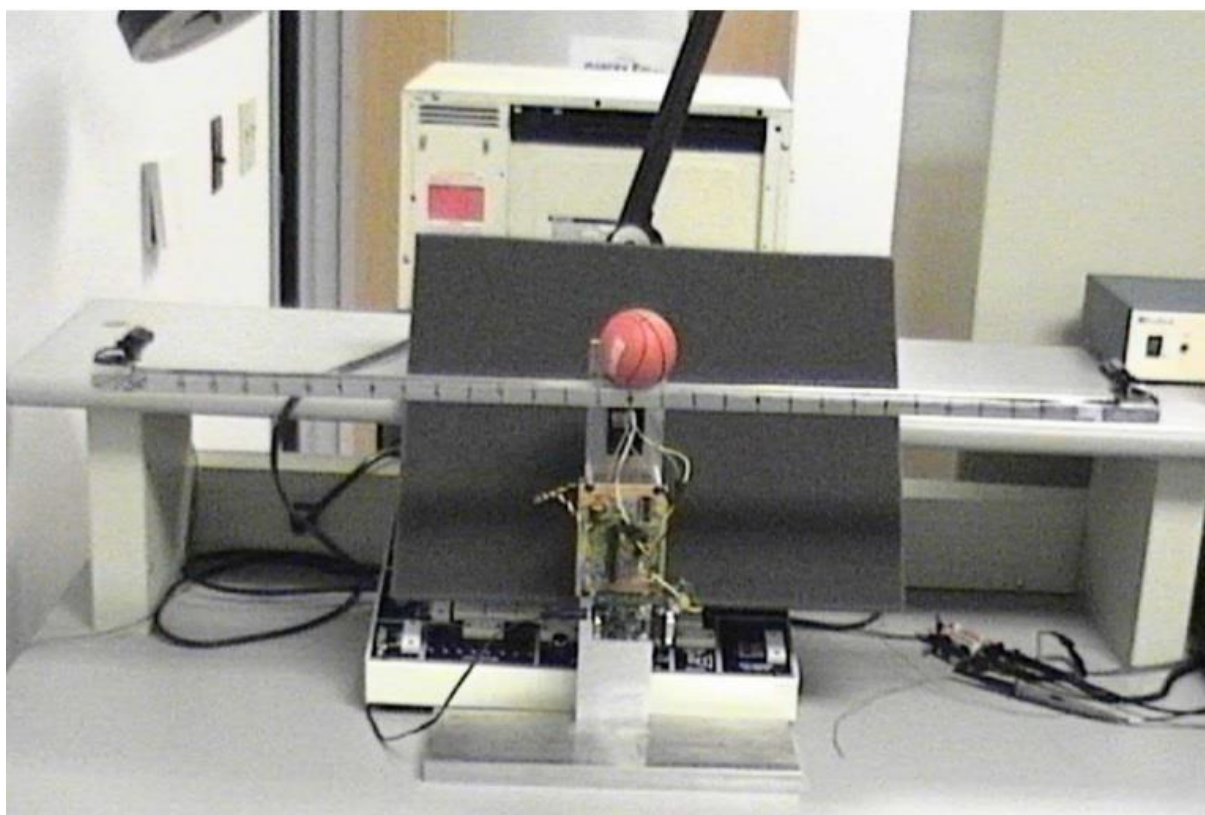


Figura 4 - Mecanismo *Ball and Beam* construído
Fonte: Choi, Dorsett, Evanko (2013, P. 06)

Diniz e Schvarcz (2010) construíram um modelo que o princípio é o mesmo do anterior, que é manter a bolinha no centro, porém o que pode-se notar de curioso neste projeto é que ele fez o modelo no *software Adams* e através dele foi gerado equações do processo de controle as quais foram exportada para o *matlab* e foram solucionadas, fazendo assim uma simulação do controle. Também foi feito um modelo matemático e aplicado um controle prático, para comparar assim os dados obtidos tanto computacionalmente, quanto matematicamente (tal comparação pode

ser observada no gráfico 1). O que difere também é que ele utiliza como sensor ultrassom, o qual também não tem uma precisão muito boa, e também a maioria das partes mecânicas são feitas de madeira, o que pode ser um problema pois é um material que não tem tanta resistência quanto os metais como em outros projetos, e também sua durabilidade é menor, o projeto feito pode ser visto na figura 5 abaixo.

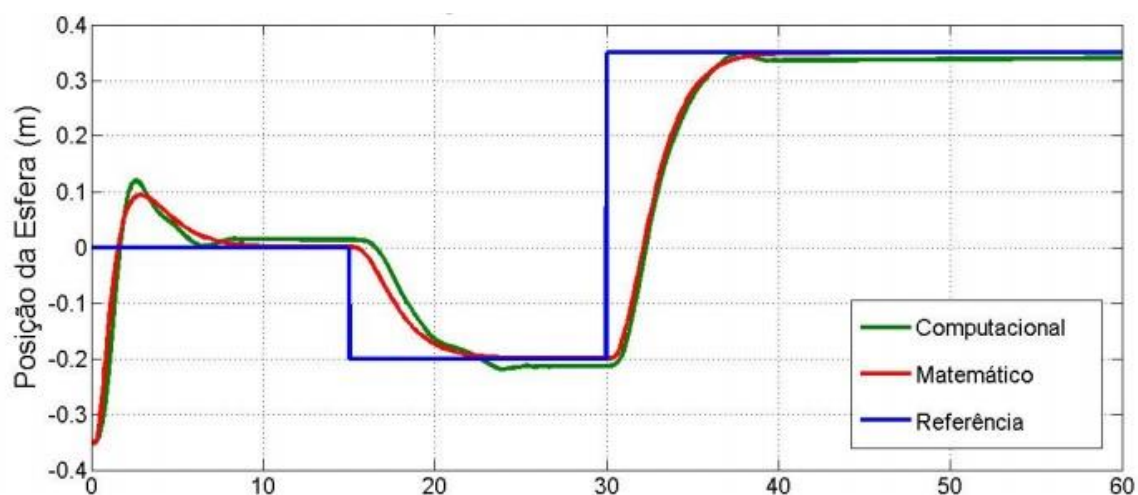


Gráfico 1 - Posição da esfera
Fonte: Diniz,Schvarcz (2010, P.06)



Figura 5 - Modelo Físico
Fonte: Diniz,Schvarcz (2010, P. 06)

Wang (2007) criou um mecanismo com uma haste de alumínio revestida por dentro por policarbonato e uma bola de aço, a qual rola sobre um fio resistivo, este é um dos pontos curiosos desse projeto, a posição da bolinha é dada através da variação da tensão que é gerada pela bolinha ao rolar no fio resistivo (tal funcionamento pode ser melhor compreendido na figura 6), outro ponto também é que ele procurou materiais resistentes e com boa duração, como aço, alumínio, policarbonato entre outros, pode-se analisar a construção mecânica deste projeto na figura 7.

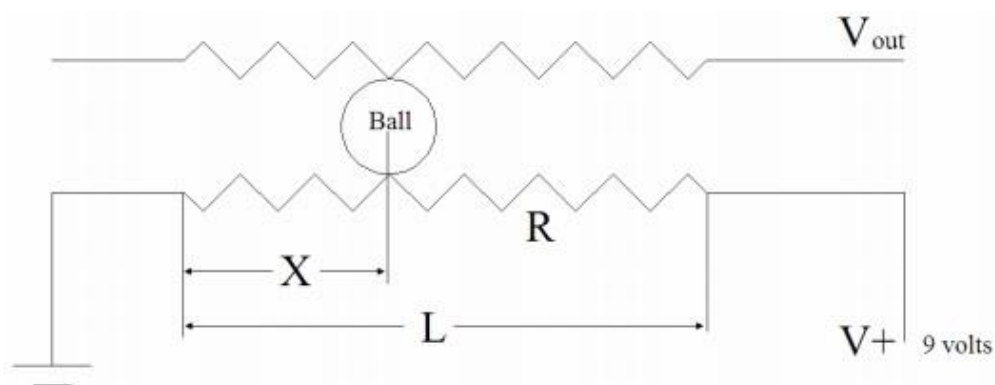


Figura 6 - Funcionamento do sensor de posição por fio resistivo
 Fonte: Wang (2007, P.19)

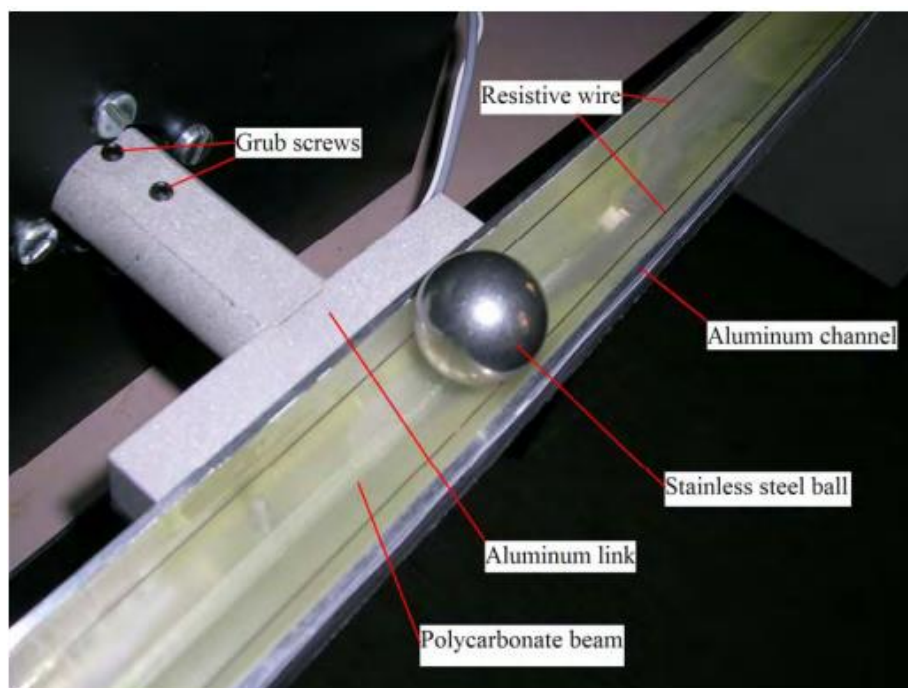


Figura 7 - Mecanismo com seus materiais
 Fonte: Wang (2007, P. 15)

Ele também difere por usar um motor de corrente contínua e não servomotor como a maioria, e acoplado a este motor possui um *encoder* angular, para assim ler a posição angular do motor, além de sua velocidade (observar figura 8). O mais interessante deste trabalho além dos fatos já citados acima, é que o autor implementou vários métodos de controle para fazer uma comparação, além de ter feito o modelo matematicamente e via software.

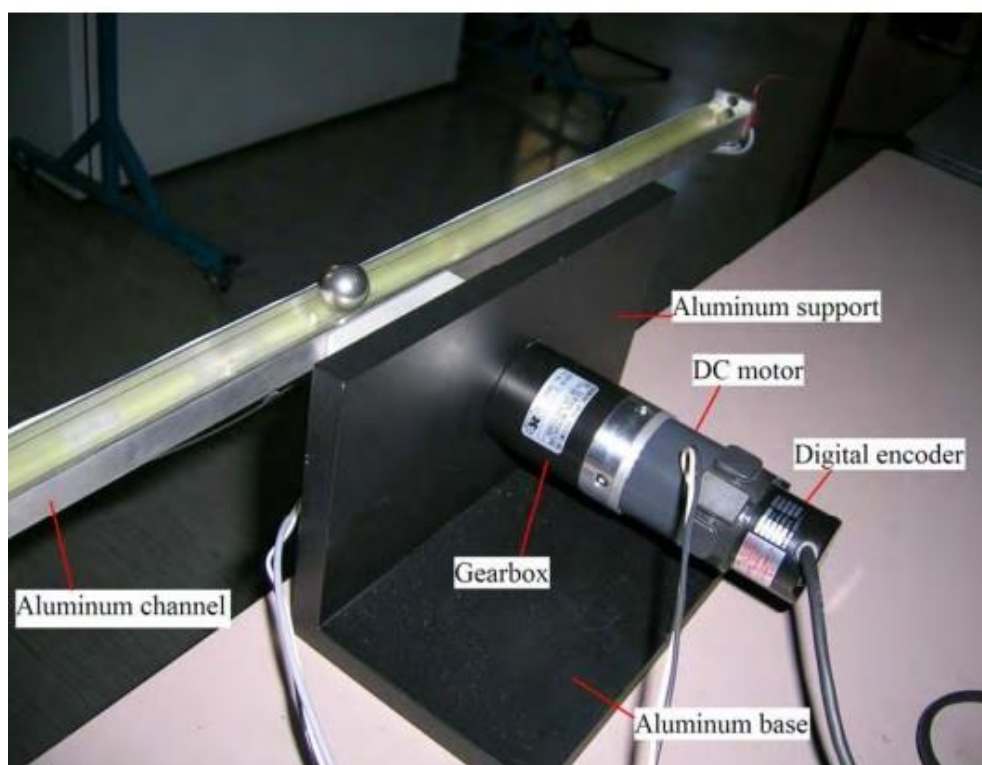


Figura 8 - Vista de trás do mecanismo *Ball and Beam*
Fonte: Wang (2007, P. 17)

2 CONSTRUÇÃO MECÂNICA DO BALL AND BEAM MODIFICADO

A construção mecânica deste projeto pode ser dividida em duas partes: uma relativa à montagem do carrinho e outra relativa à estrutura em que o carrinho irá deslizar, a parte que corresponde ao mecanismo de quatro barras.

A primeira coisa a ser feita, foi estudar os outros projetos e como poderia ser feito o modelo deste projeto. Feito isso, foram escolhidos os materiais a serem utilizados, e neste quesito entraram alguns materiais de impressora sem uso, da área de ferromodelismo, aeromodelismo entre outros, tais materiais podem ser vistos abaixo. Tendo como seria mais ou menos o modelo, e os materiais que seriam usados, foi decidido antes fazer uma representação em *Solidworks* para ver como ficaria aproximadamente, o que pode ser analisado nas figuras 09, 10 e 11.

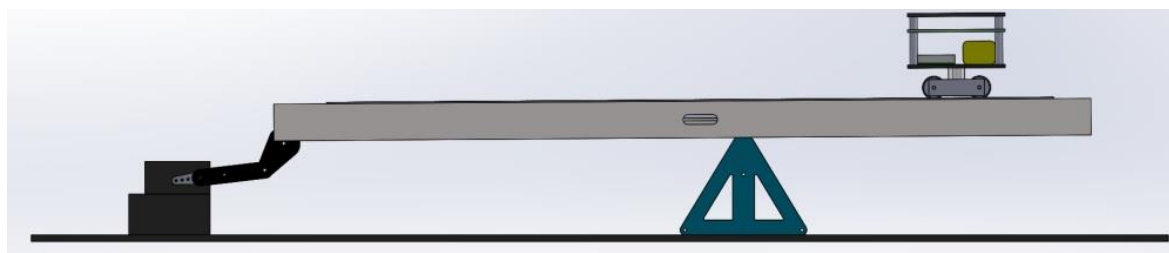


Figura 09 - Vista frontal do Mecanismo feito em *Solidworks*
Fonte: Autoria Própria.

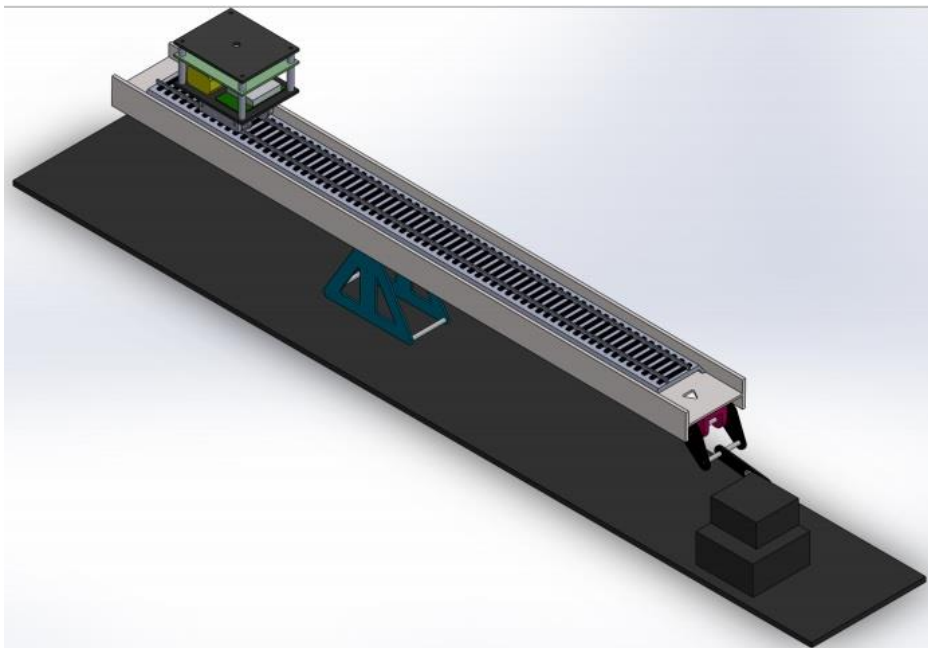


Figura 10 - Vista isométrica do Mecanismo feito em *Solidworks*
Fonte: Autoria Própria.

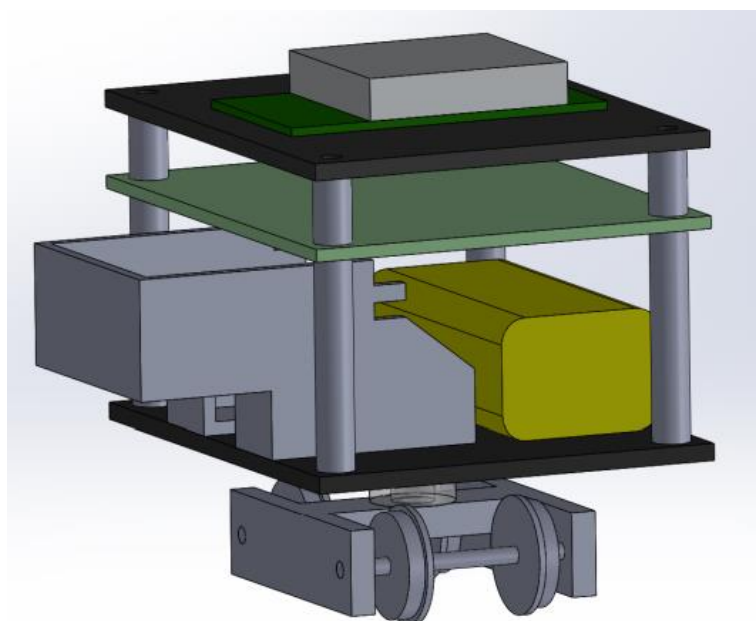


Figura 11 - Vista isométrica do carrinho feito em *Solidworks*
Fonte: Autoria Própria.

MATERIAIS UTILIZADOS:

- Balanceador de hélice de helicóptero (figura 12);
- Placa de fibra de carbono;
- Parafuso;
- Espaçador;
- Tubo de acrílico;
- Trilho de ferromodelo (figura 13);
- Suporte de fita de encoder;
- Carrinho obtido de ferromodelo (figura 14);
- Calha de alumínio (figura 15);
- Ferramentas diversas.



Figura 12 - Balanceador de hélice de helicóptero
Fonte: Store Hobbies (2013, P. 01)

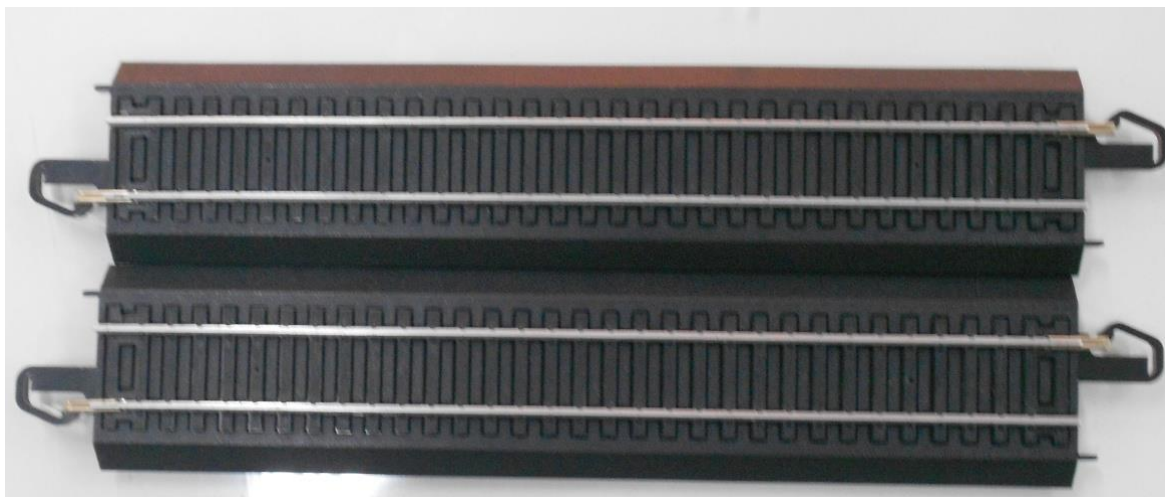


Figura 13 - Trilha Ferromodelo
Fonte: Autorial Própria.



Figura 14 - Carrinho Obtido de Ferromodelo
Fonte: Autorial Própria.



Figura 15 - Calha de alumínio
Fonte: Autorial Própria.

Com o modelo aproximado em *Solidworks* feito, e os materiais em mão, foram montados a estrutura do mecanismo e o carrinho, através de processos de

corte, furação, torneamento, entre outros e o resultado pode ser observado nas figuras 16, 17, 18, 19 e 20.

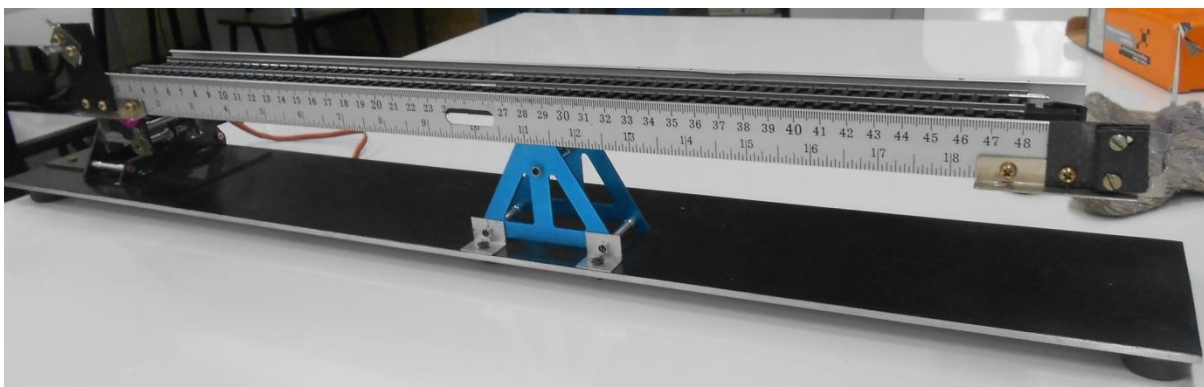


Figura 16 - Modelo Físico da estrutura do *Ball and Beam*
Fonte: Autoria Própria.



Figura 17 - Modelo Físico da estrutura do *Ball and Beam*
Fonte: Autoria Própria.

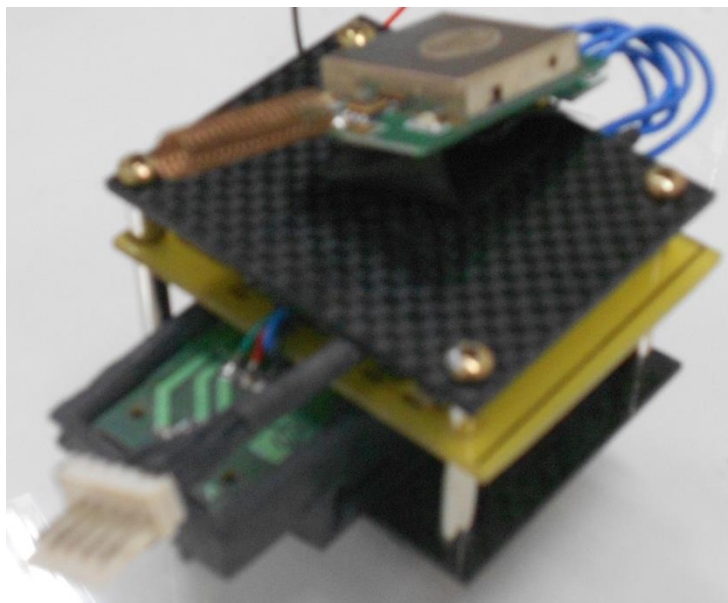


Figura 18 - Modelo Físico do carrinho do *Ball and Beam*
Fonte: Autoria Própria.



Figura 19 - Modelo Físico da estrutura e carrinho do *Ball and Beam*
Fonte: Autoria Própria.

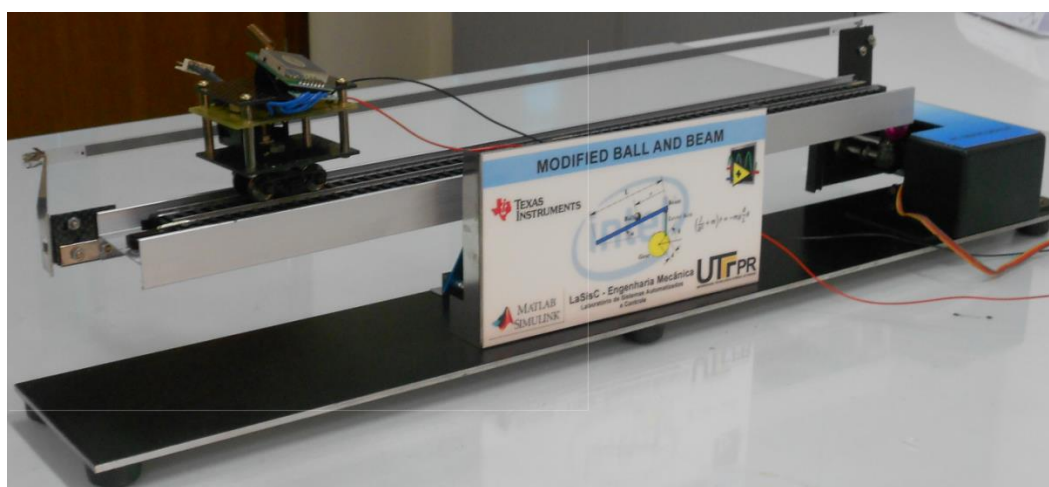


Figura 20 - Modelo Físico da estrutura e carrinho do *Ball and Beam*
Fonte: Autoria Própria.

3 CONSTRUÇÃO ELÉTRICA E COMPUTACIONAL DO MECANISMO BALL AND BEAM MODIFICADO

3.1 INTRODUÇÃO

Antes de falar realmente da construção da parte eletrônica e computacional, será explicado como se dará o funcionamento da mesma.

O carrinho possui uma placa que faz a leitura da sua posição, através de um sensor encoder, esse sinal é transmitido via RF para um receptor *wireless* o qual está ligado a entrada serial do computador, então o programa *Labview* realiza a leitura desses dados, os compara com a posição desejada e através de métodos de controle irá gerar um sinal de retorno para a placa com microcontrolador, onde a mesma irá transformá-lo em PWM RC e enviá-lo para o servomotor, o qual mudará sua posição, conseqüentemente gerando uma nova medida para o encoder e o processo se repetirá até que a posição desejada seja alcançada. Tal funcionamento pode ser melhor visualizado através do esquema da figura 21.

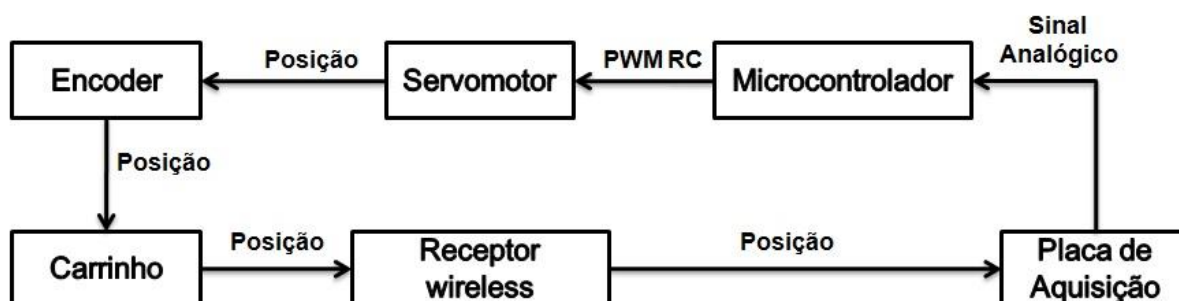


Figura 21 - Funcionamento do mecanismo *Ball and Beam*.
Fonte: Autoria Própria.

3.2 PLACA DO CARRINHO

A placa do carrinho, confeccionada no laboratório de sistema e controle (LaSisC), tem como função realizar a leitura do encoder linear e transmiti-lo via *wireless* para o computador que realizará o controle do mecanismo. A leitura e transmissão de tal sinal é feita utilizando o microcontrolador dsPIC30F2010. Para

melhor compreensão da construção da placa será introduzido a seguir uma revisão sobre encoder, microcontrolador dsPIC30F2010, Eagle PCB.

3.2.1 Encoder

De acordo com Matias (2002, p.34), o encoder “é um transdutor que converte um movimento angular ou linear em uma série de pulsos digitais elétricos”. O autor coloca, ainda, que “esses pulsos gerados podem ser usados para determinar velocidade, taxa de aceleração, distância, rotação, posição ou direção”.

O encoder possui um disco no caso se for rotativo e uma fita se for linear, a qual possui janelas transparentes opacas, alternadas. Essa fita ou disco é iluminado na perpendicular e suas janelas geram na parede oposta onde é encontrado o receptor imagens, as quais indicam a posição que o disco/fita se encontra e então é gerado um pulso elétrico na em sua saída como pode ser notado nas figuras 22 e 23.

Uma notação sobre encoders é que eles são divididos em incremental e absoluto, ambos funcionam através de dois pulsos defasados em 90° que nada mais são do que o canal A e B, tais pulsos além de indicar a posição mostram se é positivo ou negativo, isso vai depender se o canal B está adiantado ou atrasado em relação ao canal A, no primeiro caso indica sinal positivo e no segundo sinal negativo (tal funcionamento pode ser observado melhor nas figuras 24 e 25). Apesar de o funcionamento ser parecido, eles diferem no fato que o incremental mede seus pulsos através do pulso zero, ou seja quando você fornece energia para o encoder, a posição que ele se encontra é dada como a zero, enquanto a posição do absoluto é determinada através de um código que para cada posição é única, e diferente do outro tipo se ele perde a energia ele não perde seu referencial zero, toda vez que for realimentado seu zero permanecerá o mesmo sem sofrer alteração (MATIAS, 2002).

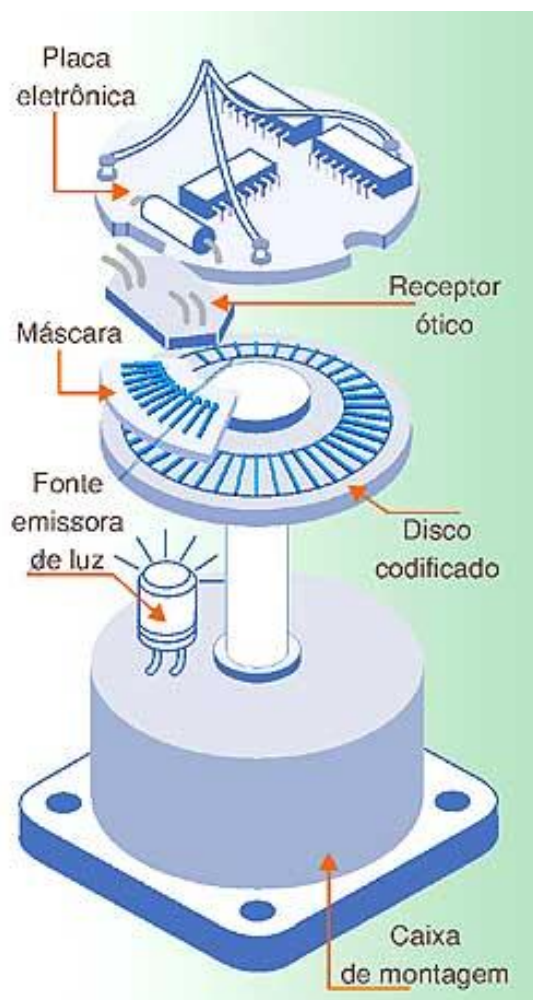


Figura 22 - Princípio de funcionamento de um encoder rotativo
 Fonte: Matias (2002, P.36)

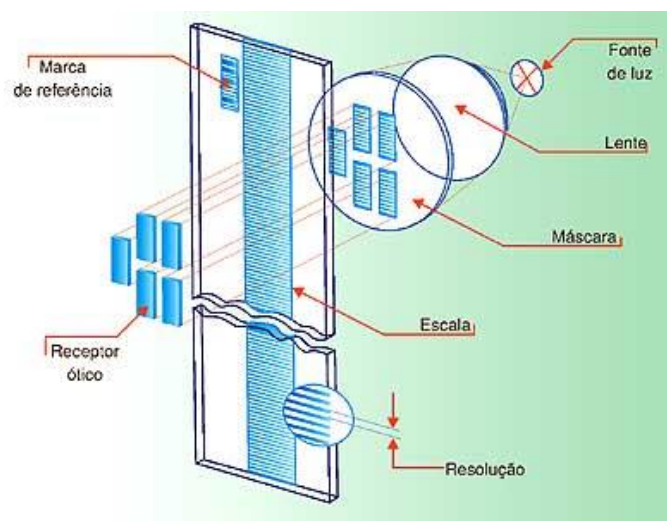


Figura 23 - Princípio de funcionamento de um encoder linear
 Fonte: Matias (2002, P.36)

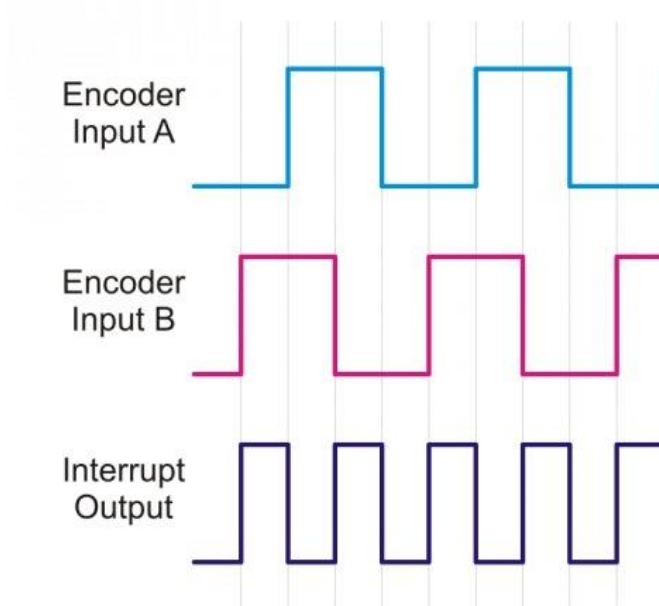


Figura 24 - Representação gráfica do funcionamento de um encoder
 Fonte: LETS MAKE ROBOTS (2013, P. 01)

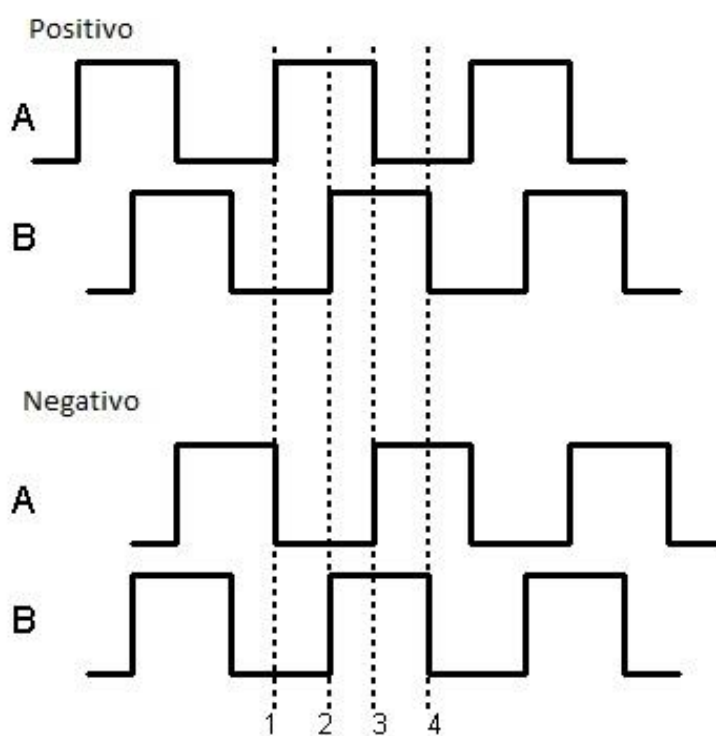


Figura 25 - Representação gráfica do funcionamento de um encoder em relação a sentido positivo e negativo.
 Fonte: QSL (2013, P. 01)

3.2.2 Microcontrolador dsPIC30F2010

Segundo Shimada (2011, p. 27) um microcontrolador “pode ser considerado um pequeno computador integrado em um único circuito, contendo um processador, memória e periféricos de entrada e saída”.

Os microcontroladores tem tido uma ampla aplicação hoje em dia, eles podem ser encontrados em controles remotos, em sistemas de automação residencial, no controle de motores, em aparelhos residenciais como microondas entre outros, este aumento tem se dado devido de ter baixo custo e consumo de energia. Eles podem ser divididos em vários tipos, porém não é objetivo do trabalho entrar em detalhes, o que cabe saber é que o que será usado é o da família da microchip dsPIC30F2010 que atende as nossas necessidades que são leitura de um encoder, transmissão de dados via RF e conversão de sinal analógico para digital, no caso PWM RC. A seguir na figura 26, pode ser observado o diagrama dos pinos do microcontrolador que será usado.

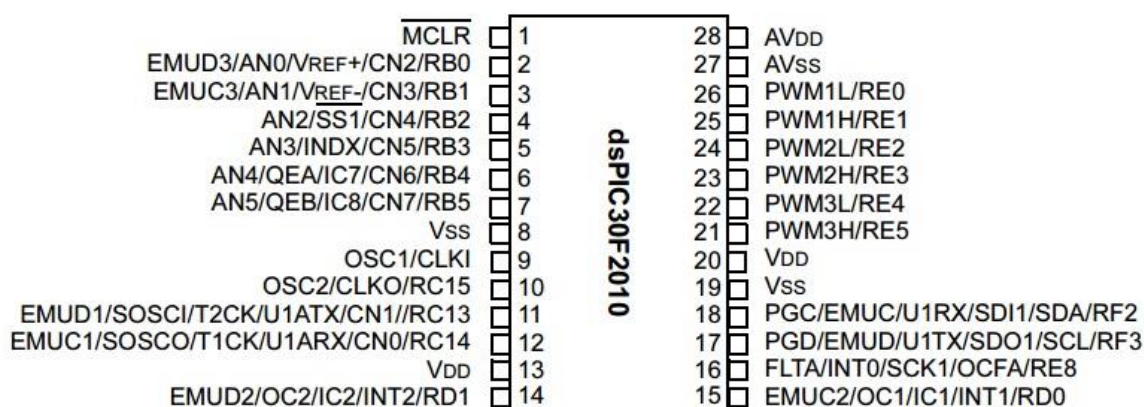


Figura 26 - Diagrama de Pinos dsPIC30F2010

Fonte: Datasheet dsPIC30F2010

3.2.3 Eagle PCB

O Eagle PCB é um software que tem o intuito de auxiliar na confecção de placas de circuito impresso, através dele é possível realizar a ligação dos componentes fazendo assim o desenho esquemático, e depois de pronto, é possível simular virtualmente para verificar se com as ligações que foram feitas o circuito está funcionando como esperado, e depois de testado ainda é possível criar o layout da

placa, realizando o roteamento e posicionamento dos componentes no tamanho de placa desejado e criando assim as trilhas necessárias através das informações do desenho esquemático.

Tendo conhecimento sobre alguns componentes e software que aqui serão usados, vamos dar continuidade a construção da placa do carrinho. Segue a seguir os materiais que foram necessários para tal processo e depois será mostrado e explicado o esquemático e layout da placa feito no software *Eagle*.

3.2.4 Lista de Materiais

- Placa de fenolite 5x5 cm;
- *Wireless Serial RS232 Port Adapter Module Freescale* (figura 27);
- Encoder da série HEDS-970x ou HEDS-972x;
- LM7805 e dois capacitores necessários;
- Bateria 7,6V (figura 28);
- dsPIC30F2010 smd e componentes periféricos necessários para seu funcionamento e programação;
- Placa dsPIC30F2010 com entradas de fácil acesso (figura 29);
- Materiais necessários para confecção de placa, como solda de estanho, líquido corrosivo, entre outros.



Figura 27 - Wireless Serial RS232 Port Adapter Module Freescale
Fonte: Autoria Própria.



Figura 28 - Bateria 7,6 Volts
Fonte: Autoria Própria.

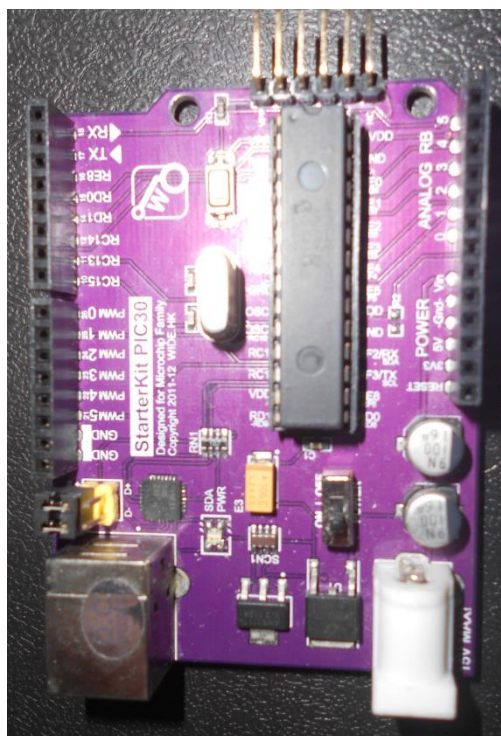


Figura 29 - Placa dsPic30F2010
Fonte: Autoria Própria.

3.2.5 Esquemático

A seguir na figura 30 pode ser visto como ficou a representação esquemática do circuito feita no *software* Eagle PCB.

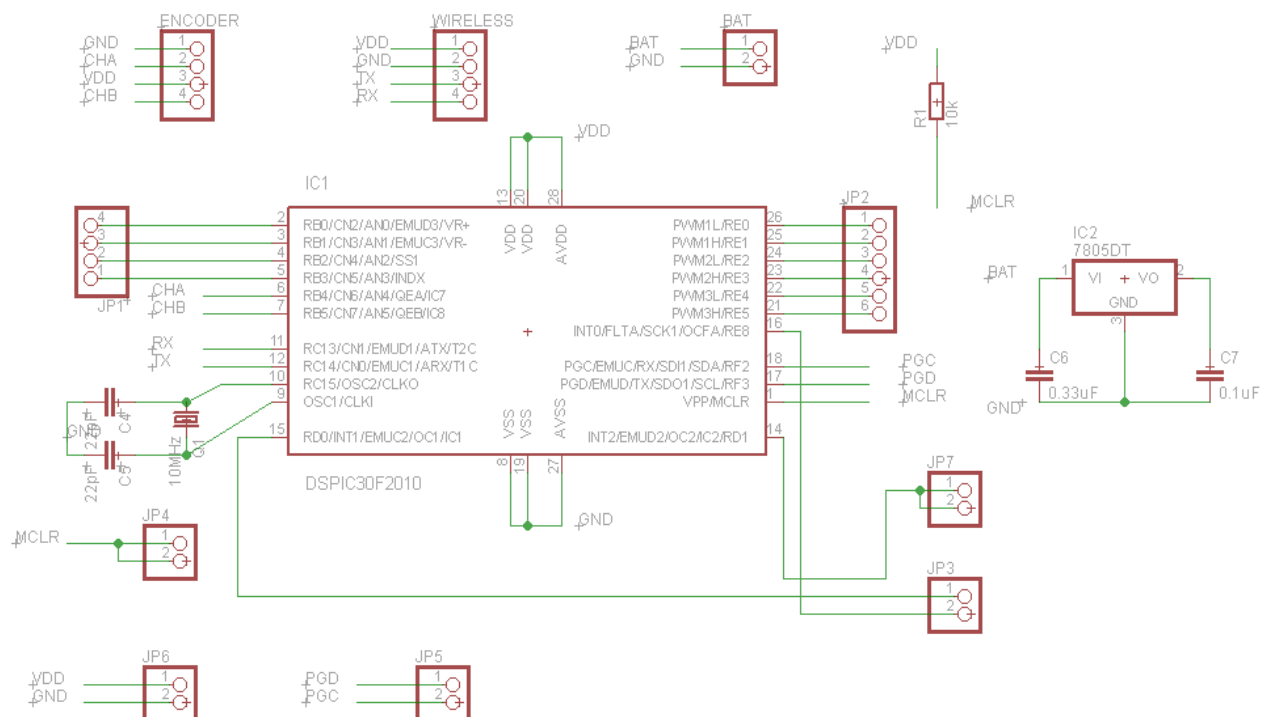


Figura 30 - Representação esquemática do circuito feito no software Eagle PCB
Fonte: Autoria Própria.

Para melhor entendimento, serão explicadas as principais ligações feitas na figura 30. Cabe ressaltar que os dados dos componentes foram tirados dos de seus *datasheets*, os quais são facilmente encontrados na *internet*.

3.2.5.1 Encoder x dsPIC30F2010

O encoder possui quatro conectores, como pode ser visto na figura 31, os mesmo são ligado à portas específicas do microcontrolador, ficando da seguinte forma, CH B ligado a porta 07(QEB), CH A porta 06(QEA), Vcc porta 20 e Gnd porta 19.

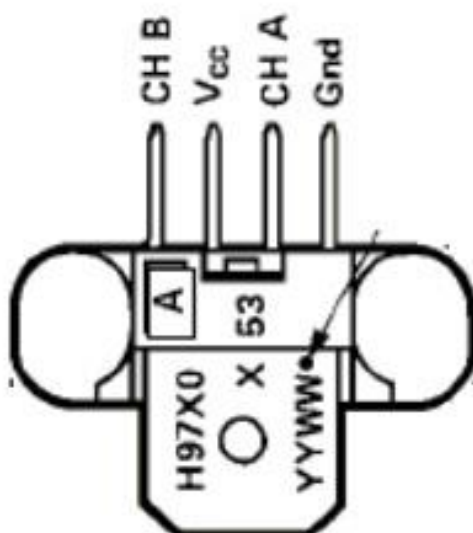


Figura 31 - Desenho esquemático do encoder da família H97X0
 Fonte: Datasheet encoder H97X0

3.2.5.2 Wireless x dsPIC30F2010

O *wireless* possui seis conectores, porém apenas quatro serão utilizados (Tx,Rx,Gnd e Vcc), os quais foram ligados da seguinte forma Tx na porta 012(ARX), Rx com a porta 011(ATX), Vcc porta 20 e Gnd porta 19. Note que o Tx do receptor está ligado com o Rx do microcontrolador e vice versa, isso ocorre pois a função do receptor nesse caso é pegar o sinal que foi lido pelo microcontrolador e mandar para outro receptor.

3.2.5.3 Bateria x LM7805 x dsPIC30F2010

Como a bateria em questão fornece 7.6 volts e o microcontrolador trabalha com 5 volts, é necessário o uso do LM7805 para abaixar a tensão para 5 volts. Na figura 32 pode-se observar como é feita a ligação no LM7805, onde V-IN é onde é conectado o pólo positivo da bateria, e o V-OUT é a saída de 5 volts que fornecerá a tensão para a placa, ele será ligado a porta 20 e tanto o Gnd do regulador quanto da bateria serão ligados juntos a porta 19.

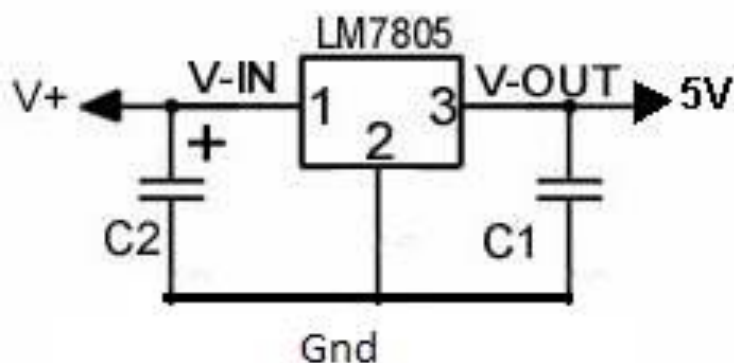


Figura 32 - Ligação Lm7805
 Fonte: Adaptado de Eletronica (2013, P. 01)

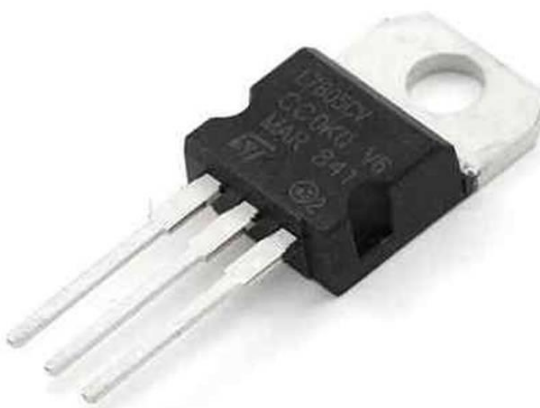


Figura 33 - LM7805
 Fonte: Robotgear (2013, P. 01)

3.2.5.4 Programador

Outra ligação que precisa ser feita é a ligação com um conector do programador do microcontrolador, que utiliza as portas 17(PGD), 18(PGC), 1(MLCR), 20(VDD) e 19(Gnd).

Após ter sido realizado o desenho esquemático da placa no *software*, o próximo passo é utilizá-lo para que faça o seu roteamento e nos dê o *layout* da placa, tanto em relação às trilhas quanto as componentes, o mesmo pode ser visualizado na figura 34 e 35. Com o *layout* pronto, o próximo passo foi imprimir ele na folha adequada, para que se pudesse realizar a técnica de transferência para a placa de fenolite através de um prensa de camiseta. A próxima etapa foi corroer a placa para que o que apenas sobrasse fosse as trilha na parte cobreada e o *layout*

dos componentes na parte não cobreada, o resultado disso pode ser visto nas figuras 36 e 37.

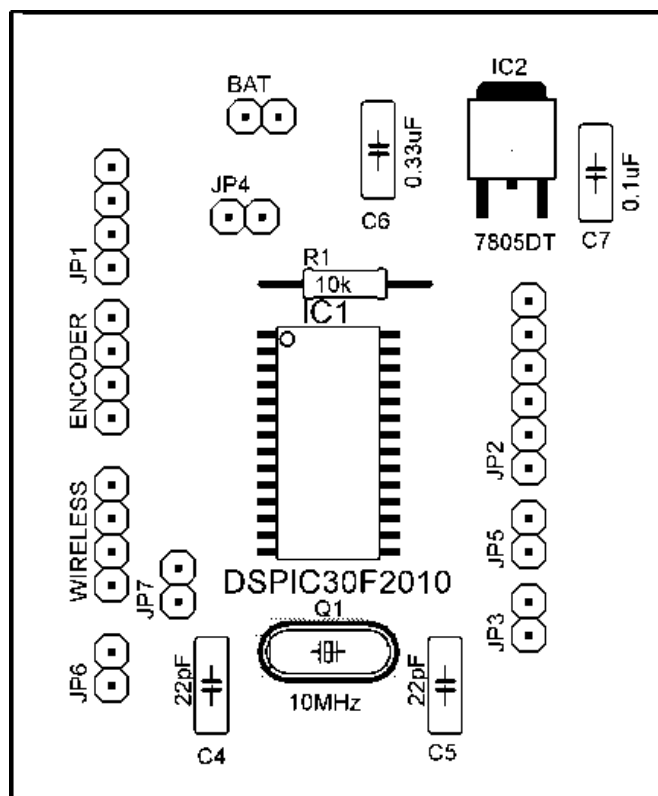


Figura 34 - Layout da placa feita em *sotware* Eagle PCB
Fonte: Autoria Própria.

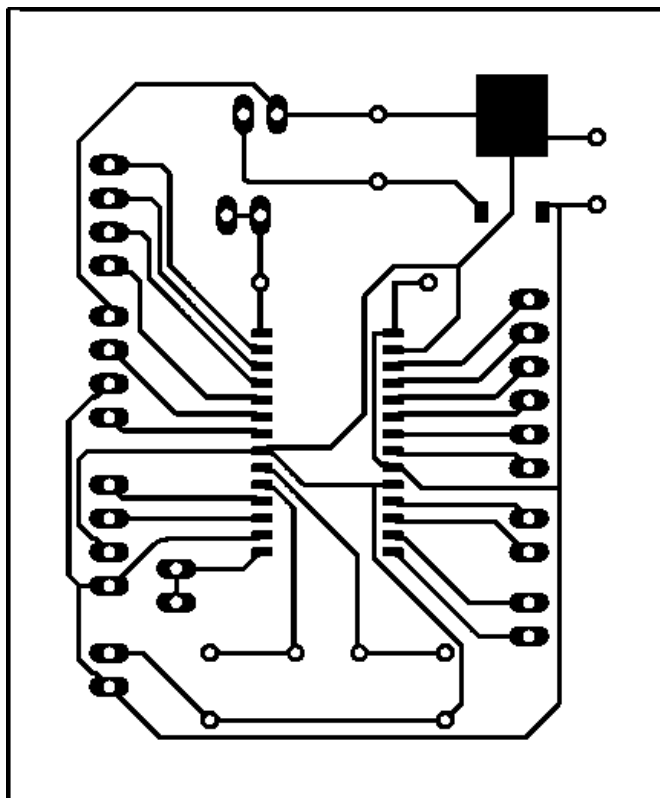


Figura 35 - Layout da placa feita em software Eagle PCB
Fonte: Autoria Própria.

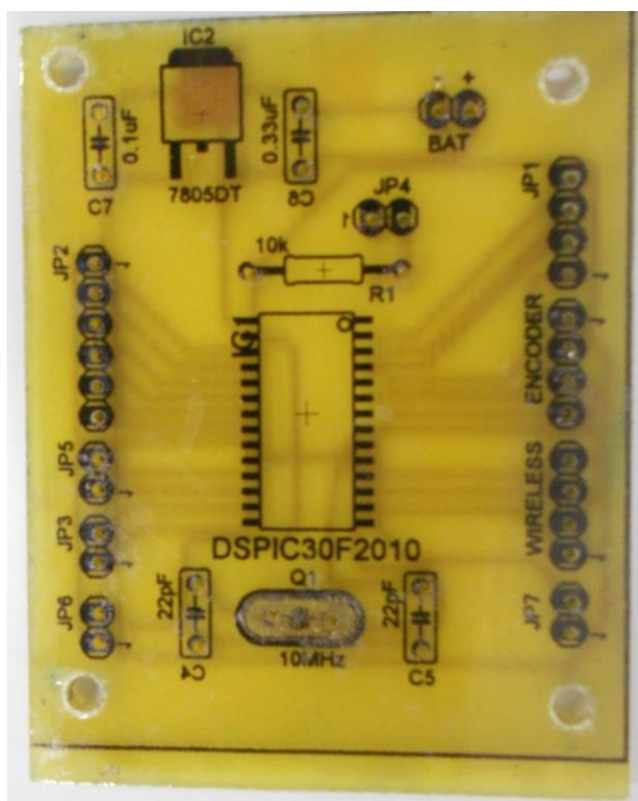


Figura 36 - Placa já com impressa e corroída parte superior
Fonte: Autoria Própria.

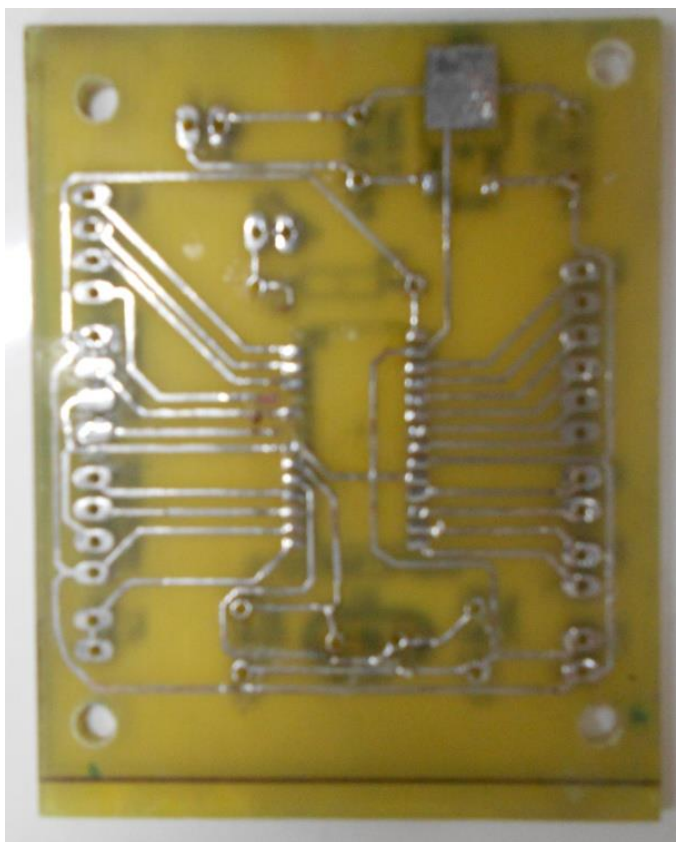


Figura 37 - Placa já com impressa e corroída parte inferior
Fonte: Autoria Própria.

Com a placa corroída, foram feitos os furos com uma mini-retífica no lugar onde é preciso soldar os componentes e fios, logo após entrou a parte da soldagem, começando com a estanhagem da trilha para diminuir a chance de ocorrer mau contato, e conseguinte soldagem de todos componentes e fios em seus devidos lugares (figura 38). Enfim o circuito está pronto, porém é preciso programar o dsPIC30F2010 para que realize a leitura do encoder e transmita esse dado por suas portas ARX e ATX para o receptor wireless, tal programa pode ser encontrado no anexo A e também é preciso realizar o teste para ver se está tudo funcionando como deveria.

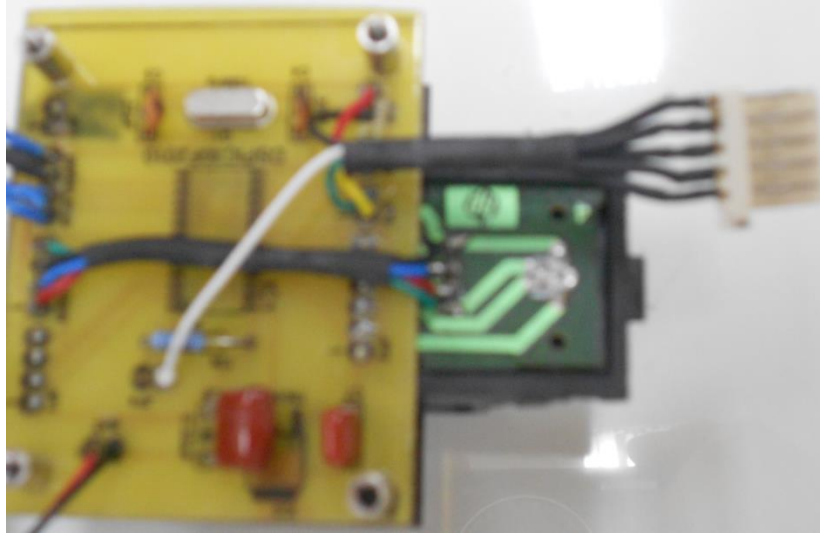


Figura 38 - Placa corroída e com todos os componentes soldados
Fonte: Autoria Própria.

3.2.6 Outros

Terminada a parte que compete ao carrinho, falta descrever as outras ligações necessárias, que são: a ligação entre o receptor *wireless* 1 com o receptor *wireless* 2; *wireless* 2 com o computador, placa de aquisição e servomotor com outro dsPIC30F2010. Porém, antes será dada uma introdução sobre alguns assuntos que serão tratados e alguns componentes que serão usados, como Labview, Sinal analógico e digital, PWM RC, placa de aquisição, servomotor.

3.2.6.1 Labview

É um *software* de programação gráfica desenvolvida pela empresa National Instruments (NI), que tem como objetivo ajudar os engenheiros ao realizar testes de sistemas de pequeno e grande porte, ele tem aplicação principalmente na área de medição e automação, ele pode ajudar muito no monitoramento, como por exemplo, em indústrias de bebidas que é necessário fazer o controle de alguns parâmetros: como temperatura, pressão entre outros. O que é interessante neste programa que além de ele ser facilmente comunicado com *hardware*, como sensores e outros componentes, ele monitora seus valores e também pode ajudar a controlar o processo, fazendo parar, reiniciar ou iniciar dada circunstância pré-determinada, no

caso de um forno, por exemplo, em que há o controle da temperatura e chegada a temperatura desejada ele deve ser desligado, ao invés de um operador ficar monitorando, o programa pode além de fazer isso realizar o desligamento, e também com o programa e *hardware* adequados pode-se monitorar um processo a longa distância, como por exemplo, se o responsável pela manutenção dos parâmetros nos valores desejados esteja longe dos mostradores ou esteja até mesmo fora da indústria, ele pode monitorá-los através de um computador de casa ou até mesmo programar para receber um bip no celular caso algo saia do comum.

No caso deste projeto, ele servirá para receber os dados do encoder e através de programação realizar o controle e gerar o sinal para gerar o PWM RC.

3.2.6.2 Placa de Aquisição

Trata-se de uma placa que é acoplada a um computador, e através dela é ligado um cabo com conectores, os quais serão usados para comunicar com alguns hardwares, e realizar leituras de encoders, gerar sinais: pwm, digital, analógico e diversas outras funções. No caso deste trabalho será usada o modelo PCI 6251 da empresa NI, e ela será utilizada para auxiliar no controle do *ball and beam*, onde ela gerará o sinal de controle para gerar o PWM RC, ela normalmente é usada comunicando junto com o software Labview, e que neste caso não será diferente.



Figura 39 - Imagem ilustrativa de uma placa de aquisição da NI série M
Fonte: National Instruments Corporation (2013)

NI 6251

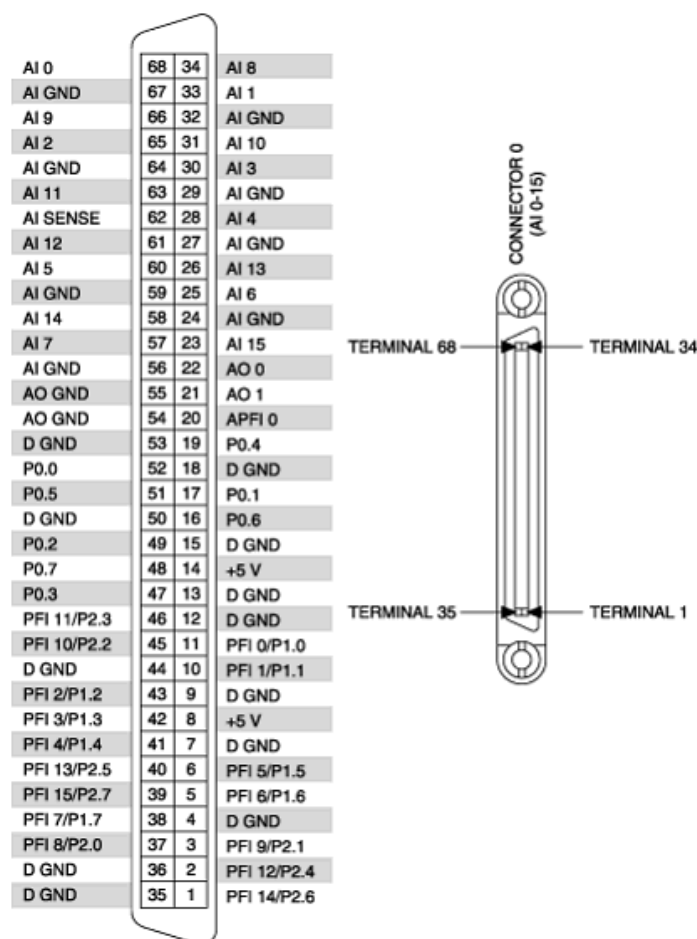


Figura 40 - Pinagem PCI 6251
Fonte: Datasheet PCI 6251

3.2.6.3 Sinal analógico x Sinal digital

Sinal digital e sinal analógico são um meio de representar dados, como leitura de encoder, sons, entre outros. O que difere um de outro é que o analógico corresponde a onda que pode assumir diversos valores em um determinado espaço de tempo, já o sinal digital só assume os valores de 1 e 0, isso porque o sinal digital é a transformação do sinal analógico em números binários assumindo assim apenas 0 e 1. Para melhor visualizar a diferença entre os dois sinais segue a figura 41.

Outra diferença entre esses dois sinais, é que o sinal analógico vai perdendo a sua qualidade a cada vez que é copiado, já o digital como ele armazena informações através de 0s e 1s, ele não perde qualidade quando é copiado, tal fato pode ser notado quando comparamos disco vinil com os cds de hoje em dia, nos

vinils eram gravados sinais analógicos e a cada vez que era copiado perdia a sua qualidade, já o cd é gravado utilizando sinal digital, o qual não perde sua qualidade a cada vez que é copiado.

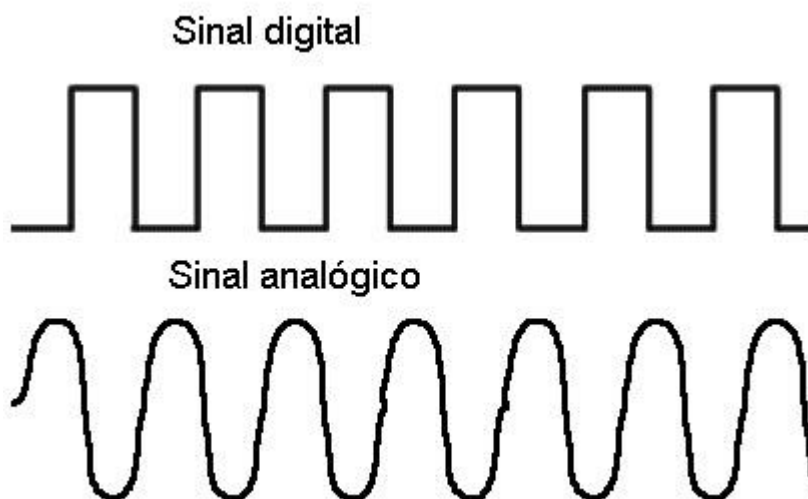


Figura 41 - Sinal Analógico x Sinal digital
Fonte: Smith (2013)

3.2.6.4 Pwm RC

Antes de ser definido o que é PWM RC é melhor compreender o que é PWM (*Pulse width Modulation* no inglês ou no português Modulação de largura de pulso), como o próprio nome diz, trata-se do controle da largura de pulso, para melhor entender, imagine um circuito onde há uma chave *on/off* e em uma das extremidades está ligada uma carga a qual deseja-se controlar, pode ser um motor DC por exemplo, e na outra extremidade a fonte fornecendo energia, como na figura 42. Através da chave é possível controlar se passa ou não corrente para carga, ou seja se a carga liga ou desliga, porém deseja-se não apenas controlar potência máxima ou mínima e sim intermediárias, então se desligar e ligar rápido a chave em períodos pequenos podemos determinar potências intermediárias, por exemplo se deseja 50% e durante um período inteiro a chave ficar desligada metade do tempo e ligada na outra metade temos uma potencia média de 50%, mas esse período tem que ser muito pequeno, e não seria possível desligar e ligar rapidamente com uma chave comum, então para solucionar este impasse é considerado a substituição da mesma por componente que desenvolvem a mesma função de liga/desliga com um

intervalo de tempo bem pequeno, são eles os FETS de potência, IBGT, SCR, transistor bipolar, entre outros.

Deste modo que surge o PWM, que é a variação de um pulso durante um período para encontrar a potencia desejada, alterando o período que a chave fica ligada e desliga em um determinado tempo, tal efeito pode ser mais bem visto na figura 43. É importante ressaltar que o tempo em que o pulso fica alto, ou seja, o período que fica ligado é chamado de *Duty Cycle*.

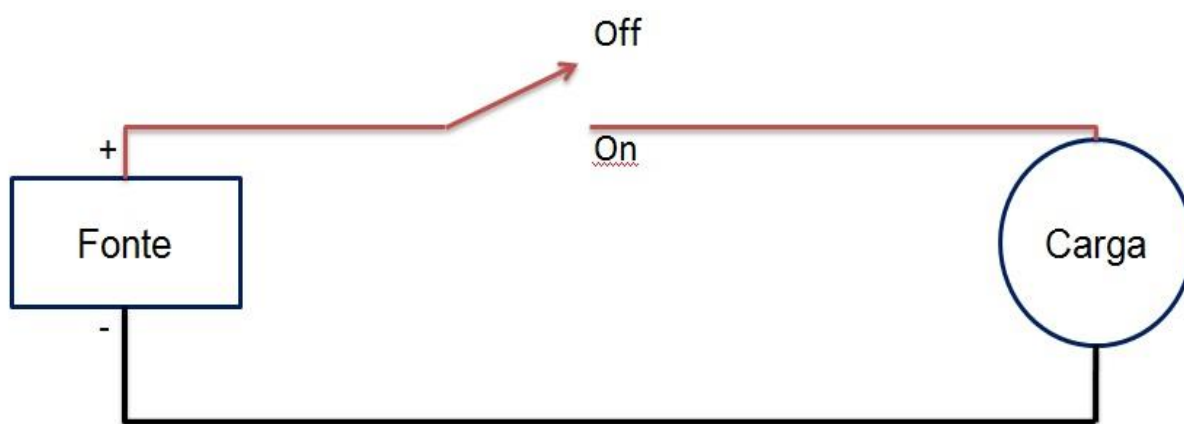


Figura 42 - Esquemático de um circuito com uma carga e uma chave
Fonte: Autoria Própria.

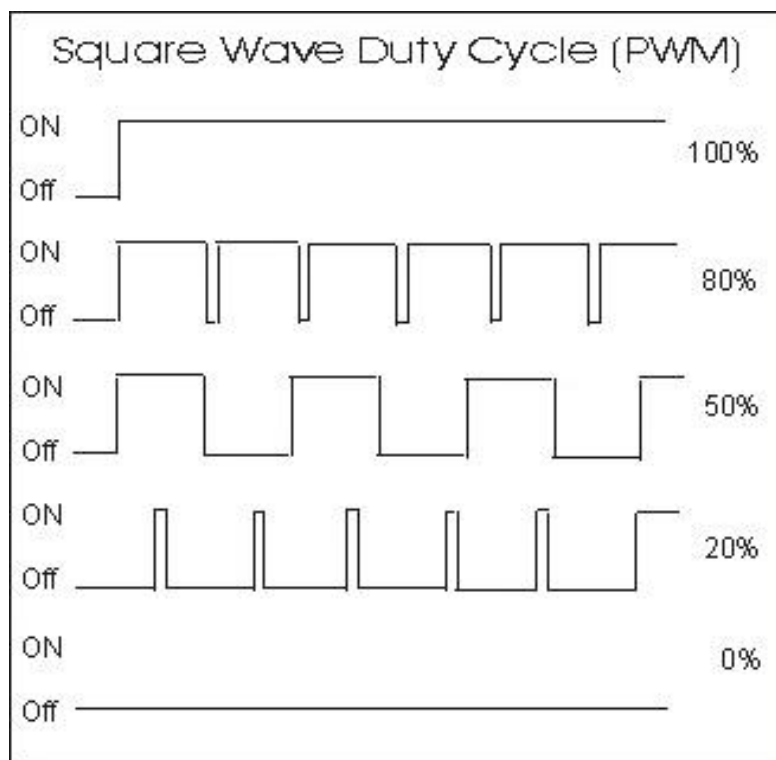


Figura 43 - Representação do sinal PWM de acordo com o *Duty cycle*
Fonte: Imagesco (2013)

Após compreender como funciona o PWM, poderá ser melhor entendido como funciona o PWM RC, o qual será usado neste trabalho para realizar o controle de um servomotor. Esta forma de PWM é geralmente usada quando se trata de brinquedos de controle remotos, como os aeromodelos, automodelos, entre outros, que trabalham com o padrão *futaba*, que neste caso será controlado a posição do servomotor e não a potência como no caso de um motor DC.

O que o RC difere do comum, é que ele além de como foi dito é normalmente usado para controlar posição, é que também ele não trabalha como os outros que definimos vários *Duty cycle* e conseqüentemente períodos diferente de Ton, neste caso ele normalmente trabalha apenas na faixa de 1 a 2 ms com a chave em alta, ou seja com período Ton, e a cada posição dentro desta faixa dá um posição ao servomotor, sendo 1 ms 90 graus para esquerda, 1,5 ms neutro e 2 ms 90 graus para direita (pode ser melhor observado na figura 44), podendo variar em alguns servos com maior varredura de ângulo, mais geralmente se dá desta forma.

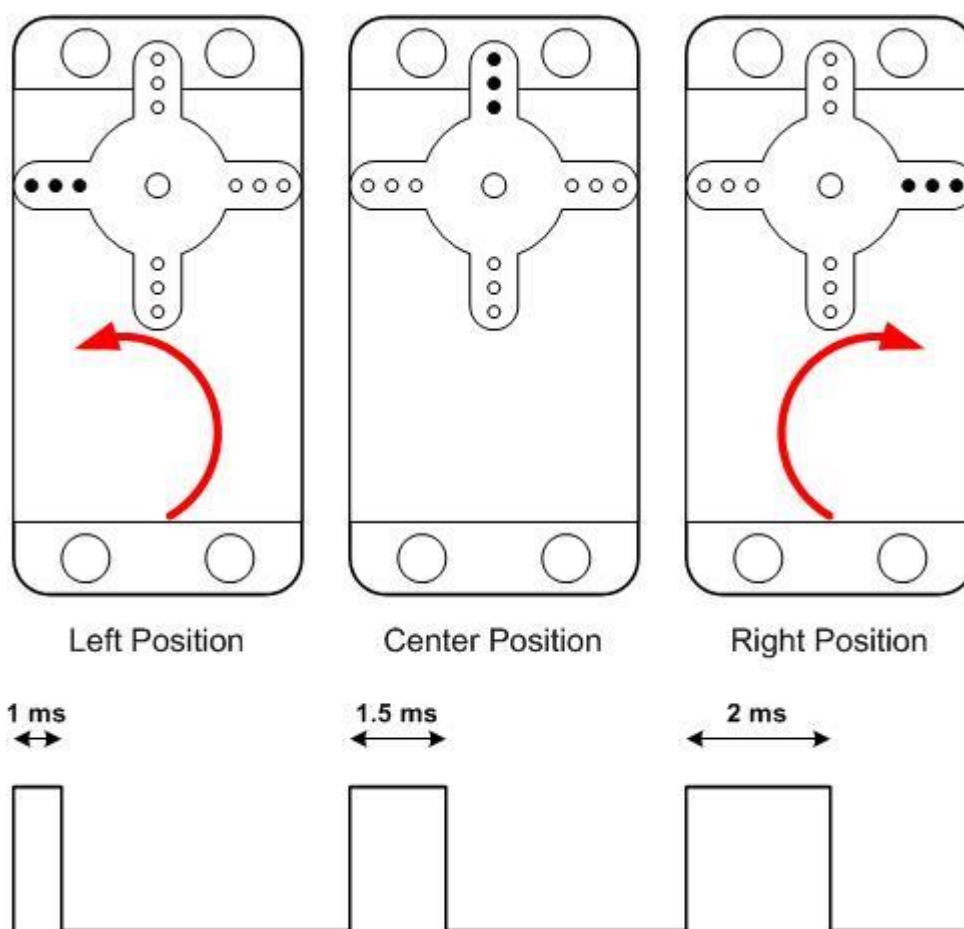


Figura 44 - Funcionamento de um servo com Pwm RC
 Fonte: EBLDC (2013)

3.2.6.5 Servomotor

Diferente dos motores convencionais de corrente contínua os servomotores não podem girar com 360 graus de liberdade, normalmente eles giram apenas 180 graus, e além disso, eles não ficam girando indefinidamente quando ligados, eles só giram quando é dado um sinal, pois ele trabalha com em malha fechada e sua posição pode ser controlada através de um sinal de controle, ou seja ele não possui apenas duas alimentações como os outros, ele possui três, uma para negativa, outra positiva e outra de sinal para dizer para onde ele vai e quando vai, se esta não receber sinal o servo não se movimenta. Outra curiosidade é que eles utilizam como sinal, o PWM RC e eles tem aplicação quando se quer ter controle da posição de alguma coisa, como no caso projeto aqui realizado, que quer se controlar a posição

da haste principal para poder controlar o carrinho, ou seja, ele é muito usado na parte de robótica e afins.



Figura 45 – Servomotor
Fonte: Wikipedia (2013)

3.2.6.6 *Wireless* 2 x Computador

Pode-se notar que a ligação *wireless* 1 e 2 não foi citada, isto é porque essa comunicação se dá via transmissão de dados sem fio e não há necessidade de cabos. Para a comunicação *wireless* 2 com o computador foram utilizados uma placa com uma porta Rs232 a qual é ligado de um lado no *wireless* 2 e do outro lado através de um cabo serial é ligado a uma porta serial do computador, estas ligações podem ser melhores entendidas nas figuras 46 e 47.

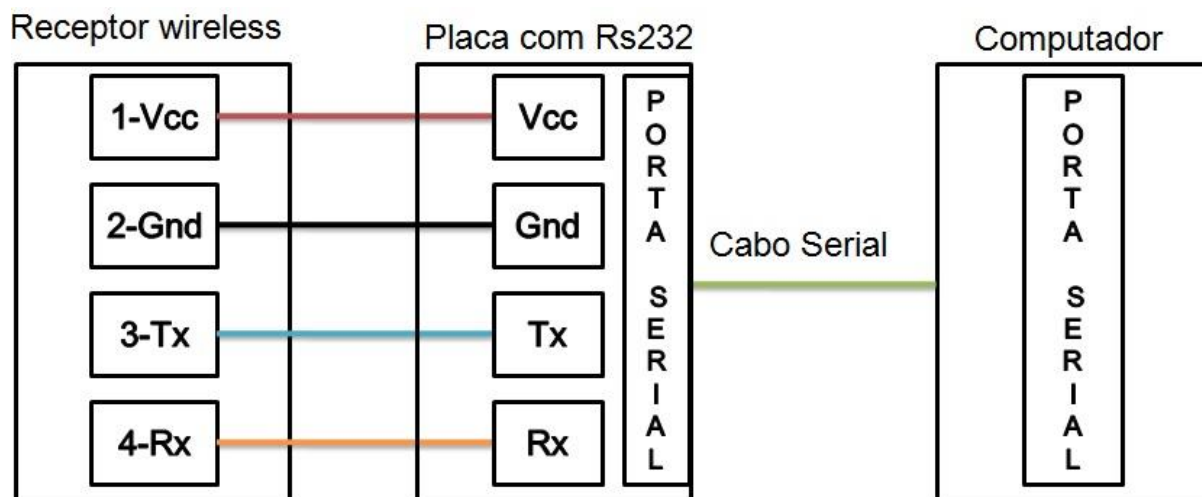


Figura 46 - Ligação Receptor Wireless x Computador

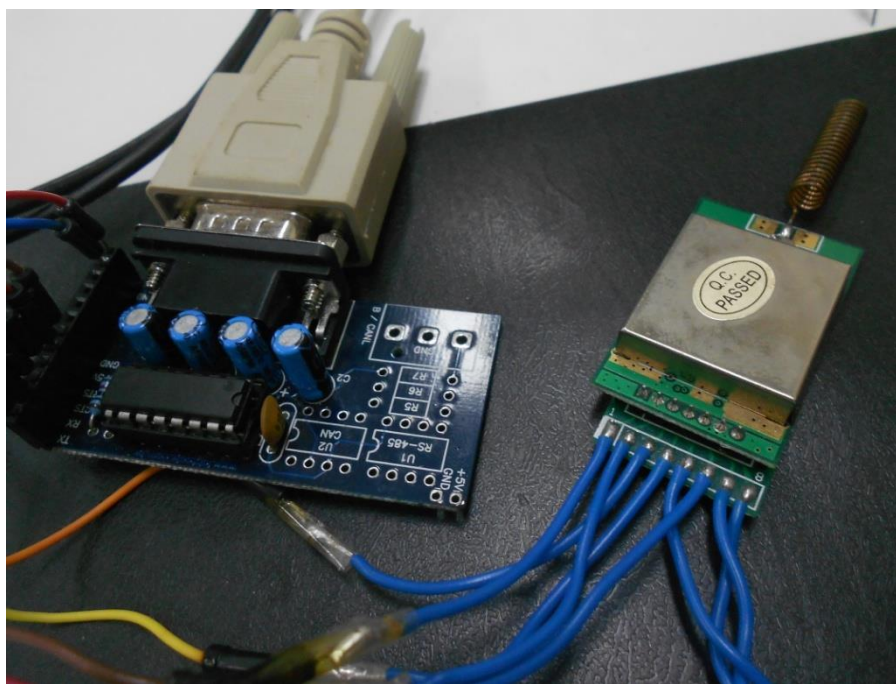


Figura 47 - Foto da Ligação Receptor Wireless x Computador
Fonte: Autoria Própria.



Figura 48 - Ligação cabo serial com porta serial do computador
Fonte: Autoria Própria.

3.2.6.7 Placa de aquisição x Microcontrolador dsPIC30F2010

Após o computador receber os dados via serial, o Labview aplica o controle e manda um sinal analógico para a placa de aquisição (NI PCI 6251) que sai através da porta 22(AO0) e 55(Gnd) da placa de aquisição, as quais são ligadas a uma outra placa com microcontrolador (esta placa foi comprada montada e não confeccionada como a do carrinho, figura 29), sendo que a porta 22 é ligada na porta *ANALOG RB0* e 55 com o Ground da placa, tal ligação é representada pela figura 49.



Figura 49 - Ligação NI PCI 6251 x dsPIC30F2010
 Fonte: Autoria Própria.

3.2.6.8 Servomotor x dsPIC30F2010

O sinal analógico recebido pela placa de aquisição pelo dsPic30f2010 é transformado então em sinal digital e aqui no caso PWM RC, o qual dependendo da sua variação dará uma posição para o servo motor. A Ligação destes dois é feita da seguinte forma o Gnd da placa do microcontrolador é ligado ao *ground* do servomotor e também ao ground uma fonte variável, o pino RD0 é ligado então ao cabo de sinal do servo motor e sobra ainda um cabo do servo que é o Vcc e é ligado a uma fonte variável que fornece 6 volts, a figura 50 a seguir mostra melhor essas ligações.

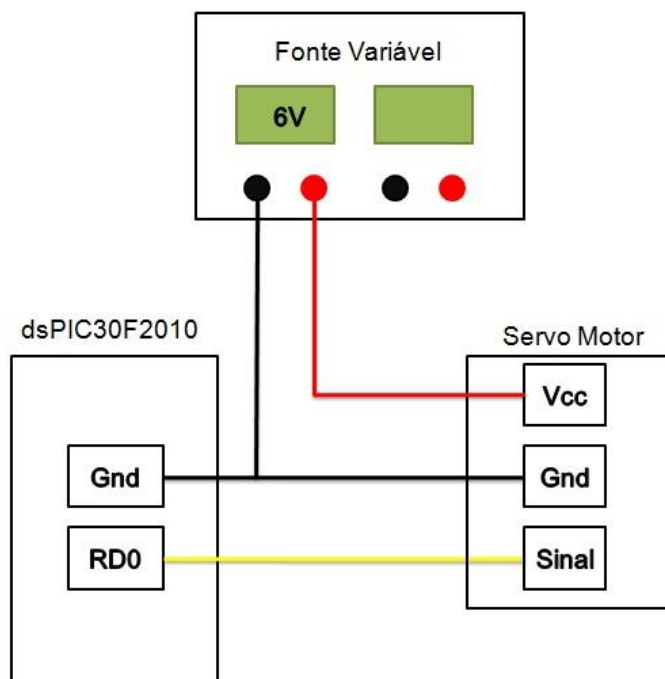


Figura 50 - Servomotor x dsPIC30F2010

Fonte: Autoria Própria.

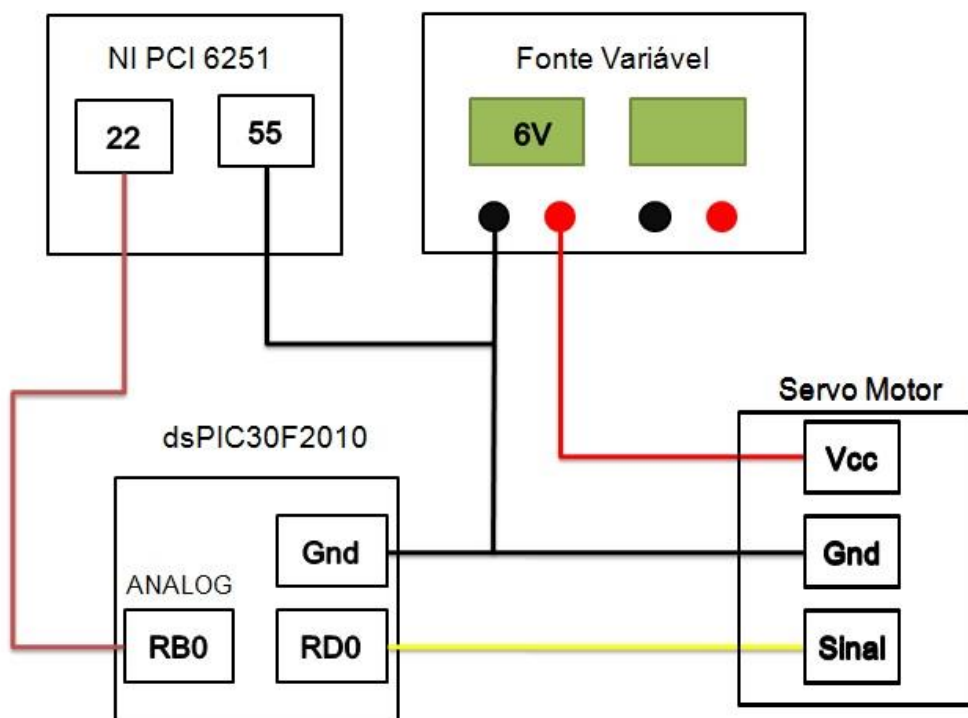


Figura 51 - Ligação completa do mecanismo

Fonte: Autoria Própria.

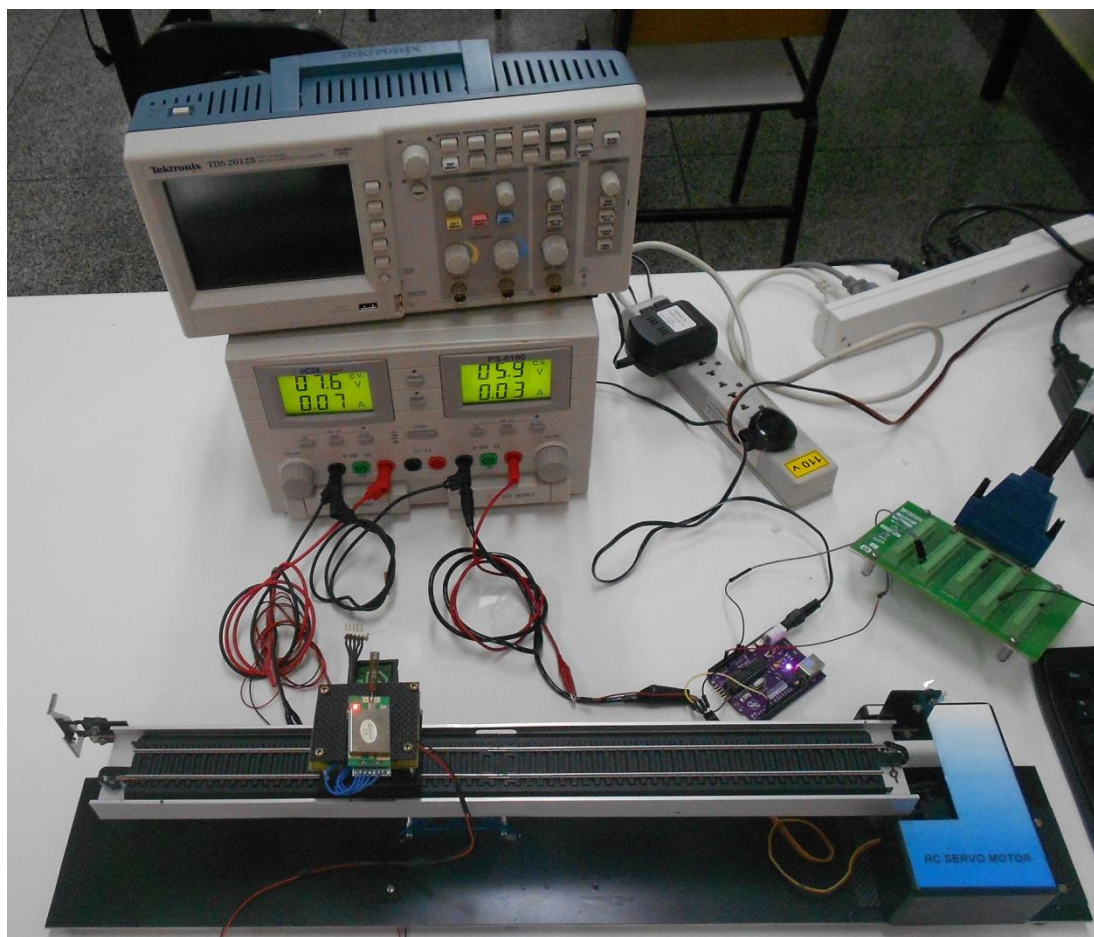


Figura 52 - Imagem da ligação completa do mecanismo
Fonte: Autoria Própria.

4 SINCRONIZAÇÃO COM COMPUTADOR E APLICAÇÃO DE SISTEMA DE CONTROLE

4.1 INTRODUÇÃO

Concluída a parte de construção, foi dado início na parte da aplicação do controle do mecanismo em questão, antes de entrar realmente no assunto, será abordado superficialmente o que é o *software* Adams, Matlab/simulink, sistema de controle e como funcionam os que aqui serão aplicados.

4.1.1 *Software* Matlab/Simulink

Matlab é um *software* muito interessante e funcional para diversas áreas. Ele é um ambiente para desenvolvimento de algoritmos e com as ferramentas que podem ser acopladas pode executar várias outras funções, como na área de redes neurais, simulação na parte de fluidos, simulação de circuitos elétricos, entre outros, mas para o trabalho aqui desenvolvido além da parte do elaboração de algoritmo outra parte que foi utilizada foi a do *Simulink*, que compete a área de programação em blocos bem similar ao Labview, aonde pode-se construir diagramas e simular sistemas dinâmicos diversos, sistemas de freio, absorção de choque e diversos outros, o que foi usado foi a parte de simulação de sistema de controle.

4.1.2 *Software* Adams

Adams é um tipo de programa que está cada vez mais sendo usado, por tratar de um *software* de simulação dinâmica, ele possibilita simular várias plantas. O que é preciso fazer é desenhar o modelo que representa o seu projeto, colocar as juntas necessárias e então aplicar a simulação e verificar o comportamento e obter informações bem úteis, como matrizes de estados e muito mais. Seu uso tem aumentando, pois alguns projetos são muito caros de construir para poder testar e então ver se funciona, então é primeiro simulado neste ambiente para depois ser construído.

No caso deste projeto, foi feito um modelo simplificado no próprio *software*, colocado as juntas necessárias e simulado em determinado espaço de tempo para obter as matrizes de estados, as quais vão ser usadas pelo programa feito no Matlab/Simulink.

4.1.3 Sistema de controle

Segundo Dorf e Bishop (2009, p. 02) “um sistema de controle é a interconexão de componentes formando uma configuração de sistema que produzirá uma resposta desejada do sistema”.

Sistema de controle é quando se possui um sistema e ele controla a ele mesmo ou outro sistema, por exemplo, o controle de temperatura de uma caldeira de forma automática, sem precisar da interferência de um operador, é quando se deseja fazer um controle automático de alguma variável.

O sistema de controle é dividido em clássico e moderno como já foi explicado anteriormente, e em sistema de malha aberta (figura 53) e malha fechada (figura 54). A diferença entre esses dois tipos vem da realimentação da resposta, no caso da malha aberta, ele faz sua atuação e não confere a resposta, já o caso do malha fechada ele realiza a atuação e verifica se a resposta foi a desejada, caso não seja ele atua de novo até chegar ao valor desejado. Para entender melhor serão citados exemplos: um exemplo de malha aberta é a máquina de lavar roupa que realiza a lavagem da roupa, mas não verifica se ela realmente está limpa ou não, cabe ao usuário fazer esta análise e quanto a malha fechada podemos citar como exemplo o controle de nível automático de um tanque, que tem como função deixar dentro do nível certo, e para isto ele sempre faz a checagem para ver se tá dentro da faixa desejada, e caso não esteja ele faz a atuação até que entre dentro desta faixa.

No caso deste trabalho será utilizado o sistema de controle de malha fechada, em que se faz o controle da posição do carrinho e a atuação é o posicionamento do servomotor, e enquanto o carrinho não estiver no local desejado, o controle não para de atuar sobre o servomotor.



Figura 53 - Funcionamento de um controle de malha aberta

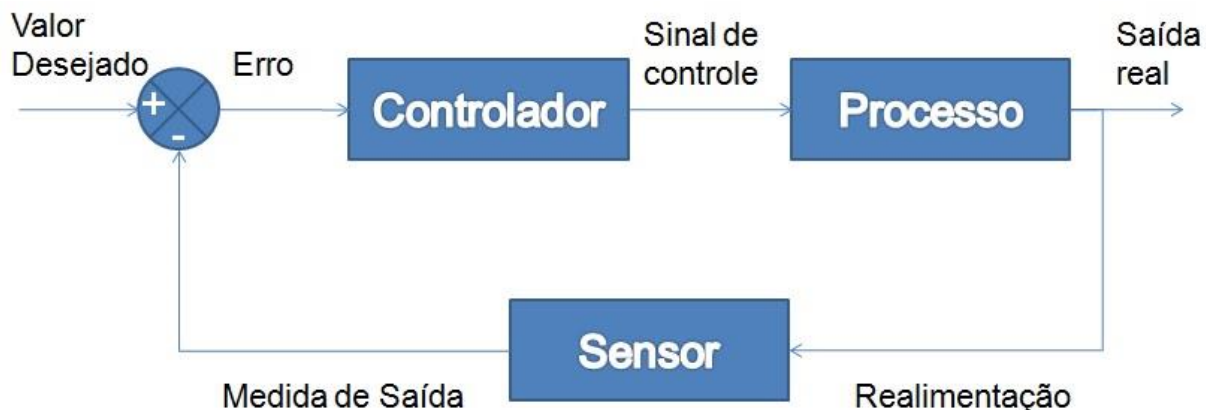


Figura 54 - Funcionamento de um controle em malha fechada

Fonte: Autoria Própria.

4.1.4 PID (Proporcional-Integral-Derivativo)

PID é um dos algoritmos de controle mais utilizados na indústria, por ser robusto e ter um bom desempenho, além de ser de fácil entendimento, tornando-se fácil de aplicar.

Primeiro cabe ressaltar que é um tipo de controle de malha fechada, que trabalha com a diferença do valor desejado (*offset*) e do valor real, assim como pode ser visto na figura 55. Como o próprio nome sugere este algoritmo trabalha com três coeficientes: o integral (K_i), o derivativo (K_d) e o proporcional (K_p), esses três parâmetros são variados até encontrar um valor ideal para o controle, ou seja, até que o sistema consiga responder de forma rápida e precisa.

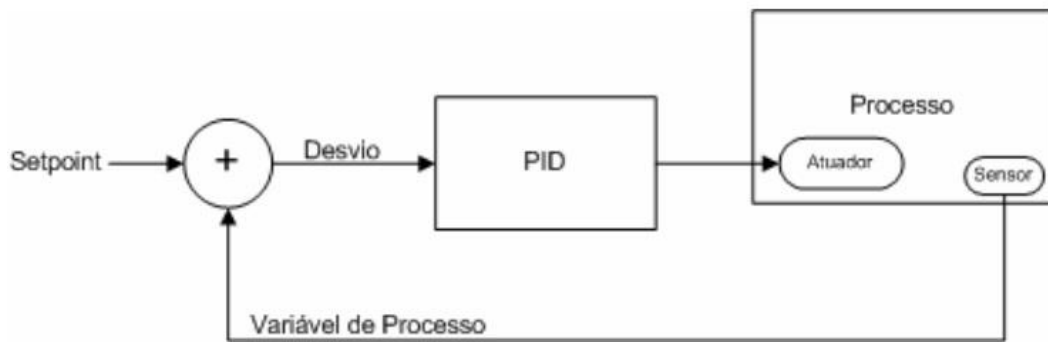


Figura 55 - Controle de malha fechada com PID
Fonte: Guerra (2009, P.12)

Lages (2010, p. 01) descreve uma das equações (1) – Expressão PID Contínuo – que podem representar este método de controle, onde $e(t)$ é o erro que consiste na diferença entre o valor lido e o valor desejado, $u(t)$ é o valor da saída do PID, t é o tempo correspondente e τ é uma variável de integração.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (1)$$

Para facilitar à aplicação do método a equação acima pode ser reescrita como uma Expressão PID contínuo com parâmetros discretizados (2) (LAGES, 2010, p. 01) onde T_i é o tempo integral e T_d o tempo derivativo, e $K_i = K_p/T_i$ e $K_d = K_p/T_d$.

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{d}{dt} e(t) \right) \quad (2)$$

A seguir, pode-se entender como cada parâmetro do PID influencia na resposta através do quadro 1 e também através dos gráficos abaixo. No gráfico 2 nota-se a aplicação apenas do fator proporcional; no gráfico 3 do proporcional e do derivativo e por ultimo, no gráfico 4, a aplicação dos três, finalizando assim o controle.

Parâmetro	Função
Proporcional (K_p)	<ul style="list-style-type: none"> Influencia no tempo de resposta Diminui o Erro estacionário
Integrativo (K_i)	<ul style="list-style-type: none"> Elimina o erro de regime
Derivativo (K_d)	<ul style="list-style-type: none"> Diminui overshoot Melhora a estabilidade do sistema

Quadro 1 - Influência dos fatores do PID

Fonte: Autoria Própria.

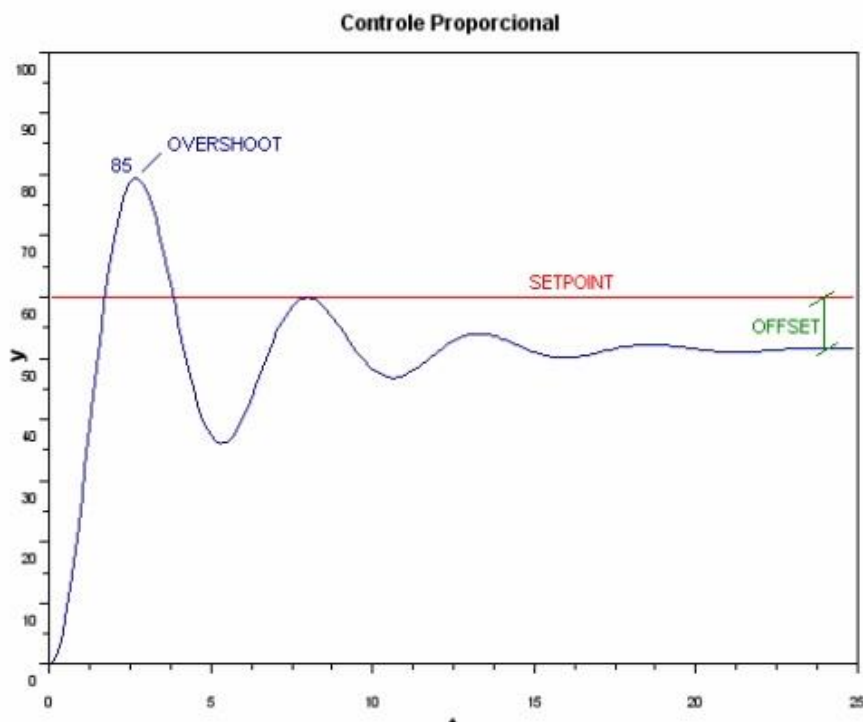


Gráfico 2 - Controle com apenas ganhos proporcional P

Fonte: Guerra (2009, P.13)

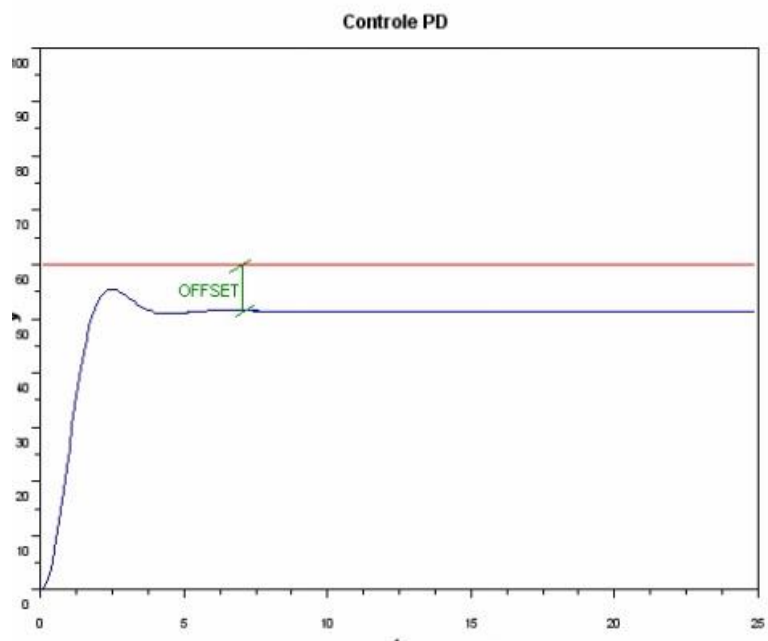


Gráfico 3 - Controle com ganho proporcional e derivativo (PD)
Fonte: Guerra (2009, P.14)

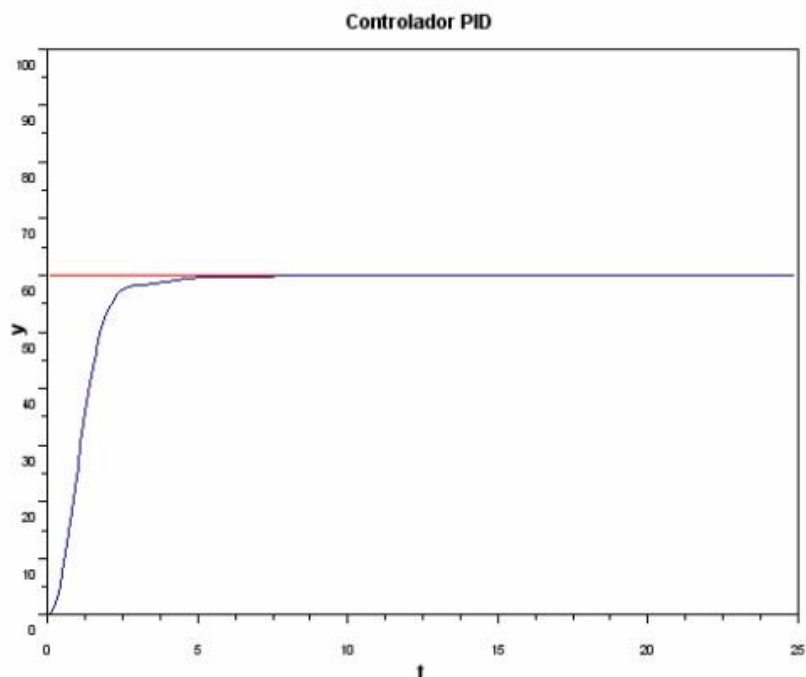


Gráfico 4 - Controle com ganho proporcional, derivativo e integral (PID)
Fonte: Guerra (2009, P.15)

4.1.5 Sistema Seguidor

Um sistema linearizado em malha aberta pode ser representado por equações de estado de n -ésima ordem e de saída de p -ésima ordem como segue abaixo:

$$\dot{x} = Ax + Bu \quad (1)$$

$$y = Cx = \begin{bmatrix} E \\ F \end{bmatrix} x \quad (2)$$

Onde y é um vetor $px1$ e $w=Ex$ é um vetor $mx1$ que representa as variáveis de saída que são necessárias para resultar em um vetor r . O controle por realimentação é necessário para o vetor de saída w siga o comando de entrada r para que a resposta do regime seja:

$$\lim_{t \rightarrow \infty} w(t) = r(t) \quad (3)$$

Quando r é um comando de entradas constante por partes. O método consiste na adição de um vetor comparador e integrador que satisfaça equação 4 abaixo, segundo D'Azzo e Houpis (1995 , p. 541).

$$\dot{z} = r - w = r - Ex \quad (4)$$

O sistema de malha aberta composto é então governado pelas equações aumentadas de estado e saída formadas a partir das equações (1) e (2):

$$\begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -E & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ I \end{bmatrix} r = \bar{A}x' + \bar{B}u + \bar{B}'r \quad (5)$$

$$y = [C \quad 0] \begin{bmatrix} x \\ z \end{bmatrix} = \bar{C}x' \quad (6)$$

Onde:

$$\bar{A} = \begin{bmatrix} A & 0 \\ -E & 0 \end{bmatrix}; \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}; \bar{B}' = \begin{bmatrix} 0 \\ I \end{bmatrix}; \bar{C} = [C \quad 0] \quad (7)$$

De acordo com D'Azzo e Houpis (1995, p. 542) a lei de controle a ser usada é:

$$u = K_1x + K_2z = [K_1 \quad K_2] \begin{bmatrix} x \\ z \end{bmatrix} \quad (8)$$

$$\bar{K} = [K_1 \quad K_2] \quad (9)$$

A representação do sistema de controle por realimentação em que estão contidas as equações de estado e de saída (equações (1) e (2)), e a lei de controle (equação (8)) pode ser visualizada na figura 56.

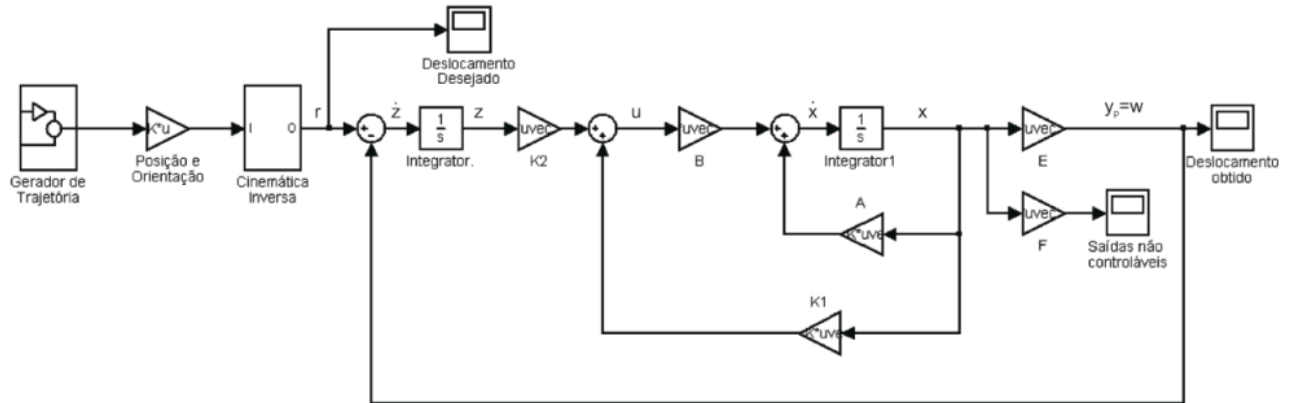


Figura 56 - Sistema de controle seguidor
Fonte: Autoria Própria.

Esta lei de controle atribui o espectro de autovalores de malha fechada, se e somente se, a planta aumentada e o par de matrizes de controle (\bar{A}, \bar{B}) são controláveis (D'AZZO; HOUPIS, 1995). Esta condição é satisfeita se (A, B) for um par controlável e o

$$\text{posto de } \begin{bmatrix} B & A \\ 0 & -E \end{bmatrix} = n + m \quad (10)$$

Para que (A, B) seja controlável é necessário que satisfaça a condição de controlabilidade (equação 11).

$$\text{posto } M_c = \text{posto}[B \ AB \ A^2B \ \dots \ A^{n-m}B] = n \quad (11)$$

As condições das equações (10) e (11) garantem que a lei de controle pode ser sintetizada tal que a saída da malha fechada w segue o comando de entrada r . E neste caso a equação de malha fechada é:

$$\dot{x}' = \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A + BK_1 & BK_2 \\ -E & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} r = A'_{cl} x' + B' r \quad (12)$$

A matriz de realimentação deve ser selecionada tal que os autovalores da matriz da planta de malha fechada estejam todos no semi-plano complexo esquerdo.

A obtenção da matriz \bar{K} é realizada a partir da seleção dos autovalores a serem atribuídos à matriz da planta de malha fechada A'_{cl} na equação (12).

$$[\bar{A} + \bar{B}\bar{K}]v_i = \lambda_i v_i \quad (13)$$

Que pode ser colocada na forma

$$[\bar{A} - \lambda_i I \quad \bar{B}] \begin{bmatrix} v_i \\ g_i \end{bmatrix} = 0 \text{ para } i = 1, 2, \dots, n + m \quad (14)$$

Onde v_i é o autovetor e

$$g_i = \bar{K}v_i \quad (15)$$

Para satisfazer a equação (14), o vetor $[v_i^T \quad g_i^T]^T$ deve pertencer ao *kernel* de

$$\bar{S}(\lambda_i) = [\bar{A} - \lambda_i I \quad \bar{B}] \text{ para } i = 1, 2, \dots, n + m \quad (15)$$

A notação $\ker \bar{S}(\lambda_i)$ é usada para definir o espaço chamado de nulo que contém todos os vetores $[v_i^T \quad g_i^T]^T$ para a equação (14) seja satisfeita (D'AZZO; HOUPIS, 1998). A equação (15) pode ser usada para formar a igualdade matricial

$$[g_1 \quad g_2 \quad \dots \quad g_{n+m}] = [\bar{K}v_1 \quad \bar{K}v_2 \quad \dots \quad \bar{K}v_{n+m}] \quad (16)$$

Donde se obtém a matriz \bar{K} como segue:

$$\bar{K} = [g_1 \quad g_2 \quad \dots \quad g_{n+m}][v_1 \quad v_2 \quad \dots \quad v_{n+m}]^{-1} = QV^{-1} \quad (17)$$

Observa-se que os autovalores podem ter valores repetidos de número igual às entradas do sistema. Isso se deve ao fato do espaço nulo ter dimensão igual ao número de entradas. Assim associa-se um autovalor repetido a um vetor da base do espaço nulo. Com isso, todas as colunas da matriz V continuam sendo linearmente independentes, e por isso a matriz V^{-1} existe.

4.2 APLICAÇÃO DO MÉTODO DE CONTROLE PID

Após toda a construção do mecanismo, e feitas as ligações necessárias, deu-se início a aplicação do método de controle PID. Para isso foi utilizado o *software* Labview, aonde foi feito um código para receber os dados do *encoder* via serial (visa) e então esse dado alimentar o bloco de controle PID o qual realiza o controle e através da diferença do valor desejado (*offset*) e o valor real lido, gera o sinal de controle que sai através da porta analógica da placa de aquisição. Segue abaixo o diagrama de blocos na figura 57 onde pode se ter uma visão geral, e as figuras 58 e 59 onde o programa da figura 57 foi dividido em dois para ter uma visão um pouco mais aproximada e o painel frontal na figura 60 onde se tem uma visão geral.

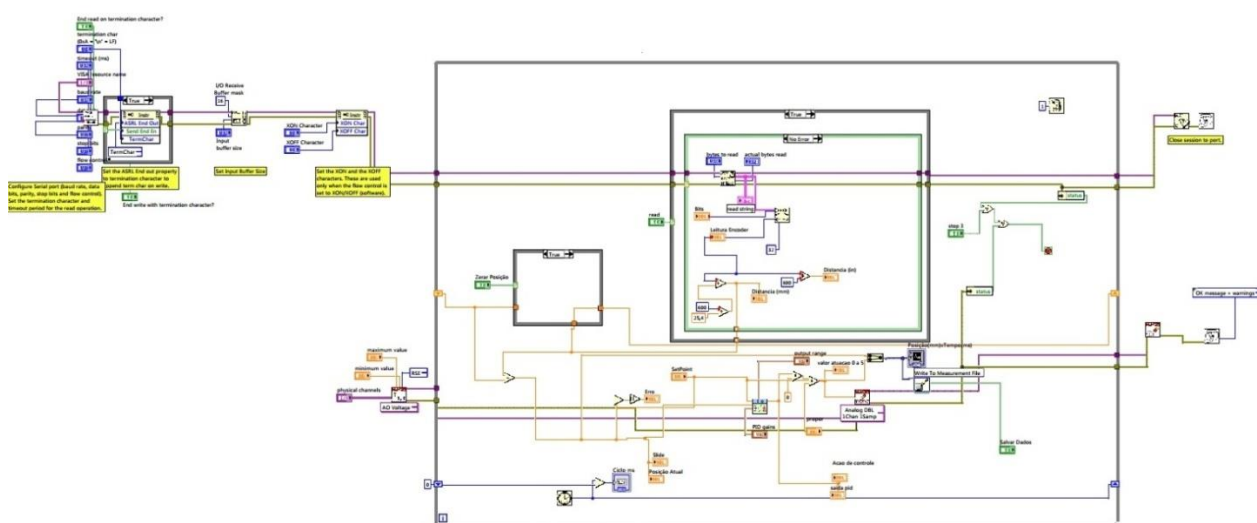


Figura 57 - Diagrama de blocos (visão geral)
Fonte: Autoria Própria.

Block Diagram

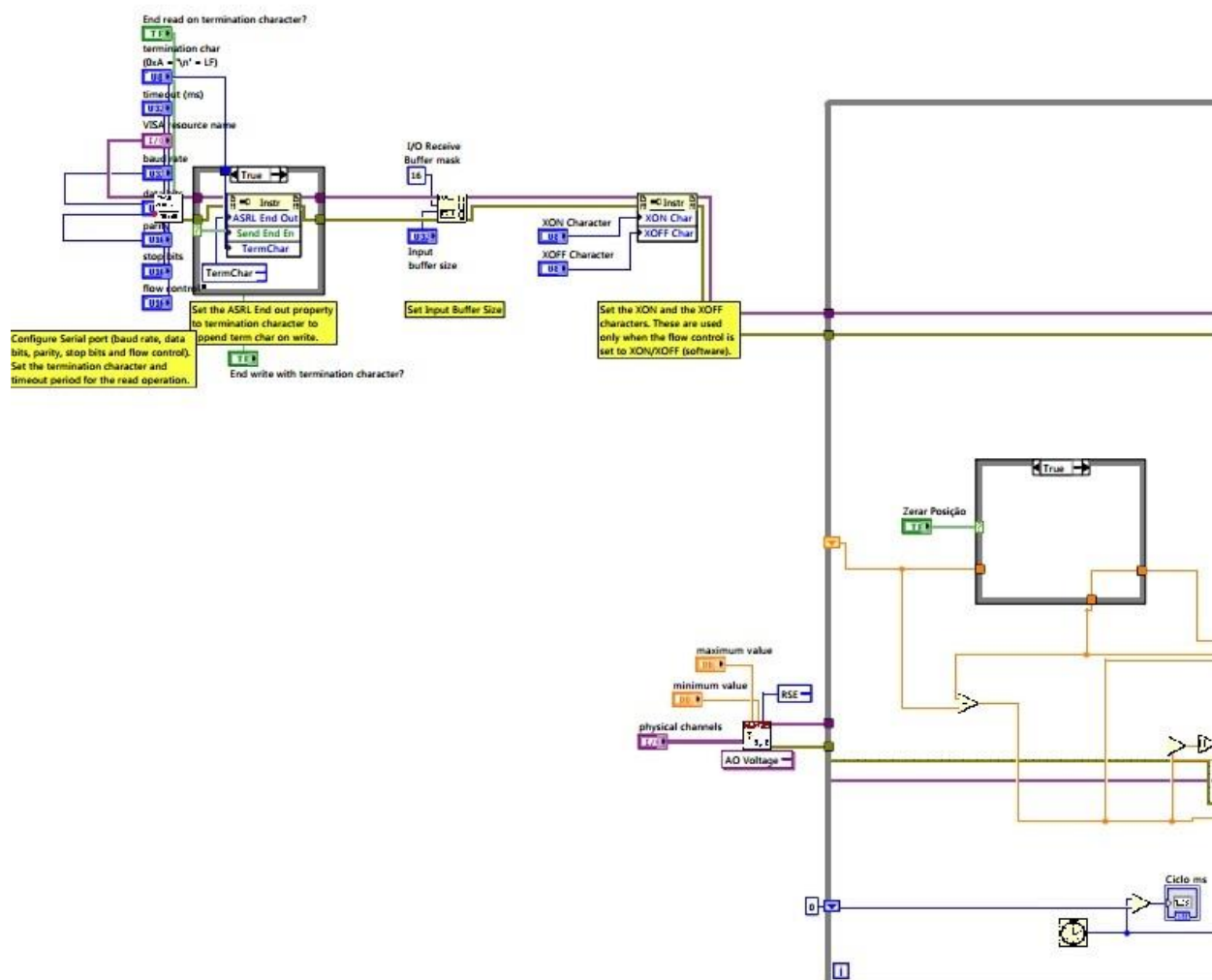


Figura 58 - Diagrama de blocos dividida em dois (parte esquerda)
 Fonte: Autoria Própria.

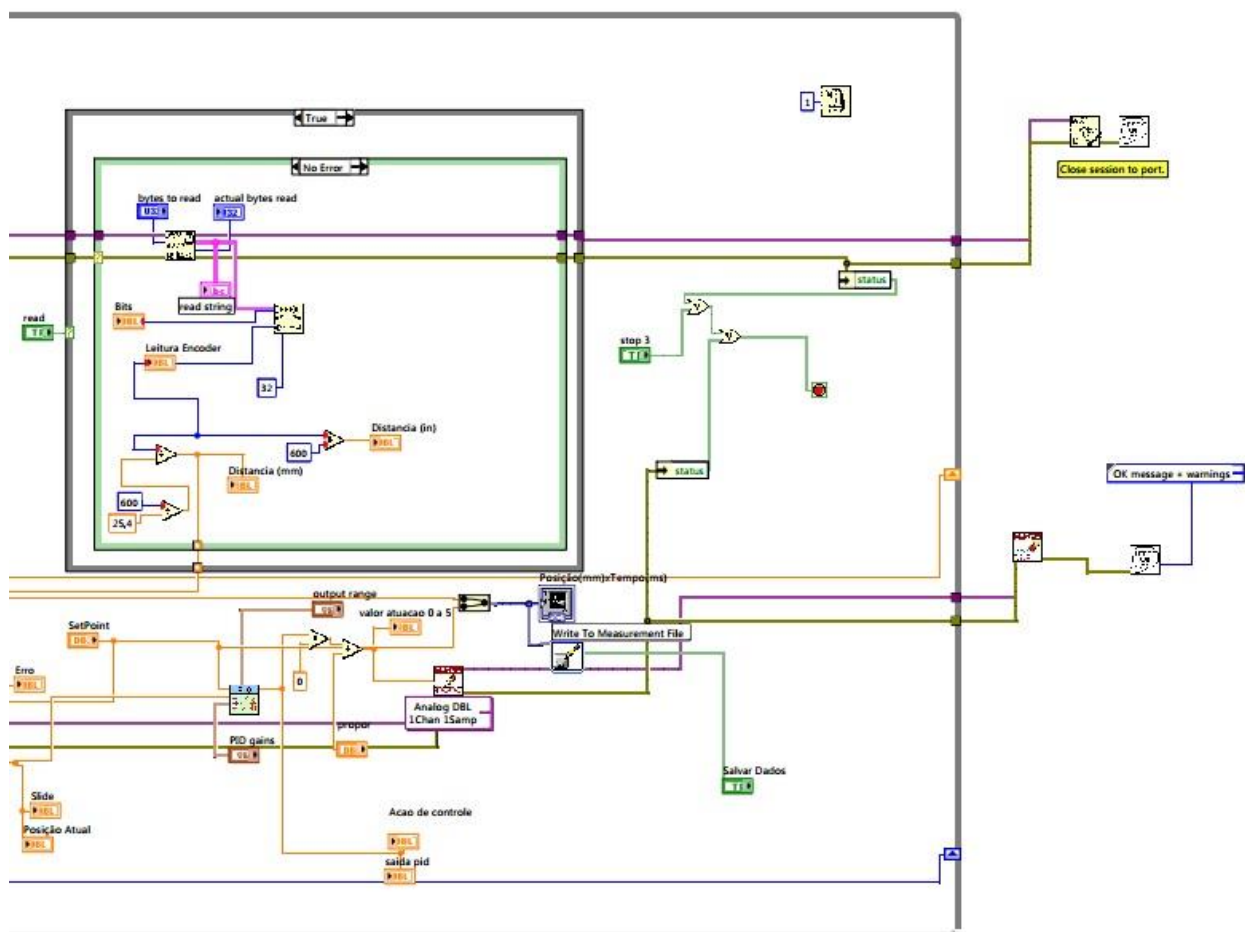


Figura 59 - Diagrama de blocos dividida em dois (parte direita)
 Fonte: Autoria Própria.

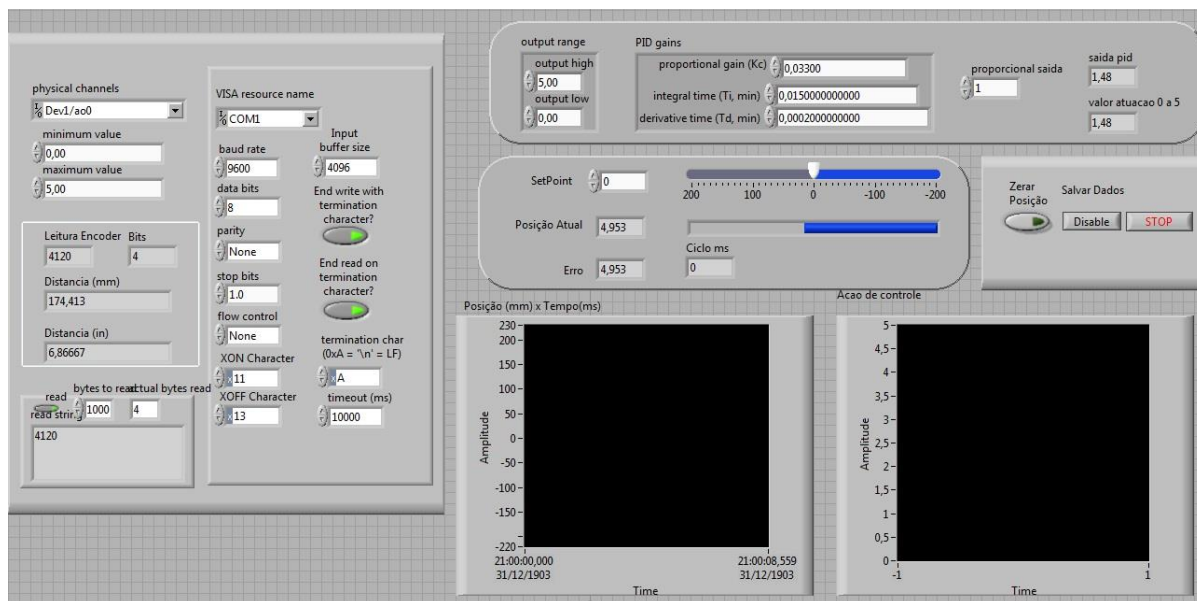


Figura 60 - Painel frontal (visão geral)
Fonte: Autoria Própria.

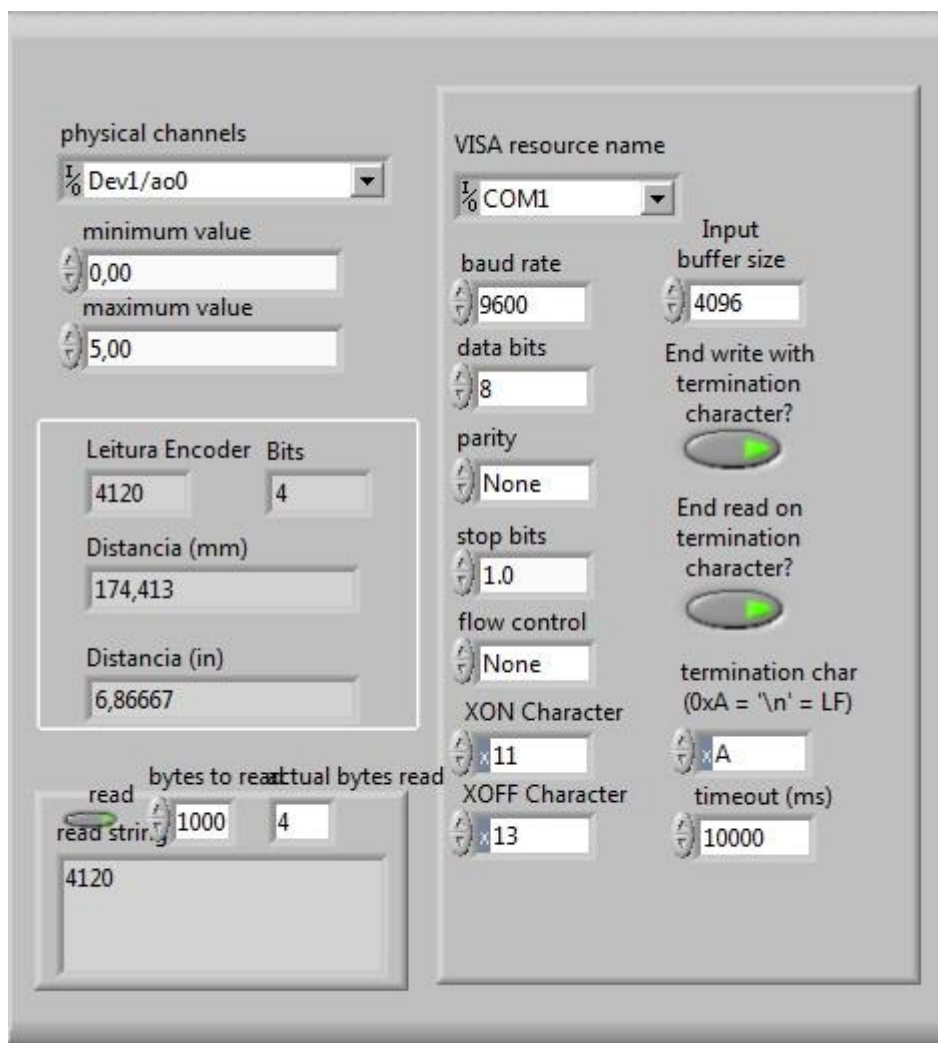


Figura 61 - Painel Frontal (Aquisição de dados)
Fonte: Autoria Própria.

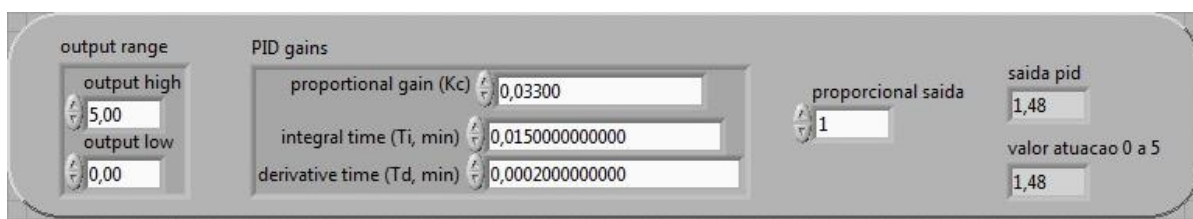


Figura 62 - Painel frontal (PID)
Fonte: Autoria Própria.

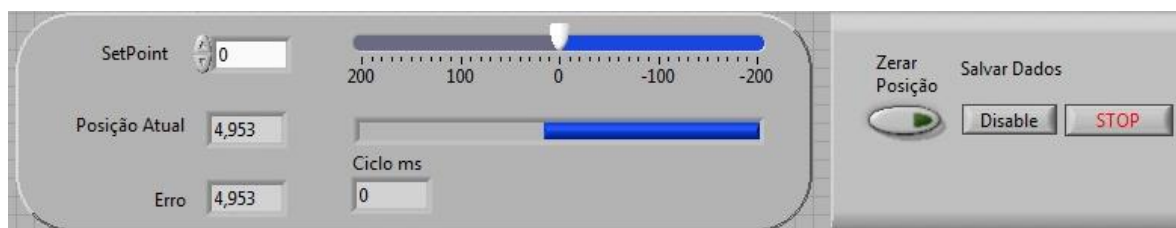


Figura 63 - Painel Frontal (Setpoint, Posição, reset do zero, gravação de dados)
Fonte: Autoria Própria.

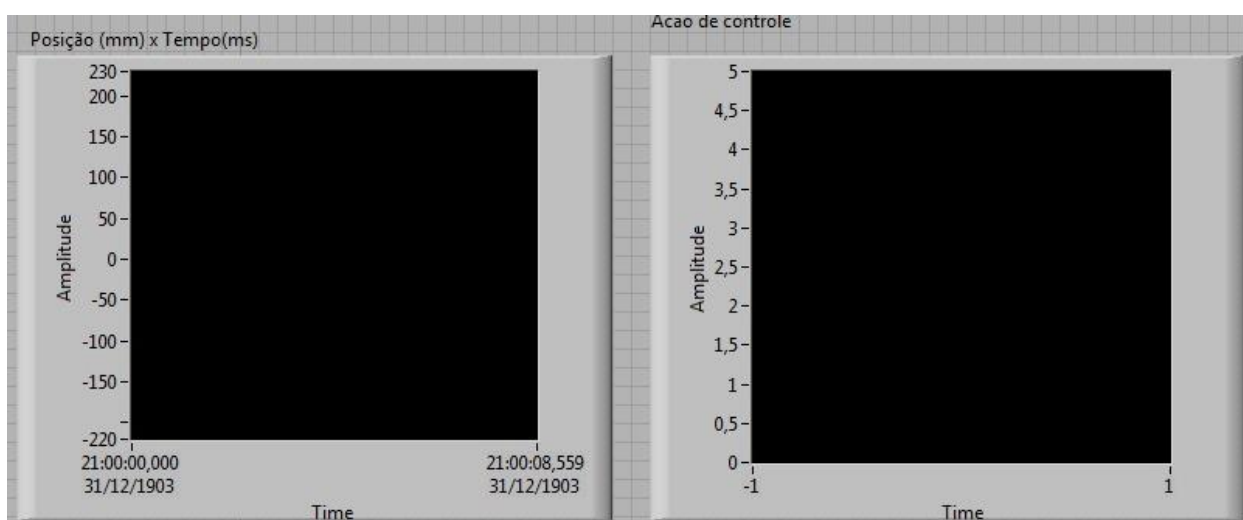


Figura 64 - Painel frontal (Gráfico de posição e ação de controle)
Fonte: Autoria Própria.

A figura 61 acima mostra a parte do painel frontal que corresponde àquela que recebe e transmite os dados. Nela estão presentes a seleção das portas: analógica e serial (visa), os parâmetros que devem ser ajustados para melhor transmissão de dados serial e também a leitura do encoder em milímetros, polegada e unidade de medida própria do encoder.

Já a figura 62 consta a parte com o sistema de controle, no caso o PID, neste bloco são ajustados os ganhos K_p , T_i e T_d , o valor máximo e mínimo da saída (volts), mostra o valor que está saindo do controle e também possui um parâmetro para amortizar a ação de controle, dividindo-a por um valor escolhido.

O bloco correspondente à figura 63 tem a função de mostrar a posição em que se encontra o carrinho (leitura do *encoder*) e o erro, relativo ao *setpoint* e a posição real, nele também pode ser encontrado onde settar o setpoint, além de

possuir três botões: um para parar o programa, outro para ativar o armazenamento dos dados do programa para um arquivo externo e por ultimo um que tem a função de zerar a posição do carrinho na posição desejada, já que o *encoder* não zera sua posição ao parar o programa.

E por ultimo, a figura 64 é onde pode ser observado melhor como está atuando o controle através de dois gráficos, o primeiro mostrando a posição do carrinho (mm) e o *setpoint* (em vermelho) e o outro mostrando o sinal que está saindo do PID, ambos em relação ao tempo (ms).

Construído o código em *Labview*, e todos os parâmetros funcionando corretamente foi ligado então o mecanismo e colocado em prática o controle. No começo o resultado foi um pouco problemático, primeiro porque havia um problema com a comunicação onde a taxa de amostragem estava muito rápida e o programa não estava acompanhando, o que gerava ruídos e atrapalhava na hora de coletar os dados e conseqüentemente o controle, tal problema foi resolvido colocando-se um *delay* na programação do primeiro dsPIC (o que realiza a leitura do *encoder*) e então o problema foi resolvido e depois o problema estava no ajuste dos parâmetros do PID, ou seja encontrar valores heurísticamente que fornecessem um controle relativamente bom, com uma resposta rápida o suficiente e com boa precisão, porém após algumas tentativas tais valores foram encontrados, os mesmo podem ser observados na figura 65 abaixo e nos gráficos 5 e 6 pode-se observar a resposta do controle quando variado o *offset* para -150mm e 150mm (tais valores de *offset* foram escolhidos para que se pudesse observar o comportamento do controle quando houvesse uma mudança brusca no valor desejado).

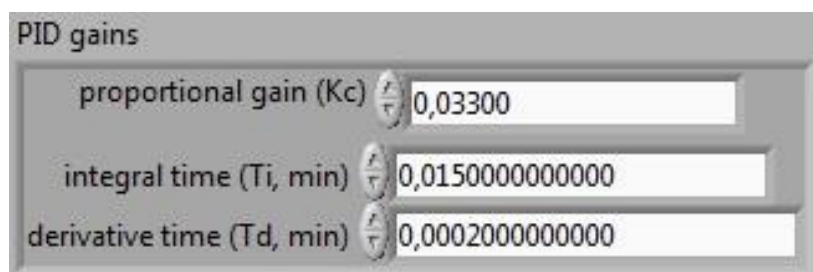


Figura 65 - Parâmetros encontrados do PID
Fonte: Autoria Própria.

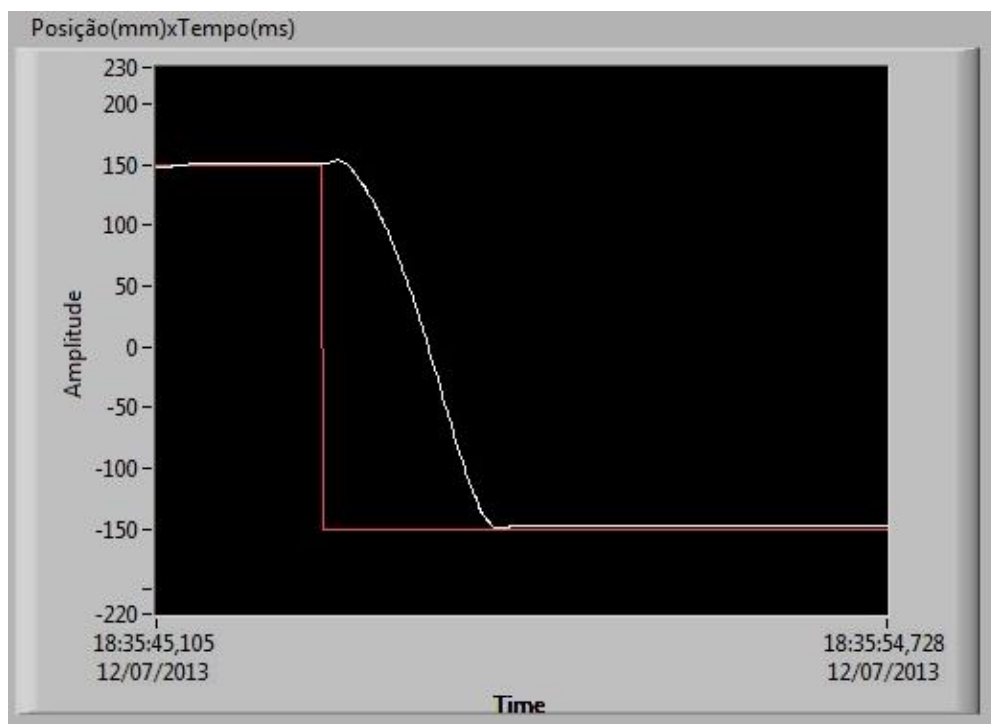


Gráfico 5 - Posição do carrinho com *setpoint* alterado de 150 para -150 mm
Fonte: Autoria Própria.

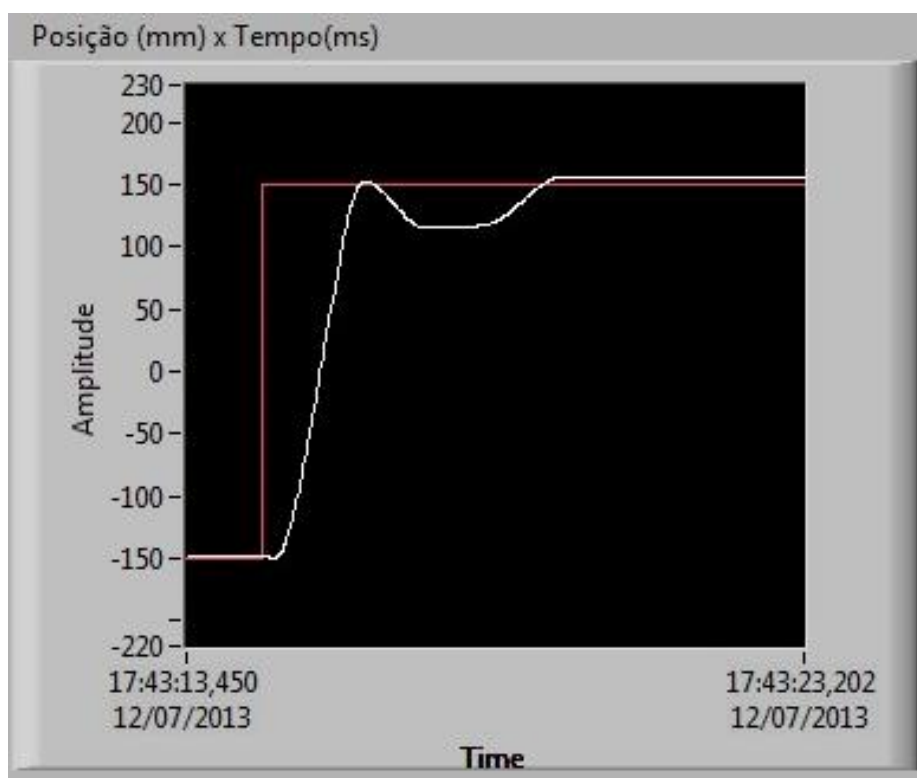


Gráfico 6 - Posição do carrinho com *setpoint* alterado de -150 para 150 mm
Fonte: Autoria Própria.

4.3 MODELAGEM ADAMS E SIMULAÇÃO SISTEMA SEGUIDOR

Mesmo com o trabalho tendo atingido seu objetivo que era construir o mecanismo, sincronizá-lo com o computador e se possível aplicar um método de controle para testá-lo, resolveu-se dar procedência e simular mais um método de controle para poder compará-lo com o já aplicado.

O método escolhido foi o de sistema seguidor, tal método foi aplicado da seguinte forma: primeiro foi gerado o modelo que seria uma versão simplificada da planta no *software Adams* (figuras 66 e 67) e depois no mesmo programa foi realizado a simulação dinâmica, a qual gerou as matrizes de estado A, B, C e D, as quais podem ser encontradas no anexo D. As matrizes de estado foram então importadas para um código feito em *Matlab/simulink* (o código usado foi obtido da tese de mestrado do Professor Doutor Marcio Aurelio Montezuma e foi modificado para ser utilizado neste projeto, o mesmo está no anexo C) e através deste código foi gerada a simulação do sistema seguidor o qual pode ser visto nos gráficos 6 e 7, as quais foram feitas com alguns *offset* de 150mm e -150mm respectivamente.

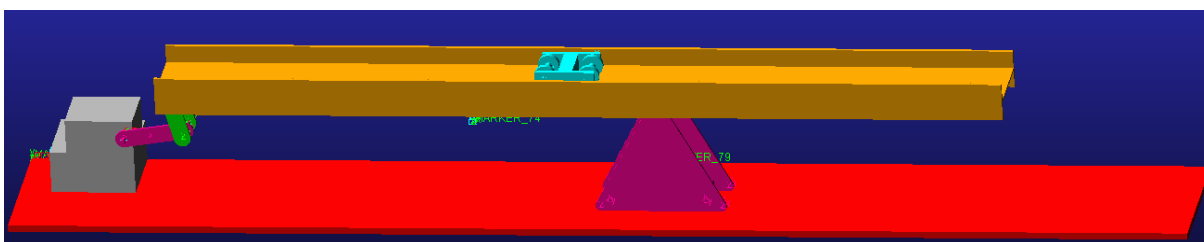


Figura 66 - Modelo Simplificado feito no *software Adams*
Fonte: Autoria Própria.

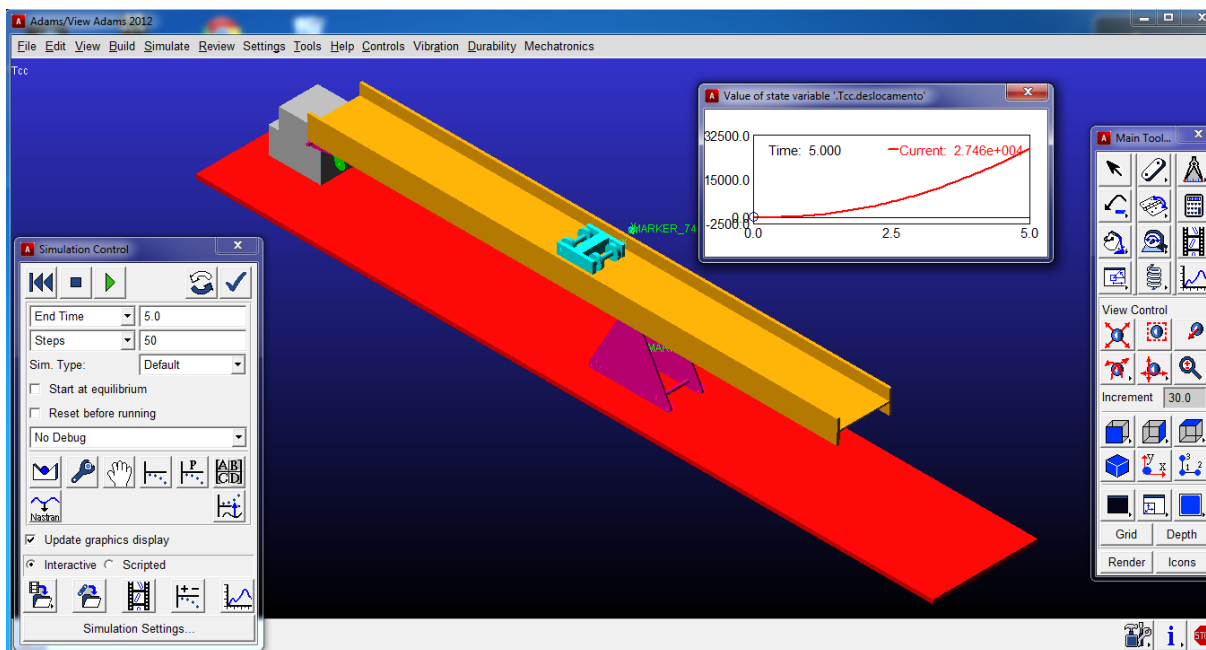


Figura 67 - Modelo simplificado feito em software Adams
Fonte: Autoria Própria.

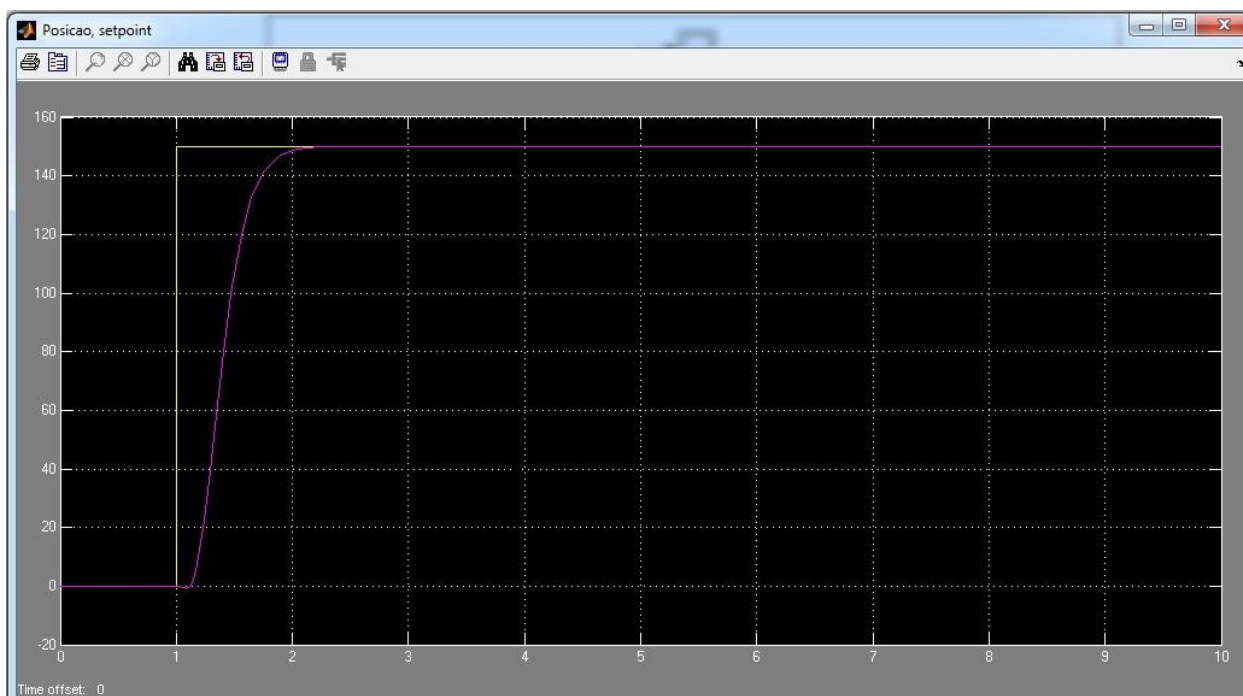


Gráfico 7 - Ação de controle com Setpoint alterado de 0 para 150 mm
Fonte: Autoria Própria.

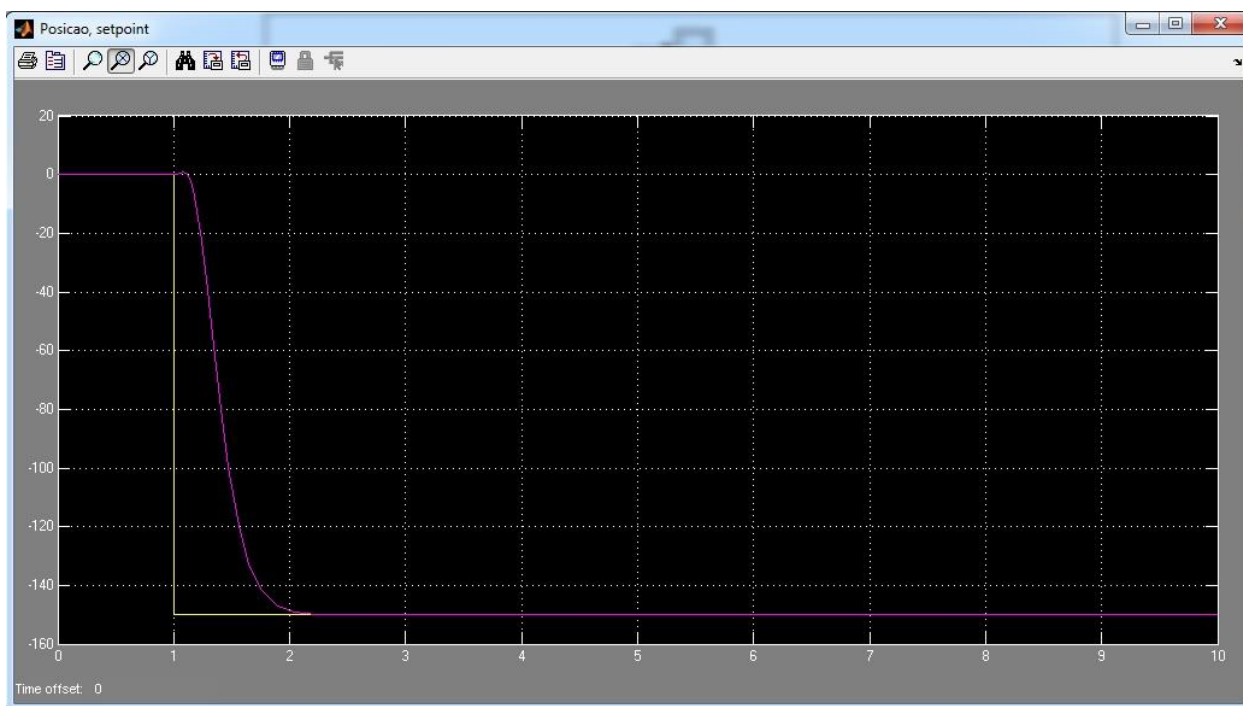


Gráfico 8 - Ação de controle com *Setpoint* alterado de 0 para -150 mm
Fonte: Autoria Própria.

5 CONCLUSÃO

Pode-se concluir que o trabalho em questão sucedeu-se como o planejado, ou seja, atingiu seus objetivos, que eram a construção de um mecanimo Ball and Beam com materiais de baixo custo, e podemos observar que isso se sucedeu, já que o encoder, seu suporte, e o suporte da sua fita vieram de uma impressora que estava inutilizada, outro fator que foi atingido é que ele foi feito de uma maneira inovadora, pois ao invés de uma esfera utilizou-se um carrinho, além do que foi utilizado uma comunicação via *wireless* e como sensor de posição encoder, método o qual em revisões bibliográficas não foi encontrado.

Também ressalta-se o sucesso da sincronização do mesmo com um computador, e a implementação de um método de controle, que apesar de algumas adversidades respondeu de forma adequada, com um erro baixo, o que é aceitável já que o projeto não exige uma grande precisão, e também com um tempo de resposta formidável. Além também de que o projeto foi além do planejado, que foi feito uma simulação de um método de controle por sistema seguidor, o qual por análise dos gráficos 6 e 7, pode-se observar que respondeu com uma precisão um pouco maior do que o PID.

Então de um forma geral o projeto foi um sucesso, cabendo salientar que com algumas mudanças pode-se melhorar, então recomenda-se em trabalhos futuros, primeiro aplicar uma melhoria na comunicação via *wireless* para diminuição ainda maior do ruído para torná-lo se não zero, pelo menos o mais perto, e também a implantação de forma pratica do sistema seguidor e de outros métodos de controle, tantos modernos quanto clássicos para poder fazer uma análise de qual responde melhor com a planta, além de poder assim entender melhor como funciona cada método e podendo assim ter uma facilidade de escolha quando for preciso o uso de algum deles.

Outras mudanças que podem ser aplicadas são: a construção de um dispositivo que não permita que o carrinho descarrile do trilho e também a colocação de um final de curso, e também o teste de outros meios de comunicação tanto *wireless* (*Bluetooth*, *xbee*, ou outros) como por fio mesmo, ligando direito o *encoder* com a placa de aquisição e verificando assim as possíveis melhoras que possam ocorrer.

REFERENCIAS

ANDREA, Cristiano Quevedo. **Aula 10.** Departamento Acadêmico de Eletrotécnica- Universidade Tecnológica Federal do Paraná, Curitiba, 2012.

BEZERRA, Romildo Martins. **Sinais.** Disponível em:< <http://www.ifba.edu.br/professores/romildo/downloads/ifba/sinais.pdf>>. Acesso 31 Jul. 2013.

CHOI, Chiu; DORSETT, Arend; EVANKO, David. **A Ball-on-Beam System with an Embedded Controller.** *Department of Electrical Engineering- University of North Florida.* Disponível em: <<http://www.asee.org/documents/sections/middle-atlantic/spring-2008/02-A-Ball-on-Beam-System-with-an-Embedded-Controller.pdf>>. Acesso em: 31 Jul. 2013.

D'AZZO, John Joachim; HOUPIS, Constantine H. **Análise e projeto de sistemas de controle lineares.** Rio de Janeiro: Guanabara Dois, 1995. 610p.

DORF, Richard C.; BISHOP, Robert H. **Sistemas de controle modernos.** 11. ed. Rio de Janeiro, RJ: LTC, 2009. 724 p.

EBAY. **2 X Wireless Serial RS232 Port Adapter Module Freescale.** Disponível em:< http://www.ebay.com/itm/2-X-Wireless-Serial-RS232-Port-Adapter-Module-Freescale-/260769348505?pt=LH_DefaultDomain_0&hash=item3cb7107b99#ht_7421wt_1037 > Acesso em: 31 Jul. 2013.

EBLDC. **UNDERSTANDING PWM.** DISPONÍVEL EM: <[HTTP://EBLDC.COM/?P=48](http://EBLDC.COM/?P=48)>. ACESSO EM: 31 JUL. 2013.

ELETRONICA. **Regulador LM 7805.** Disponível em: <<http://www2.eletronica.org/hack-s-dicas/regulador-lm7805/>>. Acesso em: 31 Jul. 2013.

GUERRA, Wladimir de Andrade. **Implementação de Controle Proporcional, Integral e Derivativo em Controladores Lógico Programáveis.** 2009. Monografia- Centro de Tecnologia e Geociências da Universidade Federal de Pernambuco.

Disponível em: <
http://www2.ee.ufpe.br/instrumentacao/monografias/Wladimir_Guerra_PROMINP_I.pdf>. Acesso em: 31 Jul. 2013.

IMAGESCO. **Activating Nitinol Wire Pulse Width Modulated (PWM)**. Disponível em:< <http://www.imagesco.com/articles/nitinol/07.html>>. Acesso em: 31 Jul. 2013.

JAHROMI, Ali Fellah; KESHMIRI, Mohammad; AMOOZGAR, Mohammad Hadi; MOHEBBI, Abolfazl; XIE, Wen-Fang. **Modeling and control of ball and beam system using model based and non-model based control approaches**. 2012. International journal on smart sensing and intelligent systems, vol. 5, no. 1. Disponível em:< <http://www.s2is.org/Issues/v5/n1/papers/paper2.pdf>>. Acesso em: 31 Jul 2013.

LAGES, Walter Fetter. **Controladores PID**. Departamento de Engenharia Elétrica- Universidade Federal do Rio Grande do Sul, 2010.

LETS MAKE ROBOTS. **How to use a quadrature encoder**. Disponível em:< <http://letsmakerobots.com/node/24031>> Acesso em: 31 Jul. 2013.

MATIAS, Juliano. **Encoders. Mecatrônica Atual**. n.3, Abril, 2002.

MONTEZUMA, Marcio Aurelio Furtado. **Metodologia para Identificação e Controle de um Protótipo de uma Plataforma de Movimento com 2 G.D.L.** Tese (Doutorado em Engenharia Mecânica) - Escola de Engenharia de São Paulo - Universidade de São Paulo, São Carlos, 2010.

NATIONAL INSTRUMENTS CORPORATION. **Explicando a Teoria PID**. Disponível em:< <http://www.ni.com/white-paper/3782/pt/>>. Acesso em: 31 Jul. 2013.

NOVUS. **Controle PID Básico**. Disponível em:< http://www.novus.com.br/site/default.asp?TroncoID=053663&SecaoID=0&SubsecaoID=0&Template=../artigosnoticias/user_exibir.asp&ID=726184>. Acesso em 31 Jul. 2013.

OGATA, Katsuhiko. **Engenharia de controle moderno**. 3. ed. Rio de Janeiro: Prentice-Hall, c1998. xiv, 813 p.

QSL. **Bascom and AVR, Encoders.** Disponível em:<
<http://www.qsl.net/p/pa3ckr//bascom%20and%20avr/encoders/>> Acesso em: 31 Jul.
2013.

QUANSER. **Ball and Beam.** Disponível em:<
http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R2-ballbeam&tmpl=1>. Acesso em: 07
Ago. 2013.

ROBOTGEAR. **Voltage Regulator – 5V(LM7805).** Disponível em:<
<http://www.robotgear.com.au/Product.aspx/Details/397-Voltage-Regulator-5V-LM7805>>. Acesso em: 31 Jul 2013.

SHIMADA, Bruno Masaharu. **Desenvolvimento de um kit didático baseado no controlador de sinais digitais dspic30f2010a da microchip.** Trabalho de Conclusão de Curso (Graduação em Engenharia Mecânica)-Universidade Tecnológica Federal do Paraná. Cornélio Proκόpio, 2011.

SCHVARCZ, Alyson F.; DINIZ, Ivando S. **Modelagem, simulação e controle de um sistema barra e bola auxiliado por computador: CAD e CAE.** XVIII Congresso Brasileiro de Automática, Bonito, MS, 2010.

SMITH, Nenuco. **Sinais digitais e sinais analógicos.** Disponível em:<
http://djnenuco.blogspot.com.br/2013_04_01_archive.html>. Acesso em: 31 Jul.
2013.

STORE HOBBIES. **Balanceador de Pás de Helicóptero R/C.** Disponível em:<
<http://storehobbies.com.br/loja/balanceadordepasdehelicopterorc-p-1167.html?osCsid=e37ded643096b432ad188138828e4004>>. Acesso em: 31 Jul.
2013.

WANG, Wei. **Control of a Ball and Beam System.** 2007. Degree of Advanced Master. School of Mechanical Engineering-The University of Adelaide, Austrália, 2007.

WIKIPÉDIA. **Servomotor.** Disponível em: < <http://pt.wikipedia.org/wiki/Servomotor>>.
Acesso em: 31 Jul 2013.

ZHONG, Jinghua. ***PID Controller Tuning: A short Tutorial.*** Department of Mechanical Engineering-Purdue University, 2006.

ANEXOS

ANEXO A - Programa rs232 dspic

```

/////////////////////////////////////////////////////////////////

// PROGRAMA DO CARRINHO WIRELESS

// LÊ O ENCODER DO CARRINHO E DEPOIS ENVIA
PARA O SERIAL WIRELESS

/////////////////////////////////////////////////////////////////

#include <30F2010.h>

#fuses HS,NOWDT,NOBROWNOUT ,NOMCLR

#use delay(clock=8000000)

#use rs232(baud=9600,xmit=PIN_C14,rcv=PIN_C13)

//////////////////Variaveis Globais ////////////////////

int tempo=0;

int16 valor_encoder=0;

void main() {

    //////////////////Configuração Encoder e Timer para Calculo de Velocidade, Taxa de
amostragem 50Hz////////////////////////////////////

    setup_qei(QEI_MODE_X4 |QEI_TIMER_INTERNAL);

    qei_set_count(0);

    delay_ms(10);

    while (TRUE)

        {

```

```
        valor_encoder = qei_get_count();  
    printf("%d\n",valor_encoder);  
        delay_ms(3);  
    }  
}
```


ANEXO B - Programa que gera sinal pwm rc

```
/*/////////////////////////////////////////////////////////////////
```

PROGRAMA PWM

Objetivo: Gera Sinal PwM RC para cristal de

```
////////////////////////////////////////////////////////////////*/
```

```
#include <30F2010.h>
```

```
#use delay (clock=12000000)
```

```
#fuses HS,NOWDT,BROWNOUT,NOMCLR
```

```
void main()
```

```
{
```

```
    unsigned int32 valor;
```

```
    unsigned int32 valor2;
```

```
    setup_timer2(TMR_INTERNAL | TMR_DIV_BY_1,59999); //Configura Timer 2 como
    interno e prescaler 64, e Timer2 máximo
```

```
    setup_compare(1, COMPARE_PWM | COMPARE_TIMER2); // Comparador 1,
    modo PWM , base de tempo TIMER2
```

```
    setup_compare(2, COMPARE_PWM | COMPARE_TIMER2); // Comparador 1,
    modo PWM , base de tempo TIMER2
```

```
while(true)
```

```
{
```

```
    ///Lê porta analógica
```

```
        setup_adc_ports(sAN0);
```

```
        setup_adc(ADC_CLOCK_DIV_2);
```

```
        set_adc_channel(0);
```

```
            delay_us(5);
```

```
        valor=read_adc();
```

```
        setup_adc(ADC_OFF);  
valor=0.045903784*valor+2999.95;  
set_pwm_duty(1,valor); //metade do servo  
    }  
}
```

ANEXO C - Sistema seguidor matlab

Código em *Matlab*

```

clear all
clc
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%           EESC-USP / Mestrado em Engenharia Mecânica           /
% Programa Desenvolvido para Determinação dos Ganhos K1 e K2     /
% de um Sistemas Seguidores Usando Realimentação de Estados     /
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%load variaveis ;
% Matrizes A, B, C, D referentes a dinamica da Plataforma de Stewart

% %-----
%
% % n representa o grau do sistema no caso do pêndulo invertido é um
% sistema
% % de 4 grau
% %-----
%
n = 4;
% %-----
%
% % Mostra as matrizes A,B,C
% %-----
%
a=load ('finala')
b=load ('finalb')
c=load ('finalc')
d=load ('finald')
%-----
% Condição de controlabilidade de controlabilidade
%-----
Matriz_M= zeros(n);
Matriz_M(:,1)= b;
Matriz_M(:,2)= a*b;
Matriz_M(:,3)= (a*a)*b;
Matriz_M(:,4)= ((a*a)*a)*b;
Matriz_M;
rank(Matriz_M);
if n ~= rank(Matriz_M)
    error('O sistema não é controlavel');
end
Mcc=[b a;d -c];
m = size(Mcc);
rank(Mcc);
if m ~= rank(Mcc)
    error('O sistema não é controlavel');
end
%-----
%format long
%-----
% Montagem das matrizes Abarra e Bbarra , cria a matriz e para completar os
% valores de abar, para que a matriz seja quadrada
%-----
e=zeros(4,1);
abar=[a e;-c d];
bbar=[b;d];

```

```

cbar=[c d];
%-----

%-----
% Conjunto de autovalores a serem atribuidos ao sistema 6x
%-----
l_1=10;
l_2=11;
l_3=12;
l_4=13;
l_5=14;
%-----

%-----
%-----

s1=[abar+l_1*eye(5) bbar];
s2=[abar+l_2*eye(5) bbar];
s3=[abar+l_3*eye(5) bbar];
s4=[abar+l_4*eye(5) bbar];
s5=[abar+l_5*eye(5) bbar];
%-----

%-----
%-----

r1=rref(s1);
r2=rref(s2);
r3=rref(s3);
r4=rref(s4);
r5=rref(s5);
%-----

%-----
%-----

a1=[r1;zeros(1,6)];
a2=[r2;zeros(1,6)];
a3=[r3;zeros(1,6)];
a4=[r4;zeros(1,6)];
a5=[r5;zeros(1,6)];
%-----

%-----
%-----

a1(6,6)=-1;
a2(6,6)=-1;
a3(6,6)=-1;
a4(6,6)=-1;
a5(6,6)=-1;
%-----

%-----
%-----

v(:,1)=a1(1:5,6);
v(:,2)=a2(1:5,6);
v(:,3)=a3(1:5,6);
v(:,4)=a4(1:5,6);
v(:,5)=a5(1:5,6);
%-----

```

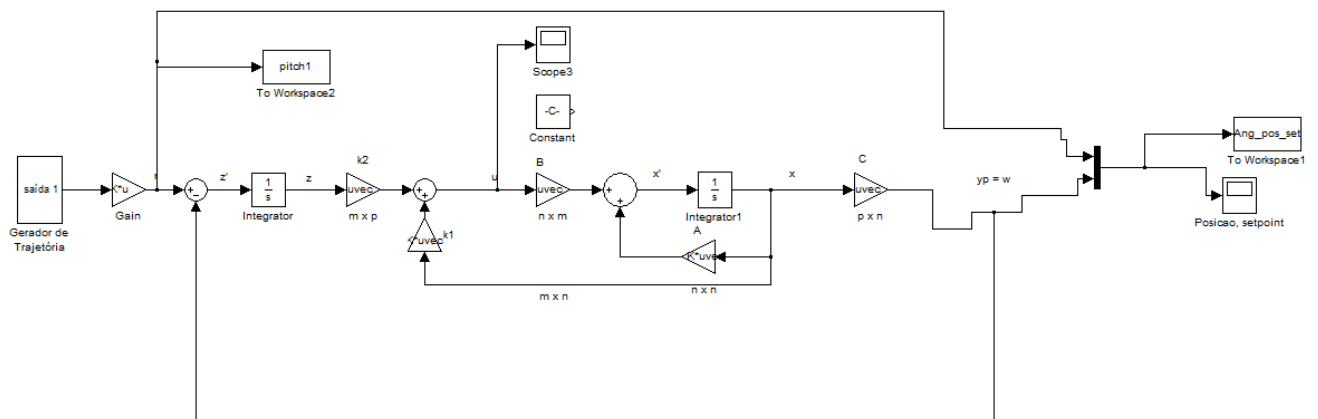
```

% -----
% -----
q(:,1)=a1(6,6);
q(:,2)=a2(6,6);
q(:,3)=a3(6,6);
q(:,4)=a4(6,6);
q(:,5)=a5(6,6);
% -----

% -----
% Matriz de Ganhos
kbar=q*inv(v);
% -----
% Determinacao de K1 e K2
% -----
k1=kbar(:,1:4);
k2=kbar(:,5);
% -----
%csaida = [c;1 0 0 0];
Aula2

```

Diagrama de Blocos Simulink



ANEXO D - Matrices de estado

Matriz A			
0.0000000000000000 000e+00	- 1,45069657351318 00E+17	0.0000000000000000 0000e+00	1,1387796971929 800E+20
1,0000000000000000 0000E+17	0.0000000000000000 000e+00	0.0000000000000000 0000e+00	0.0000000000000000 0000e+00
0.0000000000000000 000e+00	1,28605887280274 0000E+16	0.0000000000000000 0000e+00	2,8085769057688 500E+18
0.0000000000000000 000e+00	0.0000000000000000 000e+00	1,0000000000000000 000E+17	0.0000000000000000 0000e+00

Matriz B
-3,11666278848218000E+18
0.000000000000000000e+00
2,762956710483300000E+17
0.000000000000000000e+00

Matriz C			
0.0000000000000000 000e+00	- 1,0000001293853 3E+17	0.0000000000000000 000e+00	- 9,18153726222838 0E+17

Matriz D
0.000000000000000000e+00