

Aplicação da Técnica de Estabilização de Posição “*Gimbal*” na Compensação de Movimentos Involuntários em Humanos

RICARDO MODESTO DA SILVA
(Licenciado em Engenharia Eletrotécnica)

Trabalho Final de Mestrado para obtenção do grau de Mestre
em Engenharia Eletrotécnica Ramo: Automação e Eletrónica Industrial

Orientadores:

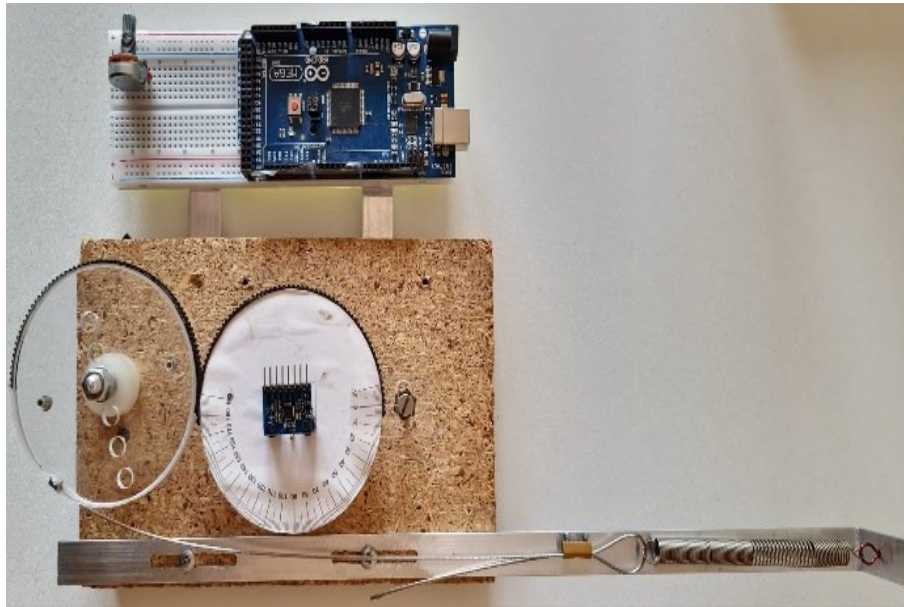
Professor Doutor Luís José Lamy Rocha da Encarnação
Professor Sérgio Fernando Pereira Delgado de Sousa

Júri:

Presidente: Professor Doutor Luís Manuel dos Santos Redondo
1º Vogal: Professor Doutor Paulo José Duarte Landeiro Gambôa
2º Vogal: Professor Doutor Luís José Lamy Rocha da Encarnação

Junho 2023

INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA
Departamento de Engenharia Eletrotécnica Energia e Automação



Aplicação da Técnica de Estabilização de Posição “Gimbal” na Compensação de Movimentos Involuntários em Humanos

RICARDO MODESTO DA SILVA
(Licenciado em Engenharia Eletrotécnica)

Trabalho Final de Mestrado para obtenção do grau de Mestre
em Engenharia Eletrotécnica Ramo: Automação e Electrónica Industrial

Orientadores:

Professor Doutor Luís José Lamy Rocha da Encarnação
Professor Sérgio Fernando Pereira Delgado de Sousa

Júri:

Presidente: Professor Doutor Luís Manuel dos Santos Redondo
1º Vogal: Professor Doutor Paulo José Duarte Landeiro Gambôa
2º Vogal: Professor Doutor Luís José Lamy Rocha da Encarnação

Junho 2023

Dissertação realizada sob orientação de:

Professor Doutor Luís José Lamy Rocha da Encarnação

Professor Adjunto do Departamento de Engenharia Eletrotécnica de Energia e Automação

Professor Sérgio Fernando Pereira Delgado de Sousa

Professor do Departamento de Sistemas e Informática (IPS)

Resumo

O aumento de diversos problemas de saúde, relacionados com a degradação da qualidade de vida, má alimentação e sedentarismo têm suscitado o aparecimento e crescimento de doenças crônicas, como o diabetes ou mesmo o Parkinson. A área do controlo de processos e eletrotecnia poderá ser usada como ponte de maneira a solucionar ou atenuar alguns problemas provenientes destas patologias.

No presente trabalho final de mestrado, tem-se como objetivo, fazer o estudo e caracterização de componentes elétricos e eletromecânicos, bem como a implementação e criação de modelos computacionais que permitirão a simulação do mesmo. Abordou-se e implementou-se métodos de controlo de processo para se poder implementar e modelar o sistema em cadeia fechada.

De modo a validar os modelos teóricos foi desenvolvida uma bancada de testes para simular uma oscilação mecânica, no limite esta oscilação poderá aproximar-se a um tremor comparável aos existentes na doença de Parkinson.

Palavras-chave:

Gimbal, Giroscópio, Parkinson, PID, Controlo de processos, Servo motor, Ziegler-Nichols

Abstract

The increase of several health problems, related to the degradation of the quality of life, poor diet and sedentary lifestyle have led to the emergence and growth of chronic diseases such as diabetes or even Parkinson's. The area of process control and electrotechnics can be used as a bridge to solve or mitigate some problems from these pathologies.

In the present master's final work, the aim is to study and characterize electrical and electromechanical components, as well as to implement and create computational models that will allow the simulation of these same components in an open loop. Process control methods were approached and implemented to implement and model the system in closed chain.

To confirm the theoretical models, a test bench was developed to simulate a mechanical oscillation, which at its limit could approximate a tremor comparable to those present in Parkinson's disease.

Keywords:

Gimbal, Gyroscope, Parkinson, PID, Process control, Servo motor, Ziegler-Nichols

Agradecimentos

Aos meus orientadores Professor Luís Encarnação e Professor Sérgio Sousa, por toda a disponibilidade, muita paciência e compreensão que permitiram a possibilidade da realização desta dissertação. Queria também fazer um agradecimento ao Rodrigo Breno, pela disponibilidade e participação nesta dissertação.

Aos meus pais, Carlos e Teresa e meu irmão Francisco por todo o carinho e por me darem a possibilidade de ter um curso superior.

Aos meus avós, que sem eles não seria possível todo este caminho longo, mas principalmente ao meu avô.

À minha namorada Patrícia Gamito, por toda força, apoio, dedicação, paciência, atenção, insistência, preocupação, esforço, carinho e amor nesta fase, sem ela isto não seria possível.

Aos meus colegas e amigos que ficaram e de alguma maneira contribuíram com a sua amizade, conhecimento ou mesmo paciência para o colmatar desta fase, Nuno Pereira, Sandra Roque, Pedro Porto, Inês Silva, João Semedo.

Lista de Siglas

3DOF – Três Graus de Liberdade (*Three Degrees of Freedom*)

DC – Corrente Contínua (*Direct Current*)

DMP – Processador de Movimento Digital (*Digital Motion Processor*)

MEMS – Sistemas Microeletromecânicos (*Micro-Electro-Mechanical-System*)

MPU – Unidade de Processamento de Movimento (*Motion Processor Unit*)

P – Controlador Proporcional (*Proportional*)

PCB – Placa de Circuito Impresso (*Printed Circuit Board*)

PI – Controlador Proporcional Integral (*Proportional Integral*)

PID – Controlador Proporcional Integral Derivativo (*Proportional Integral Derivative*)

PWM – Modelação por Largura de Pulso (*Pulse Width Modulation*)

Lista de Símbolos

f – Frequência da rede [Hz]

K_p – Ganho proporcional

K_{cr} – Constante para o ganho crítico, no método de Ziegler-Nichols

K_d – Ganho derivativo

K_i – Ganho integral

P_{cr} – Período crítico, no método de Ziegler-Nichols [s]

$R_{(\alpha, X)}$ – Matriz de rotação em torno do eixo X

$R_{(\beta, Y)}$ – Matriz de rotação em torno do eixo Y

$R_{(\gamma, Z)}$ – Matriz de rotação em torno do eixo Z

$R_{O_x O_y O_z}$ – Matriz de obtenção de graus de liberdade

V_{filtro} – Tensão à saída do filtro Passa-baixo [V]

V_m – Tensão aos terminais do motor [V]

V_{pulse} – Tensão à saída do PWM [V]

V_{trans} – Tensão à saída do transdutor de sinal [V]

α – Ângulo de rotação em torno do eixo X [rad ou °]

β – Ângulo de rotação em torno do eixo Y [rad ou °]

γ – Ângulo de rotação em torno do eixo Z [rad ou °]

θ_m – Ângulo do motor [rad ou °]

θ_s – Ângulo do sensor [rad ou °]

τ – Constante de tempo de um sistema de 1ª ordem [s]

ω – Velocidade angular [rad/s]

Índice

Lista de Siglas	vii
Lista de Símbolos	ix
Índice de Figuras	xv
1 Introdução	1
1.1 Enquadramento e Motivação	1
1.2 Objetivos.....	2
1.3 Organização e Estrutura do Documento	2
2 Estado da Arte.....	5
3 Protótipo, Modelação e Ensaio em Matlab	15
3.1 Protótipo	15
3.2 Bancada de Testes para Modelação	16
3.3 Montagem da Bancada de Testes	17
3.4 Elementos Usados	19
3.5 Ensaio para Obtenção da Característica do Servo Motor	22
3.6 Ensaio para Obtenção da Característica do Sensor	29
3.7 Cálculo do Ângulo do Sensor.....	30
3.8 Ângulos de Euler	33
3.9 Gimbal Lock.....	35
4 Implementação e Ensaio Experimentais	45
4.1 Método de Ziegler-Nichols.....	45
4.2 Implementação Laboratorial.....	52
4.3 Controlador PID	54
4.4 Resultados Experimentais	55
5 Conclusões	67
5.1 Conclusões Gerais	67
5.2 Desenvolvimentos Futuros	69
Bibliografia	71
Anexos	75
Anexo A – Características do Servo Motor	75
Anexo B – Características do Sensor MPU6050	76
Anexo C – Características do Arduíno.....	77

Anexo D – Código para Implementação no Arduino.....	78
-----------------------------------------------------	----

Índice de tabelas

Tabela 1 - Relações de Ziegler-Nichols (Original) [24]	46
Tabela 2 - Relações de Ziegler-Nichols (Original, pequena sobrelevação, sem sobrelevação) [25].	49
Tabela 3 - Relações de Ziegler-Nichols.	54
Tabela 4 - Resumo de resultados experimentais para $K_p=1$	59
Tabela 5 - Resumo de resultados para referência a 40°	65

Índice de Figuras

Figura 1 – Exemplo dos eixos de rotação, Roll, Pitch e Yaw.	6
Figura 2 – Exemplo de um Giroscópio, adaptado de [11].	7
Figura 3 - Exemplo de manipulador para a "workstation" de Laparoscopia, adaptado de [13].	8
Figura 4 - Exemplo de mecanismo gimbal para o controlo da sonda, adaptado de [14].	9
Figura 5 - Representação do modelo da interação com o coração representada com um esquema massa-mola-atrito, adaptado de [15].	9
Figura 6 - "Wrist gimbal" - Um exosqueleto com antebraço e pulso para reabilitação de ataque cardíaco, adaptado de [16].	10
Figura 7 – Exemplo de Governador de Watts, adaptado de [17].	11
Figura 8 – Exemplo de modelo biodinâmico da palma mão 4DOF, adaptado de [20].	13
Figura 9 - Esquema do modelo da colher utilizado, adaptado de [21].	14
Figura 10 - Solução proposta de protótipo.	15
Figura 11 - Diagrama de blocos do sistema.	16
Figura 12 - Bancada de testes (vista topo).	18
Figura 13 - Bancada de testes (vista lateral).	19
Figura 14 - Diagrama de blocos do servo motor.	19
Figura 15 - Vista lateral do servo motor.	20
Figura 16 - Vista dos componentes internos do servo motor.	20
Figura 17 - Diagrama esquemático dos componentes internos de um servo motor.	21
Figura 18 - Ligações e circuito integrado MPU 6050.	21
Figura 19 - Diagrama de blocos do sensor.	22
Figura 20 - Montagem da medição da característica do servo motor.	23
Figura 21 - Ensaio de posição do servo motor.	24
Figura 22 - Escalão de posição de 0° graus para 90° do ensaio experimental do servo motor.	24
Figura 23 - Análise da constante de tempo τ e ganho, para a tensão aos terminais do servo motor, do ensaio de escalão posição de 0° a 90.	25
Figura 24 - Diagrama de blocos utilizado na obtenção do modelo do servo motor.	26
Figura 25 - Diagrama de Bode do filtro passa baixo.	27
Figura 26 - Resultado experimental do ensaio da aplicação do escalão posição de 0° para 90° no servo motor.	28

Figura 27 – Resultados da simulação para modelo do servo motor quando aplicado o escalão posição 0° a 90°.	28
Figura 28 - Esquema de ligações do ensaio da característica do sensor.	29
Figura 29 – Demonstração gráfica do chip com os eixos X,Y e Z, adaptado do [Anexo B].	30
Figura 30 - Exemplo dos eixos utilizados como convenção na aviação adaptado de [22].	34
Figura 31 - Exemplo de Sistema de coordenadas para o qual acontece o gimbal lock.	36
Figura 32 - Ensaio experimental para obtenção da característica do modelo do sensor.	37
Figura 33 – Ampliação do ensaio experimental para obtenção da característica do modelo do senso, para o escalão posição 0° a 90°.	38
Figura 34 - Diagrama de blocos do modelo do sensor.	39
Figura 35 - Comparação entre modelo teórico do sensor e modelo do ensaio experimental.	39
Figura 36 - Comparação entre modelo teórico do sensor e ensaio experimental no escalão posição (0° aos 90°).	40
Figura 37 - Diagrama de blocos do modelo do sensor para os parâmetros ajustados ($G=0,95$ e $\tau = 0,0159s$).	41
Figura 38 - Comparação entre ensaio experimental e modelo do sensor com parâmetros ajustados ($G=0,95$ e $\tau =0,0159s$).	41
Figura 39 - Comparação entre ensaio do escalão posição dos 0° aos 90° e modelo sensor ajustado ($G=0,95$ e $\tau=0,0159s$).	42
Figura 40 - Diagrama de blocos do modelo do sensor ajustado com atraso ($G=0,95$, $\tau=0,0159$ e atraso de $0,039 s$).	43
Figura 41 - Comparação entre ensaio experimental e modelo do sensor com e sem atraso ($G=0,95$, $\tau=0,0159$ e atraso $0,039s$).	43
Figura 42 - Comparação entre ensaio experimental para o escalão posição dos 0° aos 90° e modelo do sensor com e sem atraso ($G=0,95$, $\tau=0,0159s$ e atraso de $0,039s$).	44
Figura 43 - Diagrama de blocos para controlador PID, usando o método de Ziegler-Nichols [23].	45
Figura 44 – Onda no limite da estabilidade para quando o ganho do controlador é $K_{cr}=0,9$.	47
Figura 45 – Sinal de saída com controlador PID, ($K_p=0,54$, $K_i=0,589$, $K_d=0,368$).	48
Figura 46 – Sinal de saída com controlador PID ajustado, ($K_p=0$, $K_i=0,605$, $K_d=0$).	49
Figura 47 - Sinal de saída com controlador PID, com pequena sobrelevação ($K_p=0,297$, $K_i=0,324$, $K_d=0,054$).	50
Figura 48 - Sinal de saída com controlador PID, sem sobrelevação ($K_p=0,18$, $K_i=0,196$, $K_d=0,032$).	51
Figura 49 - Bancada de testes com implementação do sistema Massa-Mola-Atrito.	52

Figura 50 – Fotografia da mola utilizada.	53
Figura 51 - Roda dentada em acrílico com cabo de aço.....	53
Figura 52 – Observação do sinal de saída, do ensaio experimental com referência a 15°, para $K_p=1$	56
Figura 53 – Observação do erro, do ensaio experimental com referência a 15°, para $K_p=1$	56
Figura 54 - Observação do sinal de saída, do ensaio experimental com referência a 25°, para $K_p=1$	57
Figura 55 - Observação do erro, do ensaio experimental com referência a 25°, para $K_p=1$	57
Figura 56 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=1$	58
Figura 57 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1$	59
Figura 58 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=1,35$	60
Figura 59 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1,35$	61
Figura 60 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=1,5$	62
Figura 61 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1,5$	62
Figura 62 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=2$	63
Figura 63 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=2$	63
Figura 64 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=3$	64
Figura 65 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=3$	65

1 Introdução

1.1 Enquadramento e Motivação

Perante um aumento do número de casos relacionados com a degradação da qualidade de vida, alimentação e fatores ambientais, existe cada vez mais o surgimento de diversas patologias médicas associadas a estes fatores. Sendo várias destas doenças irreversíveis e crónicas, entre todas as doenças que provocam um severo degradar do dia a dia e da qualidade de vida para o ser humano, provocando desordens como, tremores, dificuldades em locomover ou mesmo a impossibilidade total de fazer alguma tarefa básica do dia a dia, temos a doença de Parkinson. Com base num estudo epidemiológico feito em Portugal para a avaliação da prevalência da doença de Parkinson, sendo este um dos primeiros levantamentos sobre este tema feito no país, os resultados foram revelados e publicados num artigo da Revista Científica da Ordem dos Médicos, que teve como base uma amostra populacional acima dos 50 anos de idade para indivíduos do sexo masculino e feminino, calculou uma prevalência da doença de Parkinson em 180 por cada 100 000 habitantes [1]. Com base neste crescente aparecimento de casos, e da degradação da qualidade de vida das pessoas com este tipo de síndrome, abordou-se uma alternativa para poder melhorar a qualidade de vida das pessoas com esta condição de vida.

Com a motivação obtida, procurou-se arranjar uma solução para melhorar a qualidade de vida das pessoas, usando para isso conhecimentos na área de Eletrotecnia.

Têm surgido diversas técnicas para fazer uma atenuação de movimentos involuntários através da estabilização. O uso do giroscópio ou *gimbal* foi amplamente aplicado desde a antiguidade média através de um sistema mecânico, que permite uma estabilização em relação a vários eixos tendo como principal atuador a gravidade que faz uma perpendicular em relação ao centro do planeta terra.

Esta dissertação, aborda a análise de um sistema eletrónico e eletromecânico. Neste estudo procurou-se, com base na área de controlo de processos, o desenvolvimento de um dispositivo, que servirá como atenuador dos movimentos involuntários provocados pela condição de Parkinson, problema que afeta por exemplo o facto de a pessoa com esta condição não conseguir alimentar-se de maneira autónoma.

O dispositivo terá por base a técnica de estabilização *gimbal*. Esse dispositivo fará ligação á mão do paciente que terá o seu controlo nos três eixos (X, Y e Z), de maneira a poder

compensar os tremores involuntários da mão. Na sua ponta poderá ter por exemplo, um garfo ou uma colher. Com isto o paciente poderá alimentar-se mais autonomamente.

1.2 Objetivos

O objetivo desta dissertação, é estudar, caracterizar e dimensionar um sistema eletrônico e eletromecânico, para estabilização de movimentos involuntários em seres humanos, sendo assim os assuntos que estarão em foco são:

1. Dimensionar um sistema eletromecânico com um auxílio de uma bancada de testes pensada para obter a resposta dos componentes utilizados nas grandezas elétricas e mecânicas e criar um modelo;
2. Validar o modelo obtido;
3. Simular o sistema com a validação dos dados em cadeia fechada;
4. Desenvolver e implementar um protótipo que simule uma oscilação mecânica e efetuar medições com a implementação de um controlador em cadeia fechada;
5. Comparação do modelo em simulação, com os dados provenientes da bancada de testes;
6. Estudo do sistema no eixo Z, fazendo uma replicação para os eixos X e Y.

1.3 Organização e Estrutura do Documento

Esta dissertação está organizada em 5 capítulos, referências bibliográficas e anexos.

No primeiro capítulo, é apresentando o enquadramento, motivação, organização e estrutura do presente documento.

Quanto ao segundo, é alusivo ao estado da arte, onde é descrito a primeira aparição de um giroscópio e *gimbal*, bem como as suas primeiras aplicações. Durante o decorrer deste capítulo vai-se falar na evolução do giroscópio e *gimbal* como também em várias aplicações, tanto na área da aeronáutica bem como na área da biomedicina.

No terceiro capítulo é apresentada uma posposta de protótipo e como irá ser composto o sistema a ser estudado, bem como a bancada de testes, para fazer a modelação dos dispositivos usados no sistema. Também é apresentada a descrição dos ensaios laboratoriais para obter as características a utilizar nos modelos, as constantes de tempo

e ganho para a modelação dos mesmos. É ainda apresentada a modelação com os parâmetros obtidos dos ensaios realizados, bem como os passos para obtenção do modelo em *Matlab*.

Relativamente ao quarto capítulo é demonstrada a adaptação feita na bancada de testes para poder-se realizar os ensaios do sistema mecânico, que permitirá simular o estímulo que advém da doença de Parkinson, de maneira a poder fazer o controlo do processo em cadeira fechada. Com a utilização de controladores do tipo P, PI e PID, utilizando para isso o método Ziegler-Nichols. Bem como os resultados obtidos da implementação do controlo.

No quinto e último capítulo, são apresentadas as conclusões gerais, bem como perspectivas de desenvolvimentos futuros.

2 Estado da Arte

O giroscópio e *gimbal* foi descrito pela primeira vez, pelo inventor grego, Philo de Byzantium (280-220 A.C.). Philo descreveu o *gimbal* como um dispositivo de oito faces usado para guardar o tinteiro de escrita. O dispositivo, podia-se abrir de qualquer uma das suas faces permitindo assim molhar a pena de escrita sem que a tinta escorre se pelas restantes faces, sendo este suspenso por um conjunto de anéis metálicos concêntricos que permitiam a sua estabilidade [2].

Nuns indícios encontrados em escritos antigos por Sima Xiangru (179-117 A.C.), o *gimbal*, já existia na China desde o segundo século antes de cristo [3]. Numa menção mais precisa na dinastia Liang (502-557) os *gimbal* seriam usados em dobradiças de portas e janelas, e mesmo num fogão [4].

Na antiga China, há várias referências a inúmeros tipos de *gimbal* usados para queimadores de incenso até ao princípio da dinastia Tang (618-907) [5], [6]. Apesar de todas as dúvidas sobre a autenticidade das referências na antiga China, em 1997 foram feitas pesquisas que demonstram uma cópia árabe que contém a descrição do uso dos mesmos enunciando letras gregas que era relacionadas com o *gimbal* que caíram já em desuso depois do primeiro século, com isto confirmando ou sustentando que a cópia será fiel ao Hellenistic original, uma opinião também partilhada pelo historiador clássico Andrew Wilson [7].

Já na atualidade os relógios e os cronómetros são suspensos nos navios com ajuda de sistema de molas, engrenagens e todo o seu mecanismo de maneira a não sofrer impacto nas suas leituras com a oscilação das ondas em alto mar, usando para isto um mecanismo de *gimbal* [8].

No exemplo da navegação inercial, o sistema usa os dados do *gimbal* para monitorizar continuamente a posição, velocidade e aceleração de um veículo, como por exemplo um submarino, míssil ou um avião de combate aéreo ou comercial. O conceito da navegação inercial tem como base giroscópios ou *gimbal*, acelerómetros e computadores para tratar toda a informação podendo assim ser utilizada para a sua navegação.

O giroscópio fornece os dados para a referência de direções, rotações e os acelerômetros medem as diferenças na velocidade do objeto. O computador processa a informação das mudanças de direção e da aceleração e fornece esses resultados ao sistema de navegação do objeto. Há dois tipos de navegação inercial, um a usar sistemas com base *gimbal* e sistemas de *strapdown* [9].

Como estamos a tratar de sistemas em que se usa o *gimbal*, abordar-se-á, os sistemas de navegação inercial, dado que os mesmos servem para fazer a orientação dos objetos no espaço, estes podem consistir por exemplo, um míssil, avião ou tanque, estes sistemas podem usar até três giroscópios e três acelerômetros, como indicado na Figura 1, estes sensores estabelecem um eixo de referência para o *roll* (*Roll* – define a rotação em torno do eixo que vai da traseira para a frente do objeto), do *pitch* (*Pitch* – rotação em torno do eixo da esquerda para a direita do objeto) e do *yaw* (*Yaw* – rotação em torno do eixo que vai de cima para baixo do objeto). Os acelerômetros medem as mudanças de velocidade em cada um desses eixos.

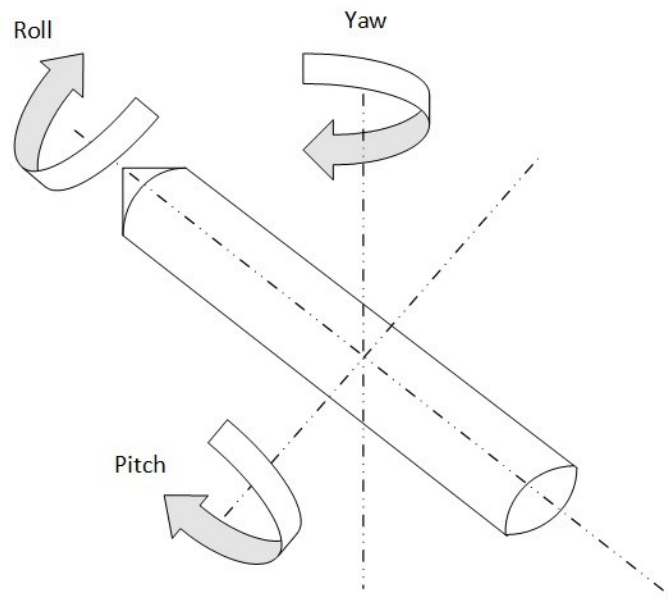


Figura 1 – Exemplo dos eixos de rotação, Roll, Pitch e Yaw.

Com base no *gimbal* foi criado outro dispositivo, o giroscópio.

Os giroscópios como os *gimbal* começaram por ser mecânicos primeiramente evoluindo depois para uma base elétrica e eletrônica, estes são baseados num princípio descoberto

no século 19 por Jean-Bernard-Léon Foucault, um físico francês que deu o nome de giroscópio a uma roda ou rotor montado no centro dos anéis do *gimbal*.

Durante os anos de 1850 Jean Foucault, conduziu uma experiência em que se usa um rotor, com esse rotor demonstrou que a rotação se mantinha no espaço independente da rotação da terra. Surgiu em 1908 o primeiro giro compasso utilizável, desenvolvido pelo inventor alemão, H. Anschutz-Kaempfe para uso de um submersível.

Em 1909, Elmer A. Sperry construiu o primeiro piloto automático usando como base um giroscópio para ajudar um avião a manter a sua rota. O primeiro piloto automático para navios, foi instalado num navio de passageiros dinamarquês por uma empresa alemã em 1916, nesse mesmo ano o giroscópio foi também usado para desenhar o primeiro horizonte artificial nos monitores de um avião.

Porque existe uma rotação associada a um disco, ou neste caso a uma massa, ao deslocar o giroscópio, como indicado na Figura 2, em qualquer rotação a orientação do eixo em que este está a rodar não é alterada. Isto esta relacionado com a conservação do momento angular dessa mesma massa [10].

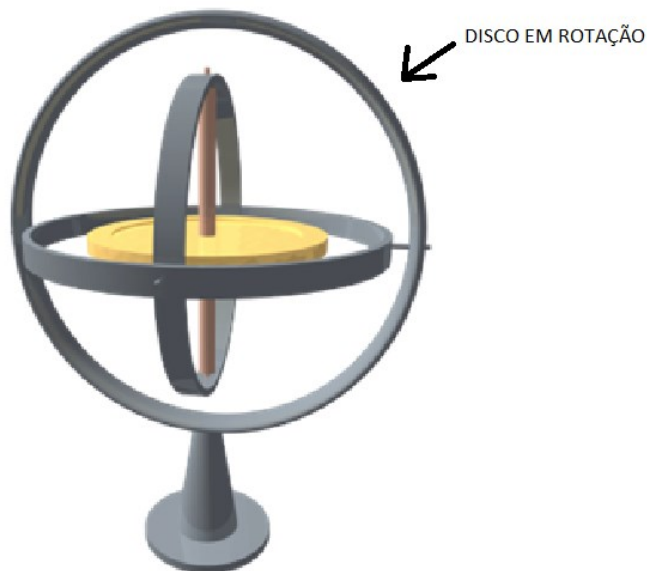


Figura 2 – Exemplo de um Giroscópio, adaptado de [11].

Em 2001, num artigo de Polla Dennis L., descreve-se algumas das possibilidades para o uso da tecnologia MEMS para aplicações Biomédicas. Neste artigo fala-se desde

microssistemas para execução de cirurgias, com recurso a materiais piezo-resistivos, ou mesmo sendo introduzida a possibilidade de microssistemas de diagnóstico e sistemas para terapia [12].

Em 1999, observa-se a utilização de um robot para fazer remotamente uma Laparoscopia, usando um protótipo de um robot com um estabilizador com 3 graus de liberdade. Para poder realizar os movimentos e conseguir fazer a intervenção médica de uma maneira estável, usaram um estabilizador na base do suporte com recurso à técnica de *gimbal* como indicado na Figura 3 [13].



Figura 3 - Exemplo de manipulador para a "workstation" de Laparoscopia, adaptado de [13].

Em 2001, no artigo com o título “*Three dimensional motion mechanism of ultrasound probe and its application for tele-echography system*”, para a conferência internacional em robots e sistemas inteligentes, fez-se referência a um *gimbal* para o desenvolvimento de um robot que tem um movimento em três dimensões. O *gimbal* é usado para controlar o ângulo da sonda de ultrassons que vai fazer contacto com o corpo humano como indicado na Figura 4 [14].

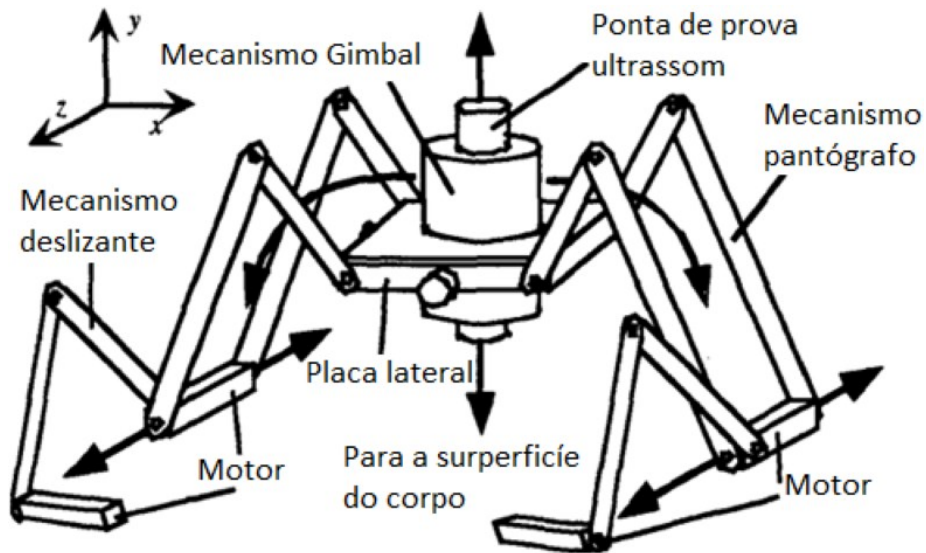


Figura 4 - Exemplo de mecanismo gimbal para o controle da sonda, adaptado de [14].

Em 2011, num artigo para o IEEE com o título “*Active Stabilization for Robotized Beating Heart Surgery*”, foi usado dispositivo com um controle associado a um estabilizador de maneira a poder compensar o movimento do batimento do coração durante o procedimento de intervenção cirúrgica de um *bypass* da artéria coronária. Nesse mecanismo com a ajuda de um estabilizador ativo com mecanismo piezo resistivo, é possível controlar o dispositivo de maneira a compensar o movimento do bater do coração, ajudando assim o cirurgião. Neste artigo é feita a modelação do sistema começando com o estudo do batimento do coração e com a aproximação desse movimento ao sistema mola-massa-atrito, com esta modelação e validação de resultados, faz com que se possa diminuir os riscos da intervenção cirúrgica aquando da implementação do sistema [15], como indicado na Figura 5 o sistema mecânico utilizado para a modelação.

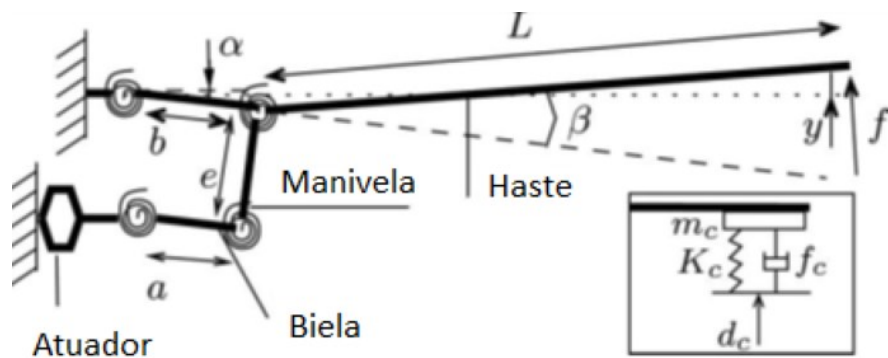


Figura 5 - Representação do modelo da interação com o coração representada com um esquema massa-mola-atrito, adaptado de [15].

Em 2013, no artigo “*Design of Wrist gimbal: a Forearm and Wrist Exoskeleton for Stroke Rehabilitation*” para a conferencia internacional para a reabilitação em robótica da IEEE, foi desenhado e implementado um *Wrist gimbal*, com 3 graus de liberdade para a reabilitação de lesões no antebraço e pulso provocado por um surgimento de um ataque cardíaco.

O *Wrist gimbal* com os seus 3DOF que dizem respeito, à supinação, flexão, extensão e adução das junções do pulso. Com a ajuda dos vários graus de liberdade facultados pelo *gimbal*, os sujeitos podem então fazer os exercícios de reabilitação, evitando assim um esforço desnecessário noutros movimentos não abonatórios para a sua reabilitação, focando e localizando nas áreas afetadas os exercícios [16], como indicado na Figura 6.

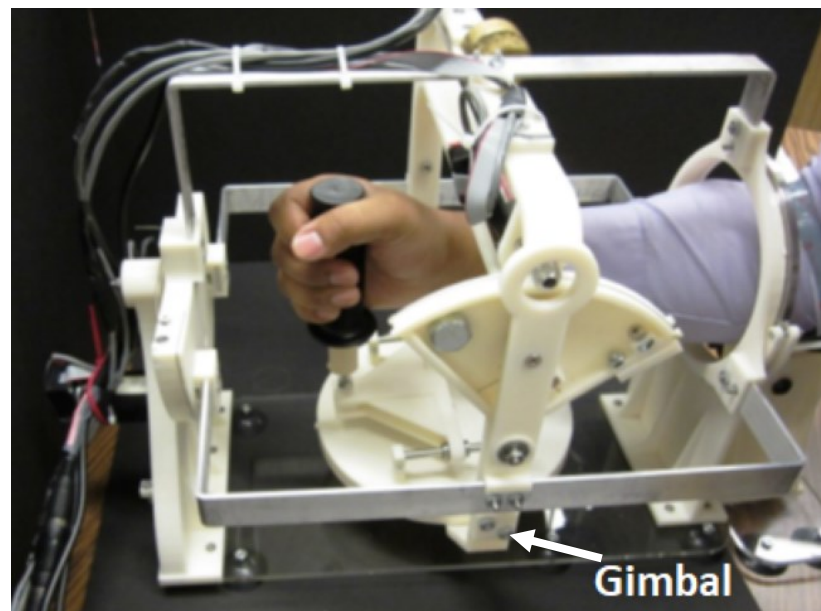


Figura 6 - "*Wrist gimbal*" - Um exosqueleto com antebraço e pulso para reabilitação de ataque cardíaco, adaptado de [16].

A aplicação de sistemas de controlo, para diversos dispositivos remonta à época das civilizações antigas, onde utilizavam diversos tipos de relógios de água para medir o tempo de uma forma mais precisa.

Os grandes impulsionadores deste tipo de dispositivos foram os gregos e árabes, foram estes que fizeram a implementação desses relógios a partir do século III a.C. No entanto, só em 1788, com a criação do regulador automático de Watts.

Este dispositivo tem como função regular automaticamente a velocidade de rotação de um motor independentemente da carga.

O princípio de funcionamento, consiste num par de massas a girar em torno de um eixo acionado pelo motor, como indicado na Figura 7 [17]. O regulador de Watts faz esta regulação de velocidade com a variação da taxa de débito do combustível.

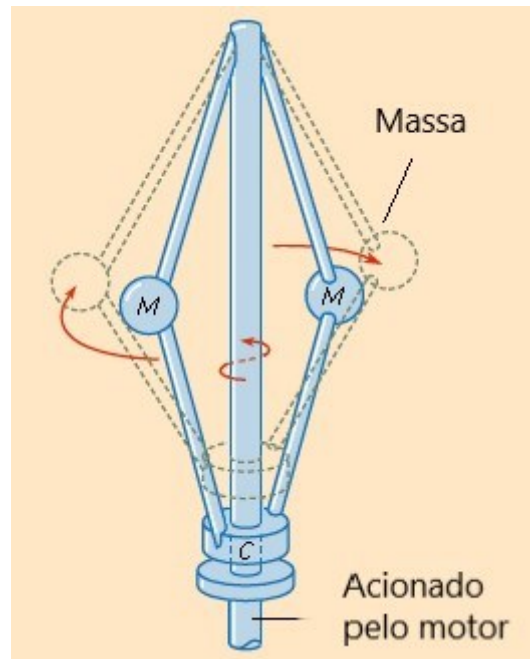


Figura 7 – Exemplo de Governador de Watts, adaptado de [17].

Foi considerado o primeiro sistema automático e o ponto de partida para a Revolução industrial. Com esta inovação, foi possível fazer o controlo de máquinas e processos industriais de forma mais eficiente.

Ao longo do século XIX, houve importantes avanços na modelação matemática que foram alcançados por Leonhard Euler, Pierre Simon Laplace e Joseph Fourier.

Em 1885, Al Butz introduziu um termostato que se tornou o precursor da empresa Honeywell e foi considerado o grande segundo dispositivo automático.

O início do século XX é conhecido como a "era de ouro" da engenharia de controlo. Durante este período, o Laboratório Bell foi responsável pelo desenvolvimento dos métodos clássicos de controlo, graças aos trabalhos de Hendrik Wade Bode e Harry Nyquist. Minorsky, um matemático russo-americano, também desempenhou um papel

importante ao criar controladores automáticos para controlar as rotas de navios, além de introduzir o conceito de controlo integral e derivativo na década de 1920.

Em 1942, no livro “*Optimum settings for automatic controllers*”, Ziegler and Nichols, criaram um método que nos dias de hoje é considerado um método clássico. Que tem duas possibilidades de ser aplicado, ou pelo método da curva de reação ou o método do ganho crítico. Sendo o método do ganho crítico o mais interessante para sistemas dos quais se desconhecem algumas dinâmicas do processo [18].

Nesse contexto, os conceitos de estabilidade, propostos por Nyquist e seguidos por Evans, ganharam destaque. No entanto, foi apenas após a década de 1950 que os métodos de controlo modernos foram desenvolvidos por Rudolf Kalman, almejando superar os métodos clássicos e as suas limitações.

Finalmente, em 1975, foram introduzidos os Controladores Lógicos Programáveis, que marcaram um novo marco na história dos sistemas de controlo industrial, com esta tecnologia, foi possível revolucionar a automação industrial, permitindo o controlo e a monitorização de processos mais complexos e de forma mais eficiente [19].

Em 2016, no artigo “*Active Force control system for hand tremor suppression by different actuators*” para a conferencia internacional para a dispositivos eletrónicos sistemas e aplicações, foi apresentado um sistema de controlo ativo para atenuar o tremor das mãos, mais concretamente de pessoas com Parkinson. Neste artigo, houve a tentativa de usar os melhores controladores e atuadores de maneira a poder fazer a atenuação do tremor. Este atuador baseia se num sistema piezoelétrico e um controlador do tipo PD.

A modelação da palma da mão, foi baseada num modelo biodinâmico de 4 graus de liberdade. Usando para isso uma combinação de sistemas Massa-Mola-Atrito, tendo cada um desses sistemas a função de representar as componentes biológicas da palma da mão, como a epiderme, derme, tecidos subcutâneos e músculo, como indicado na Figura 8 [20].

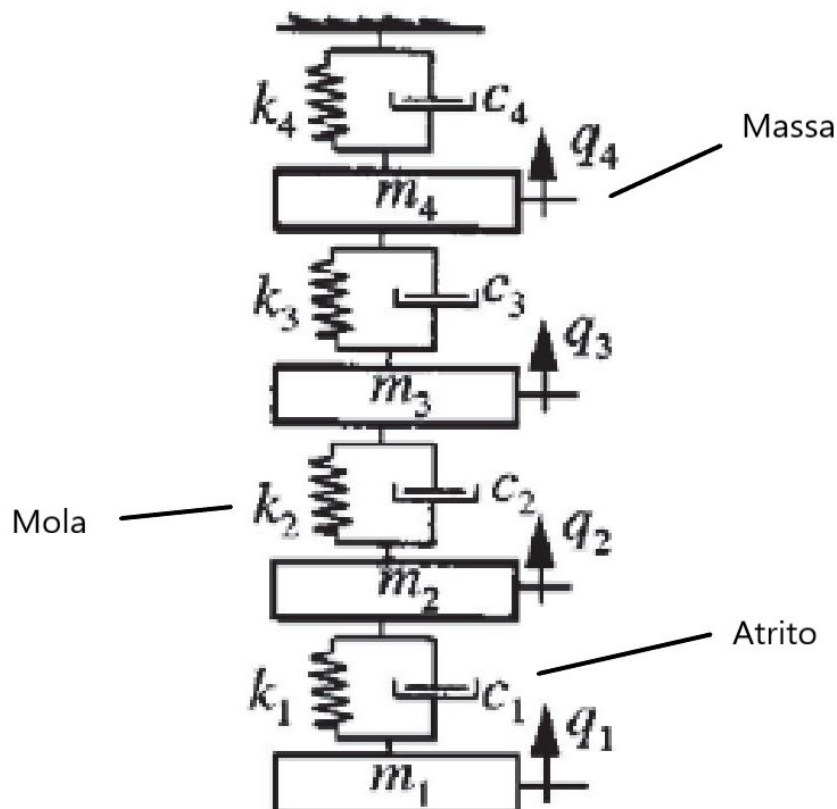


Figura 8 – Exemplo de modelo biodinâmico da palma mão 4DOF, adaptado de [20].

Em 2018, num artigo de investigação “Modeling and experimental study of a hand tremor suppression system“, em que o propósito foi arranjar uma maneira de suprimir os tremores de um doente com Parkinson que usa uma colher para se alimentar. Para isto usam um dispositivo piezoresistivo capaz de suprimir uma vibração com uma oscilação de frequência entre 3 a 12 Hz, tendo também abordagens interessantes como o uso do próprio tremor para fazer a alimentação dos componentes elétricos internos, como indicado na Figura 9 [21].

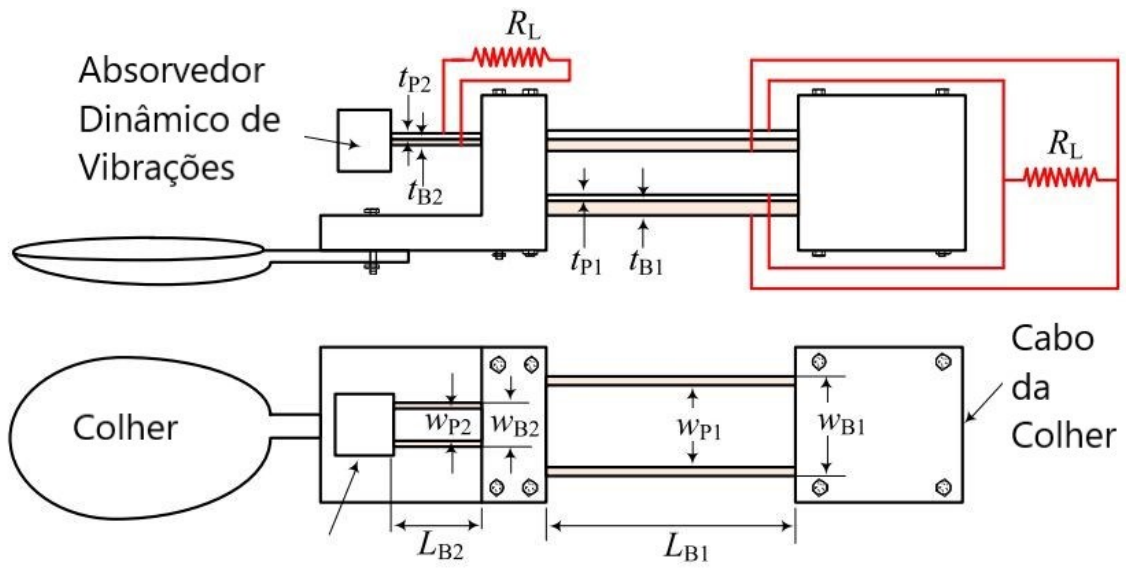


Figura 9 - Esquema do modelo da colher utilizado, adaptado de [21].

3 Protótipo, Modelação e Ensaios em Matlab

3.1 Protótipo

Com o objetivo de criar uma abordagem com dispositivos de baixo custo e de fácil acesso para a população em geral, e com a intenção de arranjar uma solução não invasiva, procurou-se a criação de um protótipo, usando o princípio do Gimbal.

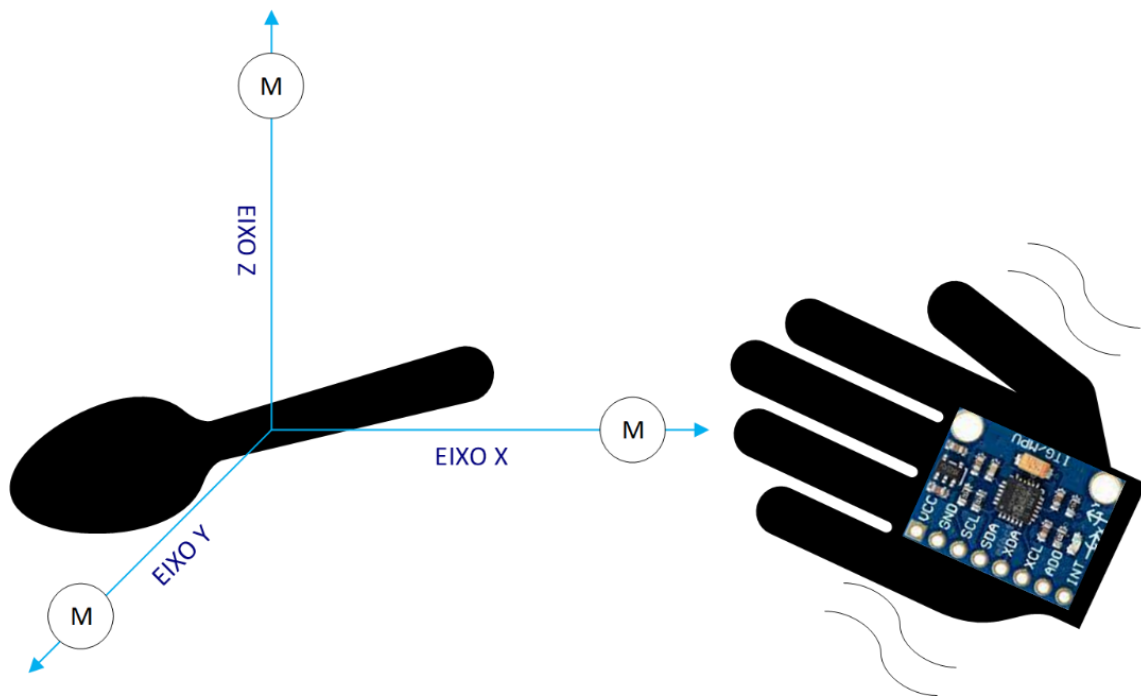


Figura 10 - Solução proposta de protótipo.

Como indicado na Figura 10, o protótipo terá um motor por eixo de forma a controlar a vibração devida à doença na mão. Na mão estaria também o sensor que irá fornecer as leituras necessárias de maneira a obter o sinal de feedback do sistema.

O sistema proposto como protótipo é o indicado na Figura 11.

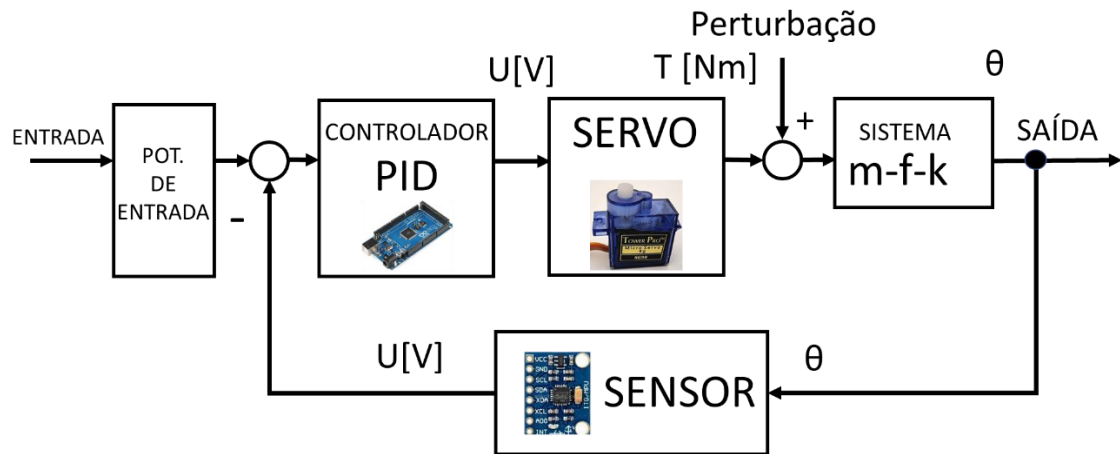


Figura 11 - Diagrama de blocos do sistema.

No sistema proposto, a entrada do sistema será o movimento a executar, o potenciômetro de entrada fará a transformação do movimento rotacional em tensão.

Depois teremos um controlador, que fará a compensação necessária, para cada eixo. Nesse controlador irá ser aplicado um sistema linear do tipo PID para implementar num Arduino.

Como atuador temos o servo motor, como indicado na Figura 11.

A representar a mão do doente de Parkinson utilizou-se o sistema de segunda ordem, Massa-Mola-Atrito, como foi abordado no artigo “*Active Force control system for hand tremor suppression by different actuators*” [20], para a modelação do sistema. Na mão será aplicado um sensor de maneira a obter o sinal de feedback do doente de Parkinson.

A perturbação do sistema, é a oscilação causada pelo tremor do paciente, como indicado na Figura 11.

3.2 Bancada de Testes para Modelação

Com o objetivo de realizar ensaios experimentais para o dimensionamento e caracterização do sistema, criou-se uma bancada de testes.

Neste capítulo, vai-se fazer uma apresentação da bancada de testes, dos seus elementos e por consequência os testes a realizar.

Os testes foram feitos em duas fases:

1. Testar o servo motor, para obtenção das suas características elétricas e mecânicas;
2. Testar o sensor, após validação do modelo do servo.

Os principais elementos utilizados no sistema são o servo motor, modelo SG90 do fabricante Tower Pro (Anexo A), que tem como objetivo exercer a força motriz que fará mover os elementos na bancada e o sensor, modelo MPU 6050 da InvenSense (Anexo B), que desempenhará o papel da retroação usado no controlo em cadeia fechada.

3.3 Montagem da Bancada de Testes

Com o objetivo de acoplar todos os elementos do sistema, criou-se uma bancada de testes para obter e retirar as características elétricas e respostas, tanto do servo motor como do sensor.

A bancada tem como base de uma tábua de madeira contraplacada, com um orifício das mesmas dimensões do servo motor, para fazer a sua montagem, ficando assim imobilizado, evitando assim algum tipo de oscilações que poderão advir dos testes, provocando possíveis erros de leitura.

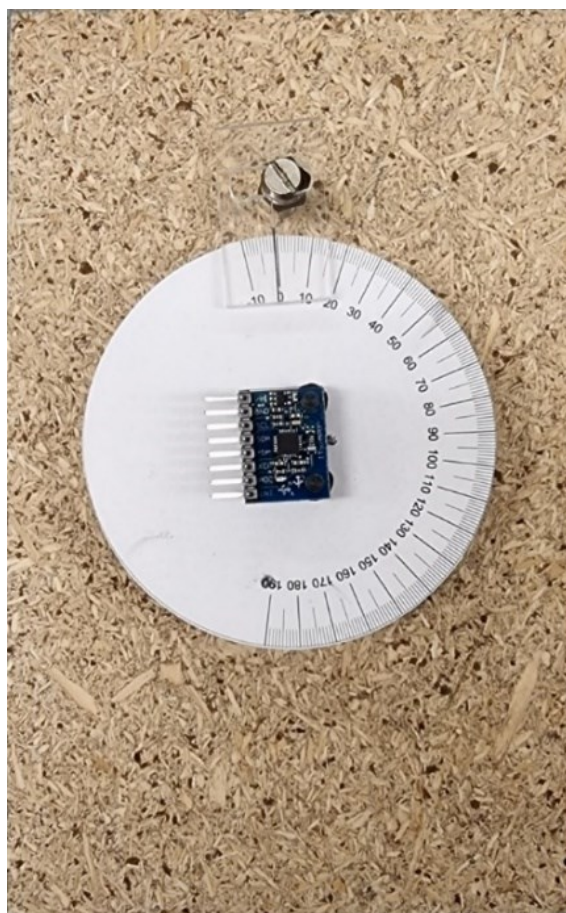


Figura 12 - Bancada de testes (vista topo).

Como indicado na Figura 12, também faz parte da bancada um disco de acrílico com uma escala angular no seu topo, que varia de -10° a 190° .

O disco possui suportes de montagem para a montagem do sensor, garantindo que o centro do mesmo esteja no eixo de rotação do servo motor. Além disso, há uma placa externa de acrílico que permite a leitura do ângulo de rotação do servo, podendo desta maneira fazer uma verificação dos dados fornecidos pelo sensor, sabendo-se assim se os mesmos são precisos ou não.

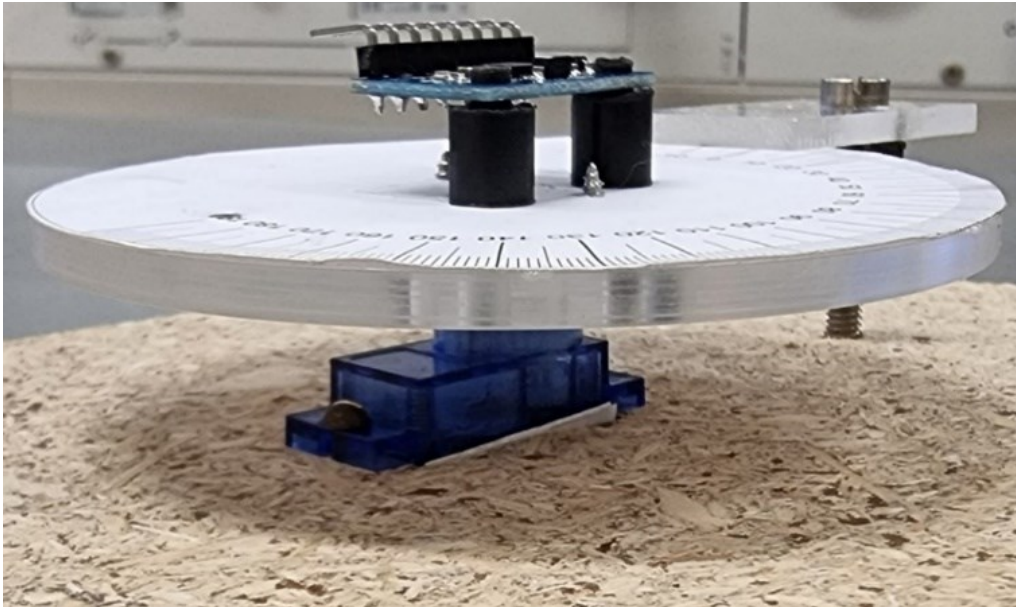


Figura 13 - Bancada de testes (vista lateral).

3.4 Elementos Usados

Com a bancada de testes criada, procedeu-se aos ensaios. Os resultados que irão ser obtidos, terão como base os diagramas de blocos abaixo.

- **Servo Motor**

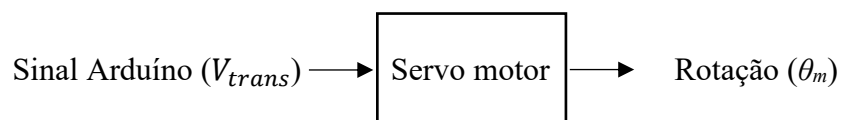


Figura 14 - Diagrama de blocos do servo motor.

O servo motor tem na sua constituição três componentes, um motor DC que fornece o movimento angular, um potenciômetro que tem como finalidade a obtenção do posicionamento do motor DC para o sinal de comando fornecido, e uma placa de circuito integrado, como indicado na Figura 15, Figura 16 e Figura 17. Não tendo o acesso a quaisquer informações sobre estes componentes internos vai-se assumir o servo motor como um único sistema, como indicado pelo diagrama de blocos na Figura 14.



Figura 15 - Vista lateral do servo motor.



Figura 16 - Vista dos componentes internos do servo motor.

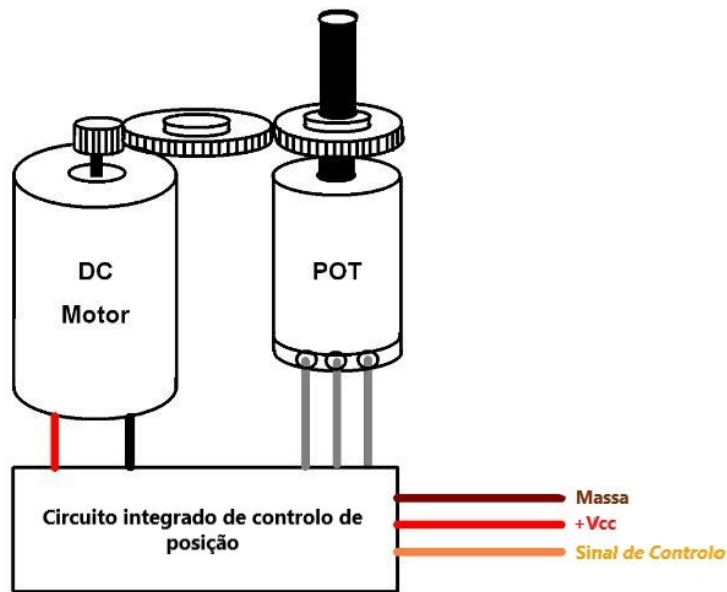


Figura 17 - Diagrama esquemático dos componentes internos de um servo motor.

O sensor, é composto por um circuito integrado onde se observa o chip pertencendo ao MPU 6050 ao centro do PCB, bem como um regulador de tensão no topo do mesmo como indicado na Figura 18. Observando as ligações do PCB, observa-se a alimentação, massa, interface I2C usada para a comunicação com o controlador, bem como pin de endereçamento e pin de interrupção. Com o objetivo de estudar o sensor, vai-se assumir o sensor como um único sistema, como indicado pelo diagrama de blocos na Figura 19.

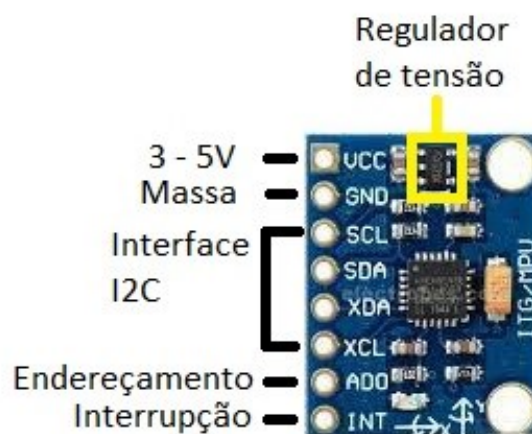


Figura 18 - Ligações e circuito integrado MPU 6050.

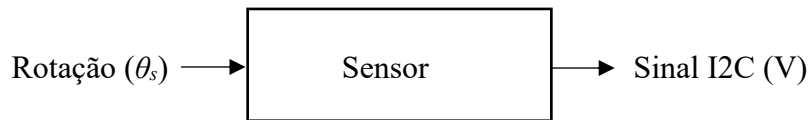


Figura 19 - Diagrama de blocos do sensor.

3.5 Ensaio para Obtenção da Característica do Servo Motor

Com o objetivo de fazer uma validação dos dados fornecidos pelo fabricante no datasheet, e sabendo que existe um fabrico em massa destes equipamentos, existe a possibilidade de cada unidade ter diferentes tempos de resposta para uma determinada rotação pretendida, procedeu-se então ao ensaio da unidade utilizada nesta dissertação.

Começou-se por aplicar um sinal em forma de onda PWM para provocar o arranque do mesmo, este sinal é aplicado em microssegundos (μs) através do controlador Arduino, modelo Mega 2560 (anexo C).

Da ficha de especificação técnica do servo motor modelo SG90, do fabricante Tower Pro, sabe-se que o intervalo de rotação do motor é de 0° a 180° , sendo que o intervalo de tempo para o controlo entre estes ângulos é correspondentemente de $500\mu\text{s}$ até $2400\mu\text{s}$. O sinal de controlo em PWM é dado numa frequência de 50Hz, tendo o motor *pulse cycle* de 20 milissegundos.

Com esta informação, conseguiu-se obter e confirmar os valores para os quais se pode iniciar o estudo sobre o componente, se estes mesmos valores fazem ou não correspondência à rotação esperada, e sendo esta confirmada pela escala angular no topo do disco de acrílico. Após fazer diversos ensaios, os dados obtidos foram para 0° de $590\mu\text{s}$, 90° de $1460\mu\text{s}$ e para 180° foi $2390\mu\text{s}$, verifica-se com a observação destes valores que os dados fornecidos pelo fabricante na ficha de especificação do equipamento, estão dentro do intervalo normalizado pelos mesmos que vão de $500\mu\text{s}$ até $2400\mu\text{s}$.

Após a confirmação dos valores obtidos, correspondentes ao tempo que necessário para provocar a rotação desejada no servo motor, efetuou-se as ligações das pontas de prova do osciloscópio. O primeiro canal foi utilizado para medir o sinal de controlo vindo do Arduino para o servo motor, no segundo canal ligou-se o segundo par de pontas de prova para fazer a medição da tensão aos terminais do motor DC. Neste canal as pontas de prova utilizadas foram do tipo diferencial, com uma escala de (1/20).

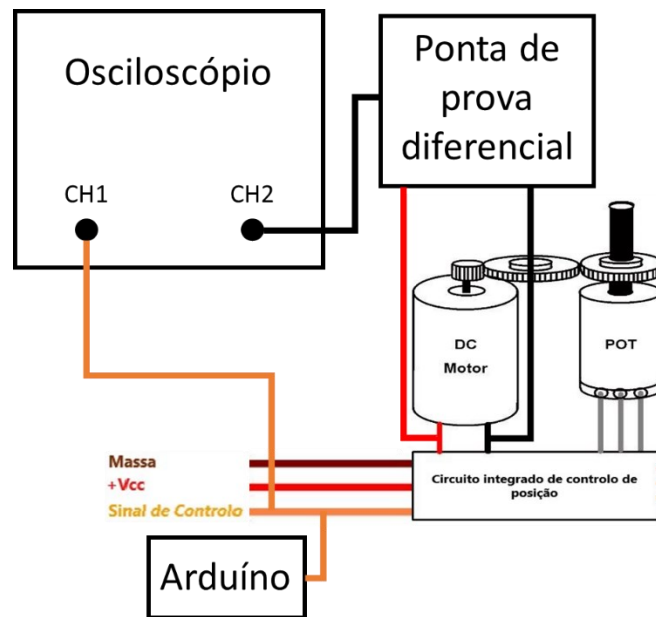


Figura 20 - Montagem da medição da característica do servo motor.

Com as ligações efetuadas, como indicado na Figura 20 ligou-se o Arduino, para aplicar escalões de tempo correspondentes às posições de 0° , 90° e 180° graus voltando a fazer os mesmos escalões no sentido inverso, com um intervalo de $1000\mu\text{s}$ entre cada um destes escalões.

No Canal 1, designado como CH1, do osciloscópio temos o valor da tensão aos terminais do motor DC que integra o servo motor com um ganho de 200mV/DIV e 500ms/DIV de escala temporal, no Canal 2 do osciloscópio designado como CH2, temos a leitura da tensão aos terminais do circuito integrado de controlo de posição do servo motor com um ganho de 500mV/DIV e 500ms/DIV , como indicado na Figura 21.

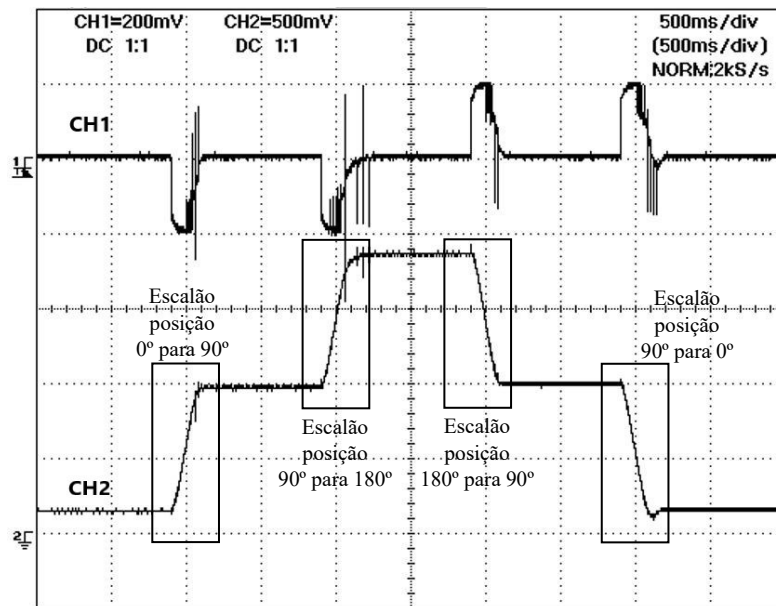


Figura 21 - Ensaio de posição do servo motor.

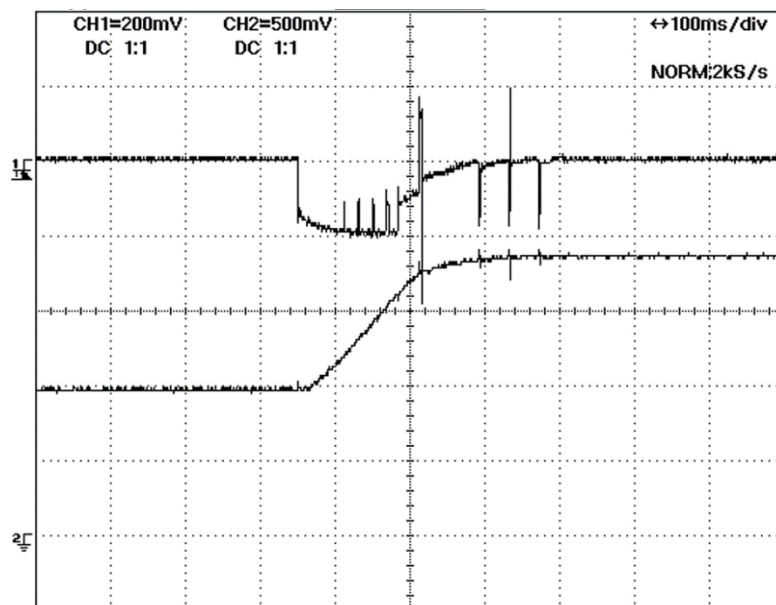


Figura 22 - Escalão de posição de 0° graus para 90° do ensaio experimental do servo motor.

Como indicado na Figura 22 observa-se um zoom na escala temporal para 100ms/div, para quando se aplica um escalão de posição de 0° para 90°.

Com a ideia de aproximar o sistema a uma dinâmica de primeira ordem, foi feita um análise com mais detalhe ao tempo de resposta do escalão de posição. Para isso temos que obter a constante de tempo τ , para 63,2% do ganho observado.

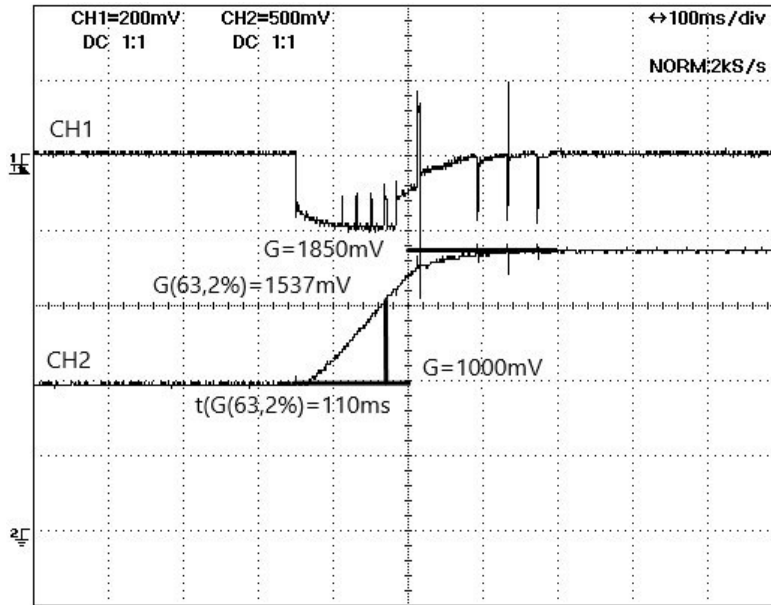


Figura 23 - Análise da constante de tempo τ e ganho, para a tensão aos terminais do servo motor, do ensaio de escalão posição de 0° a 90° .

Da análise do gráfico obteve-se a constante de tempo (τ), que é de 110ms para o tempo de resposta e 1000mV para o ganho, como indicado na Figura 23.

Ao analisar a curva representada pelo CH2 do osciloscópio, observa-se que tem um comportamento diferente de uma função de transferência de primeira ordem. Mas sendo que o objetivo é poder usar-se os controladores do tipo PID e análise de diagrama de blocos, com sistemas lineares, fez-se uma aproximação a uma curva de primeira ordem.

Partindo da equação genérica de um sistema de primeira ordem.

$$\frac{\theta_m(s)}{V_{trans}(s)} = \frac{K}{\tau s + 1} \quad (1)$$

$$\theta_m(s) = \frac{1,065}{0,11s + 1} V_{trans}(s) \quad (2)$$

O ganho foi obtido após ajuste empírico, tendo sido ajustado após algumas iterações do modelo, de maneira a ser o mais próximo da realidade possível.

Com a função de transferência obtida, simulou-se o modelo do servo motor no *software Matlab-Simulink*:

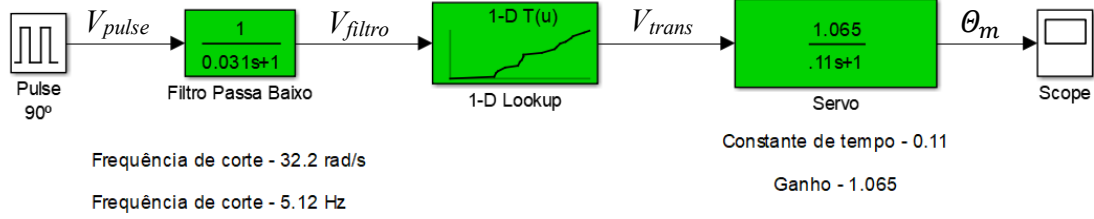


Figura 24 - Diagrama de blocos utilizado na obtenção do modelo do servo motor.

Com a intenção do modelo ser o mais aproximado possível do servo motor o sinal de controlo foi aplicado em forma de onda PWM, em vez de um escalão, como é feito na comunicação entre o controlador e o servo motor, como indicado na Figura 24. Com a intenção de controlar e simular o sinal enviado ao servo pelo Arduíno, observou-se um problema na simulação. A onda PWM sendo uma onda quadrada não é possível controlar o servo com a mesma, visto procurar um comportamento errático.

De maneira a poder fazer-se uma transdução do sinal de uma onda quadrada para um valor médio de tensão, aplicou-se um filtro passa-baixo, como indicado na equação 3. Filtro esse dimensionado de maneira empírica observando a resposta do modelo ao sinal em forma de PWM, tentando que a resposta fosse o mais linear possível.

$$V_{filtro}(s) = \frac{1}{0,031s+1} V_{pulse} \quad (3)$$

Analisando-se o filtro através de um diagrama de Bode, para obter-se a resposta em frequência.

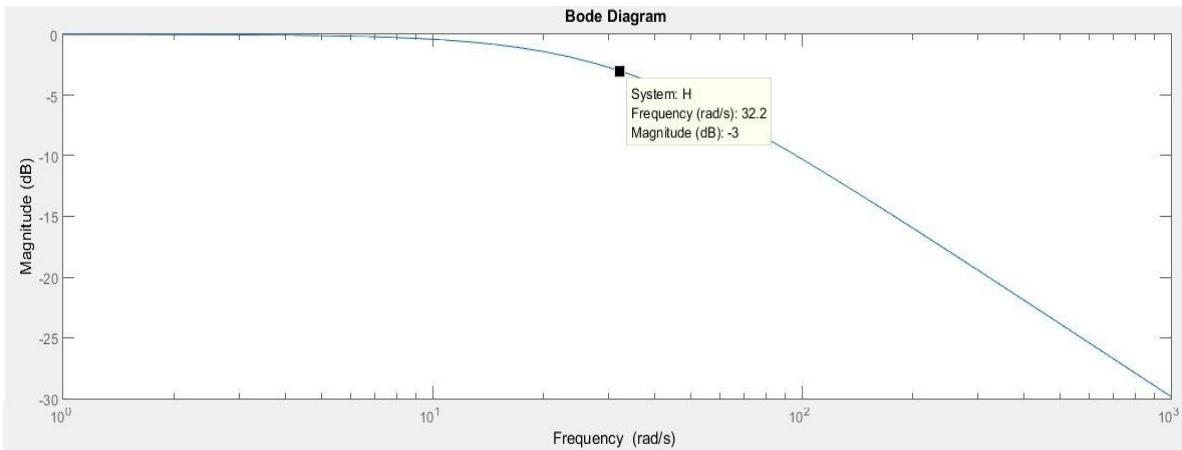


Figura 25 - Diagrama de Bode do filtro passa baixo.

Como indicado na Figura 25, observa-se que para um decaimento de -3dB, a frequência de corte é 32,2 rad/s.

Fazendo-se uma conversão de rad/s para Hz, como indicado na equação 4.

$$f_{(\text{Hz})} = \frac{\omega \left(\frac{\text{rad}}{\text{s}}\right)}{2\pi} \quad (4)$$

Substituindo na fórmula, o valor obtido através do diagrama de Bode:

$$f_{(\text{Hz})} = \frac{32,2}{2\pi} = 5,12 \text{ Hz} \quad (5)$$

Após o dimensionamento do filtro ajustou-se o ganho da função de transferência do servo motor para obter um valor correspondente ao deslocamento imposto pelo impulso do PWM, como indicado na equação 6.

$$\theta_m(s) = \frac{1,065}{0,11s+1} V_{trans}(s) \quad (6)$$

Analisando então a curva obtida no ensaio feito ao servo motor para o escalão posição de 0° para 90° , com a curva obtida para a simulação em Matlab.

Ensaio experimental para escalão de posição 0° para 90° .

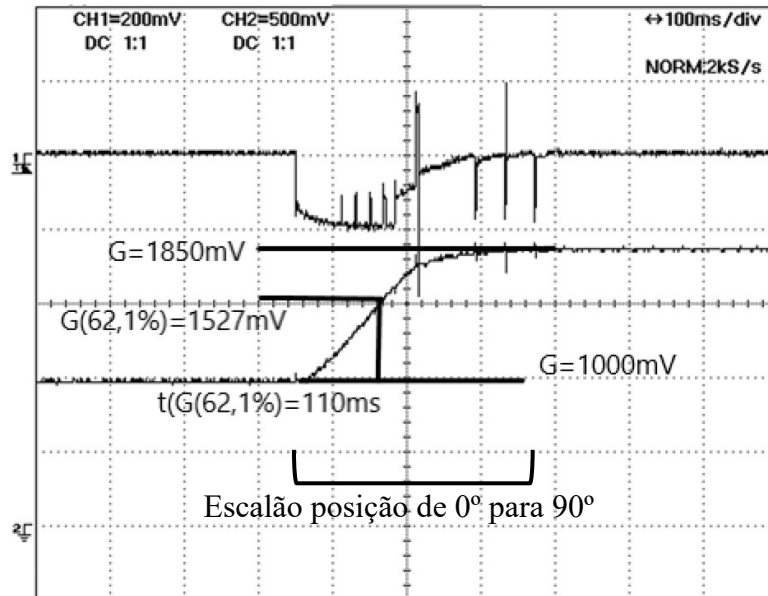


Figura 26 - Resultado experimental do ensaio da aplicação do escalão posição de 0° para 90° no servo motor.

Ensaio em simulação *Matlab* para escalão de posição 0° para 90° .

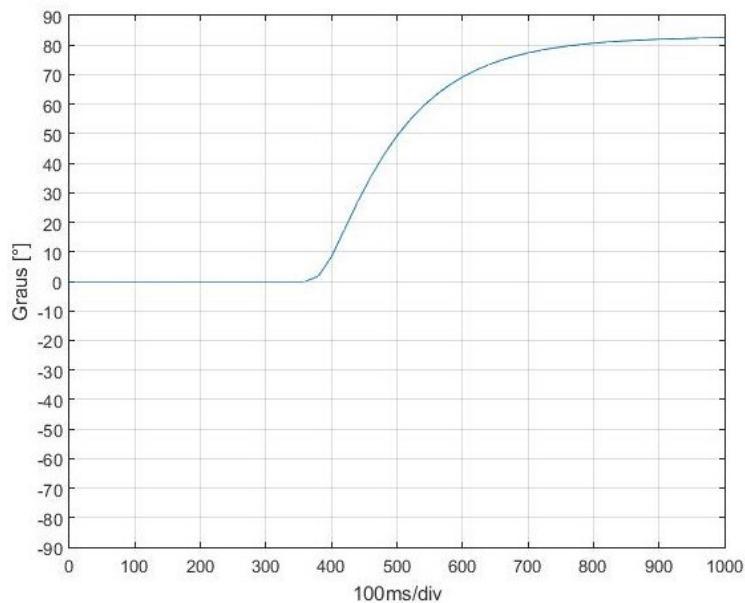


Figura 27 – Resultados da simulação para modelo do servo motor quando aplicado o escalão posição 0° a 90° .

Da comparação entre a diferença entre o ensaio experimental ao servo como indicado na Figura 26 e o resultado da simulação do modelo do servo na Figura 27, observa-se um

comportamento semelhante em termos de característica da curva e comportamento perante o mesmo tipo de escalão. Observa-se ainda que há um aumento no tempo de resposta devido não só á aproximação a uma resposta de uma função de transferência de primeira ordem, mas bem como a implementação do filtro passa baixo que provoca um atraso no modelo obtido.

Tendo em conta estes fatores, observa-se uma validação em termos de modelo obtido versus o componente para o qual se quer criar o modelo.

3.6 Ensaio para Obtenção da Característica do Sensor

Após a caracterização e validação do modelo do servo motor pela posição relativa à escala no topo do disco de acrílico, foi possível proceder com o dimensionamento do sensor com base na rotação do motor. O sensor foi montado no topo do disco, alinhado com o eixo de rotação do servo motor. Para este estudo, o eixo escolhido foi o eixo Z devido à orientação do sensor.

Com o objetivo de estudar as características do sensor, foi executado o mesmo programa utilizado para estudar as características do servo motor. Dessa forma, foi possível obter os mesmos sinais de controlo e rotações angulares, evitando assim qualquer desvio em relação ao que foi feito anteriormente.

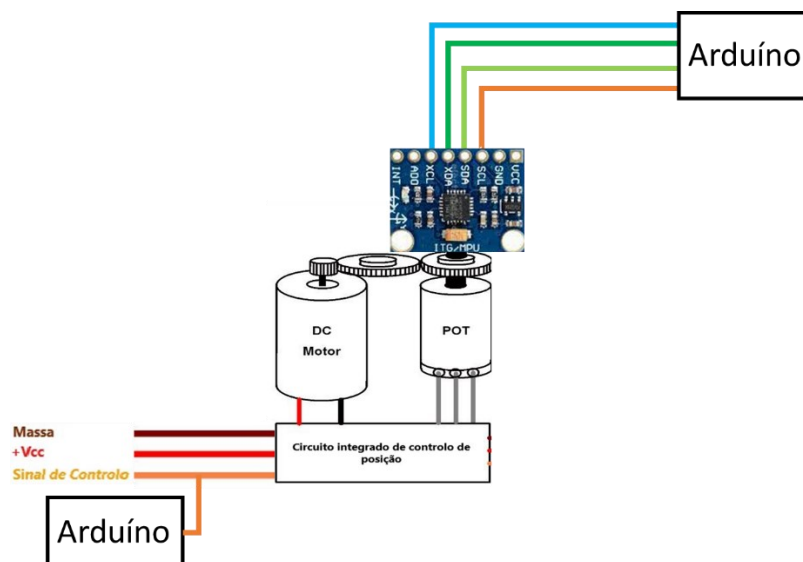


Figura 28 - Esquema de ligações do ensaio da característica do sensor.

3.7 Cálculo do Ângulo do Sensor

O MPU (Motion Processor Unit), é um dispositivo eletrônico que mede e fornece informações sobre uma força específica, um ângulo e, às vezes, um campo magnético em torno do corpo em que está aplicado. Essas informações são obtidas através da combinação de acelerômetros e giroscópios. Os MPU são normalmente utilizados em aplicações de aviação comercial, aeroespacial ou satélites.

Essas unidades de processamento de dados obtêm informações detetando a aceleração linear provocada por um deslocamento, utilizando um ou mais acelerômetros, bem como a rotação do corpo utilizando um ou mais giroscópios. O MPU é capaz de medir e interpretar esses dados de maneira a fornecer informações precisas sobre a posição, movimento e orientação do corpo em que está instalado.

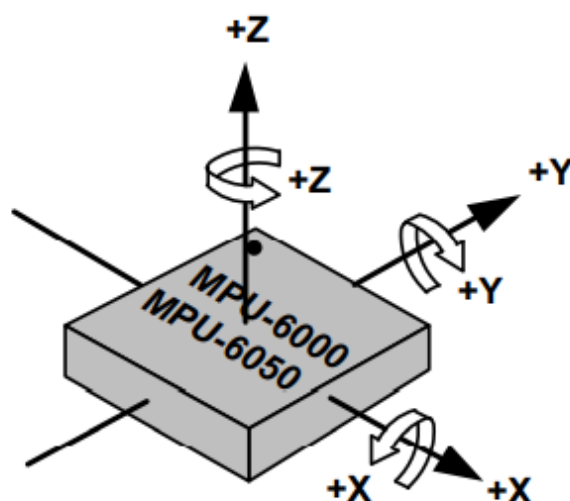


Figura 29 – Demonstração gráfica do chip com os eixos X, Y e Z, adaptado do [Anexo B].

A unidade de medida escolhida para medir o ângulo de rotação do servo motor é o MPU-6050, é uma unidade de obtenção dados de movimento de baixo custo e baixo consumo de energia. Composta por seis graus de liberdade, esta unidade é composta por um acelerómetro nos eixos X, Y e Z e um giroscópio nos mesmos eixos, como indicado na Figura 29. Além disso, o MPU-6050 também conta com um termómetro para leitura e compensação de erros, o que é especialmente importante, uma vez que as leituras podem ser afetadas pelo aumento da temperatura.

Outra característica importante desta unidade de medida é a sua capacidade de processamento de movimento digital, chamada de "*Digital Motion Processing*" (DMP). Com suporte de movimento e processamento em 3D, o MPU-6050 utiliza os valores do giroscópio e acelerómetro e sincroniza os dados em tempo real, permitindo o uso desta unidade para representações 3D em *software* como o *Processing*. Essa característica permite reduzir a carga de processamento do Arduíno, resultando em dados mais rapidamente tratados e em melhor resposta e fluidez do sistema.

O MPU-6050 ainda conta com uma série de *buffers* FIFO (*First In, First Out*), que processam as movimentações no espaço tridimensional de forma eficiente em termos energéticos, permitindo que seja utilizado de maneira portátil com uma bateria externa, sem necessidade de estar dependente da rede elétrica. Com essas características, o MPU-6050 é uma escolha eficiente e adequada para a medição do ângulo de rotação do servo motor nesta dissertação.

Outras atenuantes que limitam a obtenção do ângulo com mais precisão, são:

- O *drift* do giroscópio, que tende a haver um deslocamento que pode ser provocado pelas temperaturas e vibrações indesejáveis;
- Ruído nos acelerómetros, que são sensíveis a vibrações;
- Campos eletromagnéticos;
- Tem uma resolução relativamente baixa quando se tenta medir ângulos de deslocação pequenos, poderá ser uma limitação;
- Poderá apresentar alguma imprecisão aquando do cálculo do ângulo, porque não possui um magnetómetro podendo assim, perder o seu posicionamento.

Com estas condições em mente, tentou se implementar através de código usando os dados *raw*, diretamente do acelerómetro e giroscópio.

Equação para o *pitch*:

$$pitchAcc = arctan\left(ay \frac{ay}{\sqrt{(ax \times ax + az \times az)}}\right) \quad (7)$$

Da qual se traduz em código:

$$pitchAcc = atan2(ay, sqrt(ax \times ax + az \times az)) \quad (8)$$

Equação para o *roll*:

$$rollAcc = arctan\left(-\frac{ay}{\sqrt{(ay \times ay + az \times az)}}\right) \quad (9)$$

Da qual se traduz em código:

$$rollAcc = atan2(-ax, sqrt(ay \times ay + az \times az)) \quad (10)$$

Equação para o *yaw*:

$$yawAcc = arctan\left(\frac{ax}{ay}\right) \quad (11)$$

Da qual se traduz em código:

$$yawAcc = atan2(ax, ay) \quad (12)$$

A função $atan2(y,x)$ é uma função que retorna o ângulo em radianos entre o eixo X positivo e o ponto (x,y) no plano cartesiano. Esta função da tangente inversa, é útil quando se trabalha com coordenadas polares, pois retorna o ângulo no intervalo de $-\pi$ a π .

Ao tentar obter-se os valores do ângulo em torno do eixo Z, eixo para o qual acontece a rotação do servo motor e a rotação do sensor, observou-se que os mesmos estavam a ir de -90° a 90° . Constatando então uma medida não exata do ângulo de deslocação do servo motor. De maneira a evitar uma desfasagem de medidas e procurar uma melhor exatidão optou-se por fazer uso da característica DMP do sensor MPU6050. Do qual se obtém os ângulos ou rotações em torno do eixo através dos ângulos de Euler.

3.8 Ângulos de Euler

O propósito final desta dissertação é fazer o controlo, preservando ou conservando o estado inicial de um objeto a estabilizar no espaço de acordo com o sistema de referência. Para a representação da orientação no espaço existe os ângulos de Euler, sendo um recurso passível de ser utilizado através do MPU-6050.

Os ângulos de Euler, constituem um conjunto de três coordenadas angulares, que servem para demonstrar a orientação de um sistema de eixos ortogonais móvel no espaço, em relação a um eixo ortogonal fixo, usando para isso uma combinação de três rotações sobre os três diferentes eixos. Por convenção usa-se eixos de coordenadas distintos para descrever a orientação do MPU, posicionando o no centro do sistema de eixos de referência. Nesta dissertação apesar de não ser necessário efetuar os cálculos com as matrizes, decidiu-se estudar para melhor entender o seu funcionamento.

A parametrização de um sistema de 3 eixos, composto por ângulos de Euler utiliza o facto de que qualquer rotação pode ser descrita por três rotações sucessivas à volta dos eixos Ox, Oy e Oz.

Aos ângulos de Euler α , β e γ são atribuídas as matrizes compostas por:

$$R(\alpha, X) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \quad (13)$$

$$R(\beta, Y) = \begin{bmatrix} \cos \beta & 0 & -\sin \beta \\ 0 & 1 & 0 \\ \sin \beta & 0 & \cos \beta \end{bmatrix} \quad (14)$$

$$R(\gamma, Z) = \begin{bmatrix} \cos \gamma & \sin \gamma & 0 \\ -\sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (15)$$

Estas matrizes aparecem de forma intuitiva, das equações que definem as rotações à volta dos eixos coordenados. Uma rotação definida pelos ângulos de Euler (α, β, γ) resulta da factorização de três rotações sequenciais à volta dos três eixos de coordenadas.

Uma vez que as rotações à volta dos eixos não diferem, a ordem pela qual se faz as rotações é importante. Qualquer combinação de eixos pode ser utilizada para representar as rotações. Normalmente os ângulos de Euler, referem-se à formulação de um eixo móvel. No entanto, na perspectiva do nosso sistema é fixo.

Com as diversas convenções existentes e possíveis, foi adotado a convenção com base na aeronáutica.

- zyx - *Roll* (Ox), *Pitch* (Oy) e *Yaw* (Oz)

Esta convenção descreve os movimentos de um avião (Figura 30). Os aviões têm três graus de liberdade, i.e., possuem três eixos naturais de rotação, que são perpendiculares dois a dois:

- *Roll* que define a rotação que existe à volta da direção do avião;
- *Pitch* que demonstra uma rotação para cima ou para baixo;
- *Yaw* que é uma rotação da esquerda-direita ou direita-esquerda ao longo do eixo vertical.

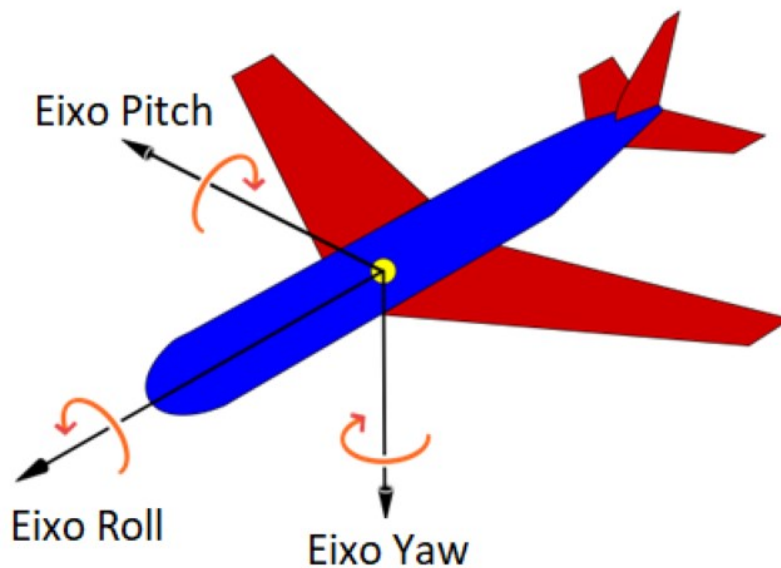


Figura 30 - Exemplo dos eixos utilizados como convenção na aviação adaptado de [22].

A sua composição e rotação, não é fácil, pois não é linear a soma dos ângulos, uma vez que todos os eixos sofrem alterações de rotação.

A escolha dos eixos e a ordem pela qual as rotações são feitas têm que ser respeitadas, visto que existem diversas convenções com os ângulos de Euler que poderão influenciar a composição da matriz final.

Vantagens:

- Fácil compreensão e intuitivos;
- A sua representação é natural e eficiente, pois utiliza três variáveis (ângulos), para representar os três graus de liberdade (3DOF). Assim não apresentando quaisquer restrições, não necessitando de reajustamentos;
- Forma simples de realizar rotações em torno de eixos coordenados móveis.

Desvantagens:

- Grande parte das vezes, não existe uma decomposição lógica/evidente da rotação geral para as três rotações feitas em torno dos eixos. Se, se quiser uma representação computacional 3D, e se efetuar uma rotação de $\pi/6$ à volta de um eixo de rotação dado por um vetor $(1,1,1)$, é bastante difícil determinar os ângulos correspondentes para os três eixos de base;
- Não existe sempre uma forma de representar uma rotação com ângulos de Euler, que correspondam a uma série de rotações concatenadas;
- Havendo uma interpolação ligeira entre duas orientações envolve uma integração numérica, o que computacionalmente pode ser dispendioso. A interpolação apresenta ainda dificuldades por existir várias formas de definir a mesma rotação.
- Mas a principal desvantagem dos ângulos de Euler, é a possível perda dos graus de liberdade (DOF), quando se realiza uma rotação á volta dos eixos. Esta singularidade das coordenadas é designada por *gimbal Lock*.

3.9 Gimbal Lock

O *gimbal lock*, ocorre em diversos casos, assumindo por exemplo, quando se considera o sistema de coordenadas na primeira Figura 31, após se realizar uma rotação de $\pi/6$ em torno do eixo Ox , obtêm-se a segunda imagem da figura, se agora se realizar uma rotação de $\pi/2$ em torno do eixo Oy , o que resulta na terceira imagem da Figura 31, como se pode observar, o eixo Oz está alinhado com o eixo Oxo . Com isto, ocorreu o *gimbal lock*, qualquer rotação em torno do eixo Oz afeta agora o mesmo grau de liberdade do eixo Ox .

Assim perdeu-se a capacidade de efetuar as rotações em torno de um terceiro grau de liberdade.

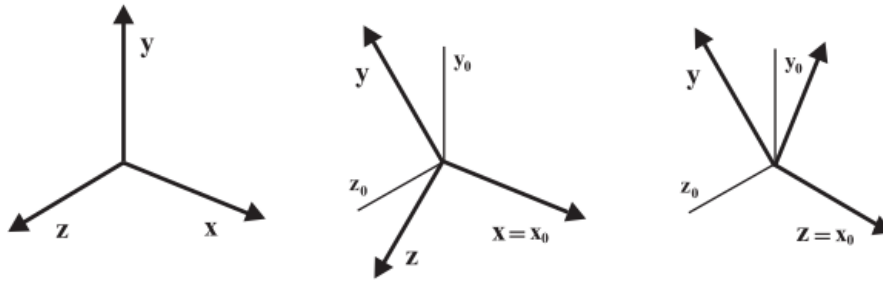


Figura 31 - Exemplo de Sistema de coordenadas para o qual acontece o gimbal lock.

O *gimbal lock*, é a razão pela qual a indústria aeroespacial foi uma das primeiras a utilizar os quatérnios, para fazer uma representação das orientações, de satélites, foguetes e aviões.

Exemplificando o problema de modo matricial, numa sequência de rotações, a primeira rotação feita é do ângulo α , em volta do eixo Ox , a segunda rotação é de $\pi/2$ em torno do eixo Oy , e a terceira de γ em torno do eixo Oz . A rotação que ocorreu em torno do eixo Oy , causou um alinhamento entre os eixos Ox e Oz , provocando a perda de um grau de liberdade da rotação, o que provoca uma perda de um grau de liberdade na matriz genérica de rotação:

$$\begin{aligned}
 R_{O_x O_y O_z}(\alpha, \pi/2, \gamma) &= R_{O_z}(\gamma)R_{O_y}(\pi/2)R_{O_x}(\alpha) = & (16) \\
 &= \begin{bmatrix} 0 & \cos \gamma \sin \alpha - \sin \gamma \cos \alpha & \cos \gamma \cos \alpha + \sin \gamma \sin \alpha \\ 0 & \sin \gamma \sin \alpha + \cos \gamma \cos \alpha & \sin \gamma \cos \alpha - \cos \gamma \sin \alpha \\ -1 & \cos \beta \sin \alpha & \cos \beta \cos \alpha \end{bmatrix} = \\
 &= \begin{bmatrix} 0 & \sin(\alpha - \gamma) & \cos(\alpha - \gamma) \\ 0 & \cos(\alpha - \gamma) & -\sin(\alpha - \gamma) \\ -1 & 0 & 0 \end{bmatrix}
 \end{aligned}$$

Com a rotação $R(\alpha, \pi/2, \gamma)$, consegue-se verificar que esta somente depende da diferença de α para γ , assim com a expressão a depender apenas de α e γ perdeu-se um grau de liberdade, para $\theta = \pi/2$, os ângulos α e γ , resultam de rotações á volta do mesmo eixo.

Devido ao tipo de montagem utilizado nesta dissertação, apenas será possível trabalhar com um eixo de cada vez, tendo sido escolhido o eixo Z, podendo após confirmação e validação dos modelos ser replicado para os restantes eixos.

O controlador executará o programa de controlo do servo motor para as posições predefinidas de 0° , 90° e 180° , e em seguida, para as posições de 180° , 90° e 0° , com intervalos de $1000\mu\text{s}$ entre cada posição, como indicado na Figura 32.

Observou-se assim com os valores obtidos no ensaio no gráfico abaixo:

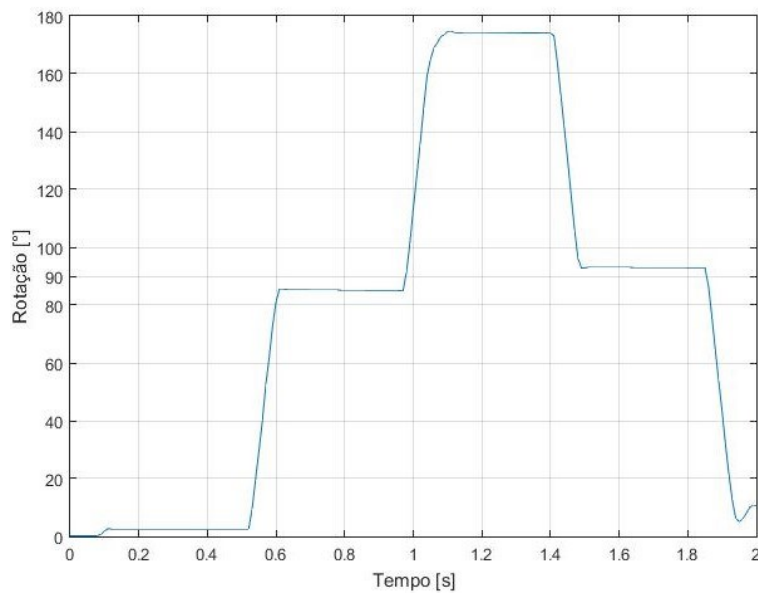


Figura 32 - Ensaio experimental para obtenção da característica do modelo do sensor.

Na Figura 32, observa-se o comportamento do sensor para os diversos escalões posição.

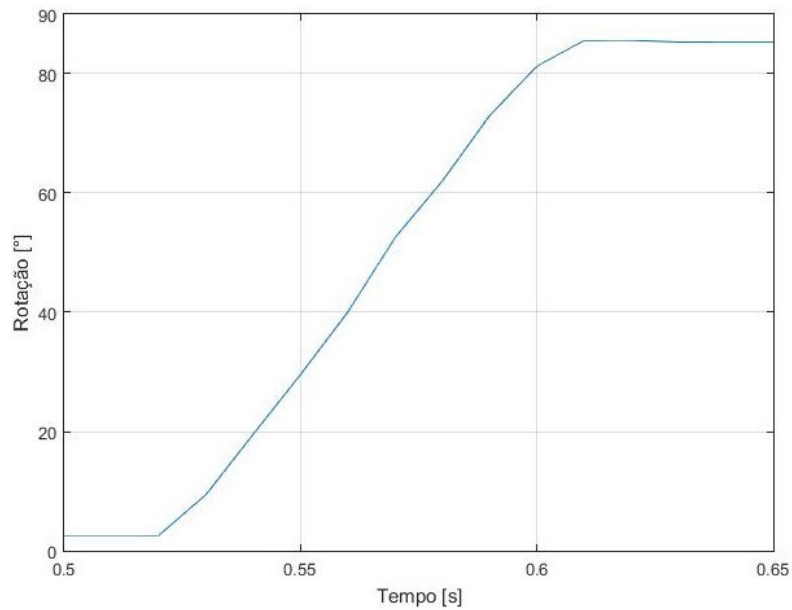


Figura 33 – Ampliação do ensaio experimental para obtenção da característica do modelo do senso, para o escalão posição 0° a 90°.

Usando a Figura 33, e utilizando a mesma metodologia do ensaio para obtenção do modelo do servo motor, pode-se então fazer a análise dos gráficos obtidos com os dados provenientes do sensor. Da análise dos gráficos obteve-se o τ , para 63,2% do ganho obtido no escalão posição. Sendo o nosso ganho desde 2,6° até 85,62°, o momento em que se atinge os 63,2% é 0,049s ao qual corresponde a constante de tempo τ , que faz coincidir com 52,46°.

Ao qual se obteve a função de transferência do sensor aproximada a primeira ordem.

$$\theta_s(s) = \frac{1}{0,049s+1} \theta_m(s) \quad (17)$$

Com base na função de transferência criou-se o modelo de simulação em *Matlab*. Tendo como base o ensaio que se realizou para o servo motor, como indicado Figura 34.

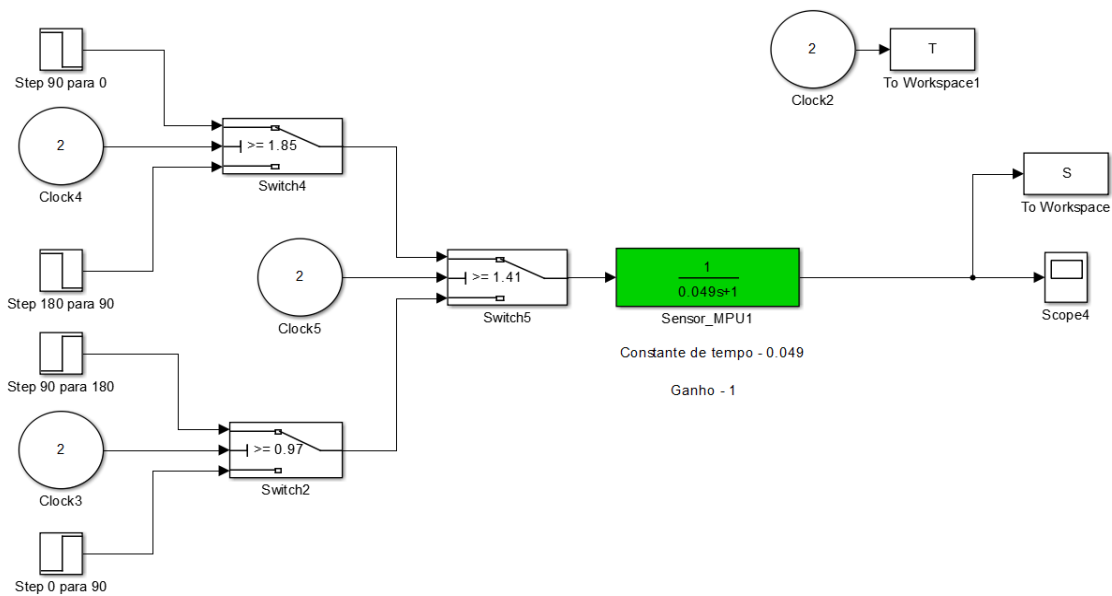


Figura 34 - Diagrama de blocos do modelo do sensor.

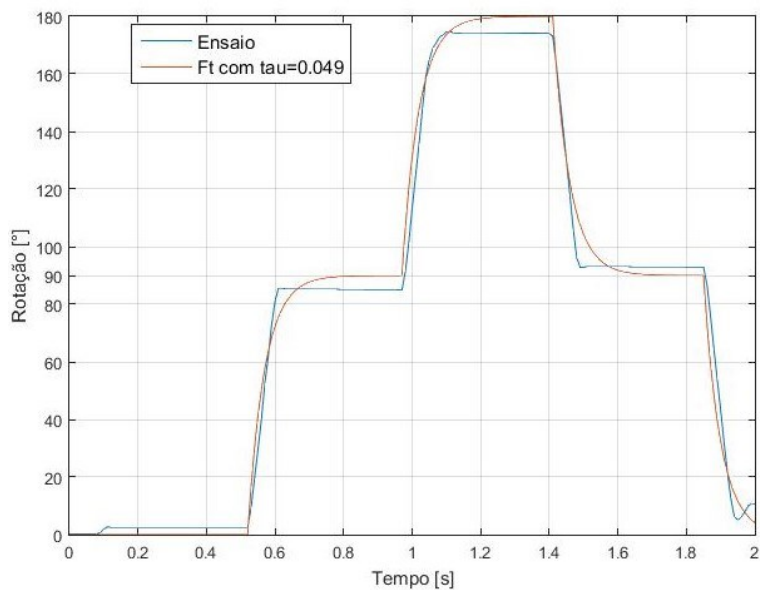


Figura 35 - Comparação entre modelo teórico do sensor e modelo do ensaio experimental.

A partir da observação da Figura 35, é possível notar um arredondamento da curva nos patamares de 90° e 180°. Com base nisso, pode-se concluir que a constante de tempo de 0,049s, obtida anteriormente, pode necessitar de um ajuste fino para se obter uma melhor correspondência com o resultado do ensaio. Esse ajuste é necessário para garantir a precisão da medição e a confiabilidade dos resultados obtidos.

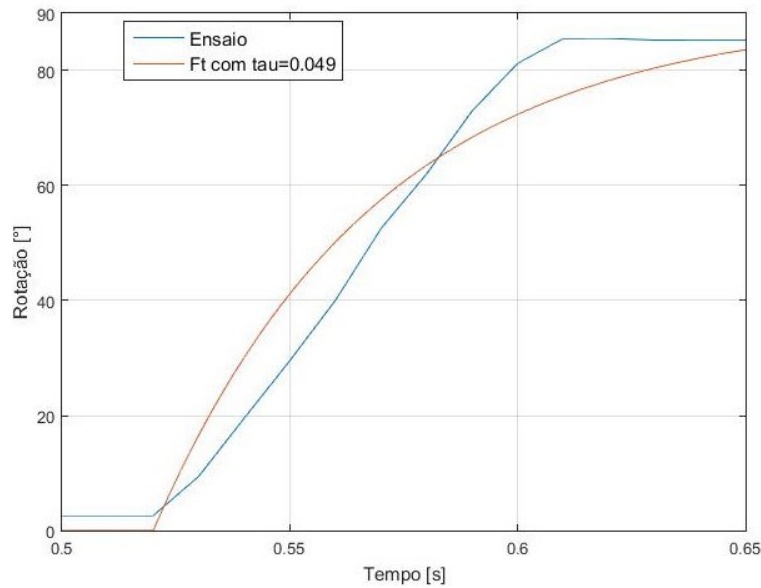


Figura 36 - Comparação entre modelo teórico do sensor e ensaio experimental no escalão posição (0° aos 90°).

Ao analisar a Figura 36, que representa a simulação do momento em que se aplica o escalão posição para a rotação do sensor entre 0° e 90°, é possível notar um pequeno desfasamento em relação ao resultado obtido no ensaio experimental. Na curva obtida pelos dados experimentais, representada em azul, a derivada em relação ao tempo é praticamente constante, o que resulta em uma representação gráfica semelhante a uma rampa. Por outro lado, a derivada em relação ao tempo de uma função de transferência de primeira ordem não é constante e, após atingir 63,2% do seu ganho, tem um crescimento que se assemelha a uma exponencial. Isso indica que a constante de tempo de 0,049s, obtida anteriormente, pode precisar de ajustes finos para melhor se aproximar aos resultados do ensaio experimental.

Após a análise comparativa desses ensaios, optou-se por realizar um ajuste com base nos resultados experimentais. Esse ajuste consistiu em modificar o valor da constante de tempo τ e do ganho para obter resultados do modelo teórico mais semelhantes aos do ensaio experimental.

Ao qual se obteve a função de transferência:

$$\theta_s(s) = \frac{0,95}{0,0159s+1} \theta_m(s) \quad (18)$$

Com esta função de transferência reproduziu-se o modelo em diagrama de blocos no software Matlab-Simulink:

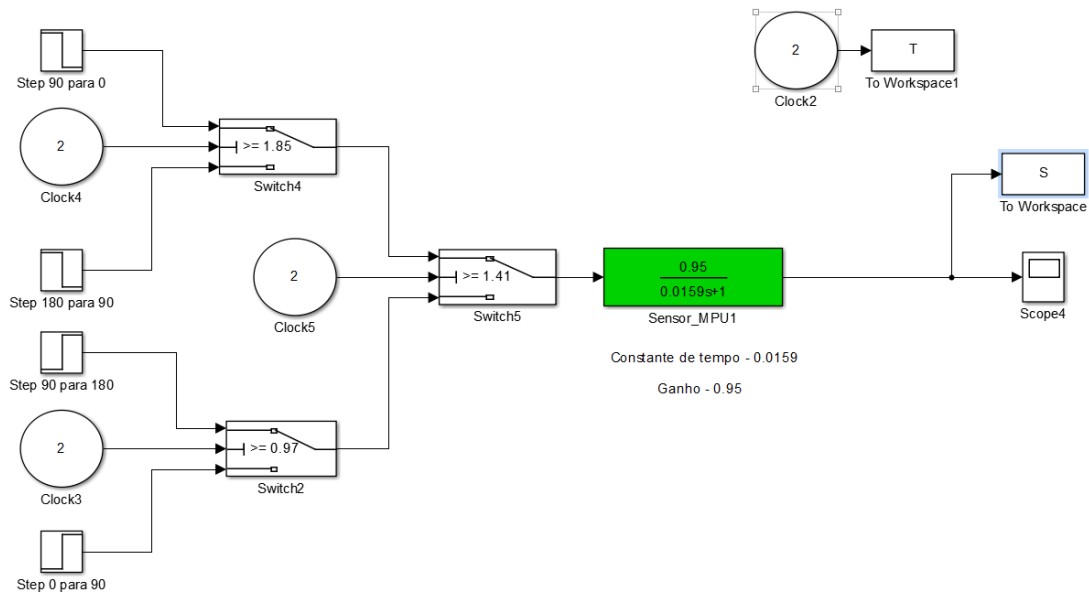


Figura 37 - Diagrama de blocos do modelo do sensor para os parâmetros ajustados ($G=0,95$ e $\tau = 0,0159s$).

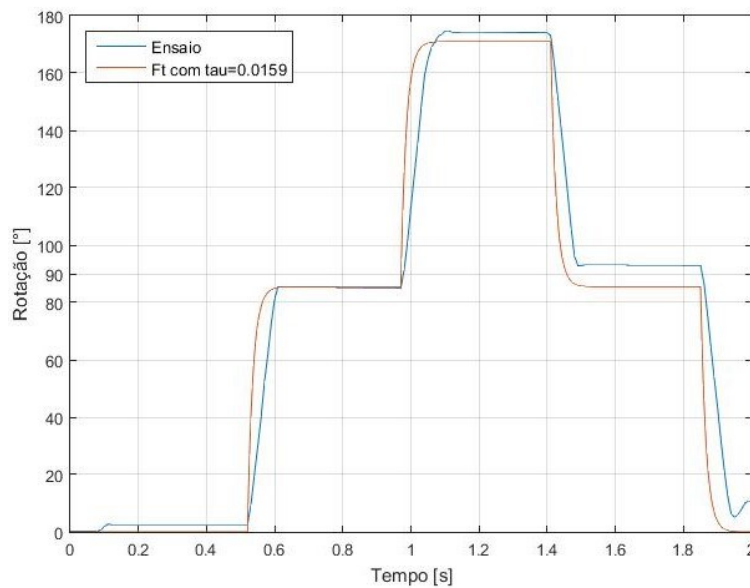


Figura 38 - Comparação entre ensaio experimental e modelo do sensor com parâmetros ajustados ($G=0,95$ e $\tau = 0,0159s$).

Como indicado na Figura 38, observa-se a diferença entre o ensaio experimental com a simulação do modelo teórico do servo motor.

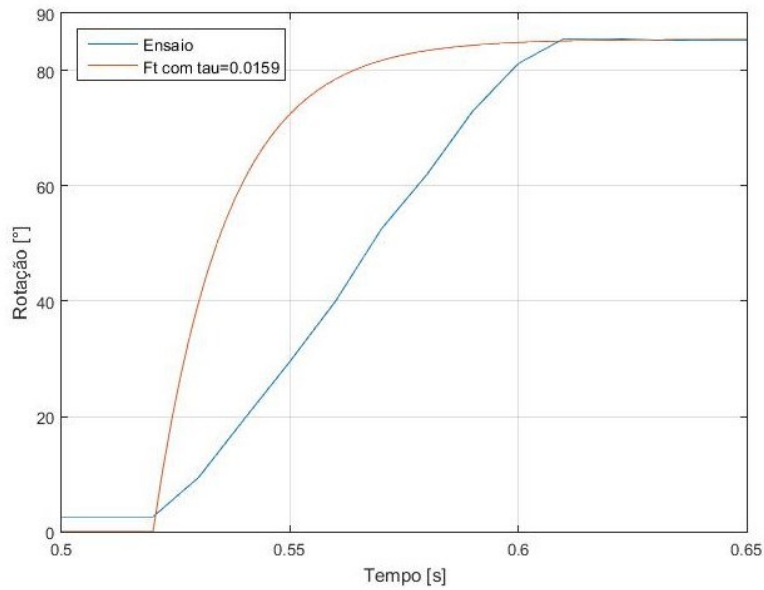


Figura 39 - Comparação entre ensaio do escalão posição dos 0° aos 90° e modelo sensor ajustado ($G=0,95$ e $\tau=0,0159s$).

Ao analisar as Figura 38 e a Figura 39, é possível observar que, após realizar um ajuste fino do ganho, foi possível obter um valor de ângulo equivalente ao obtido pelo sensor no ensaio experimental. Além disso, realizando o ajuste da constante de tempo τ para 0,0159s, foi possível perceber que a função de transferência apresentou um comportamento semelhante em relação à dinâmica da subida e ao tempo de estabilização do ângulo desejado. No entanto, ao tentar aproximar a simulação teórica representada na Figura 39 a uma resposta de 1ª ordem, constatou-se que há uma pequena diferença em relação aos resultados obtidos experimentalmente.

Com o objetivo de que a função de transferência seja o mais próxima possível da realidade introduziu-se um atraso no processo.

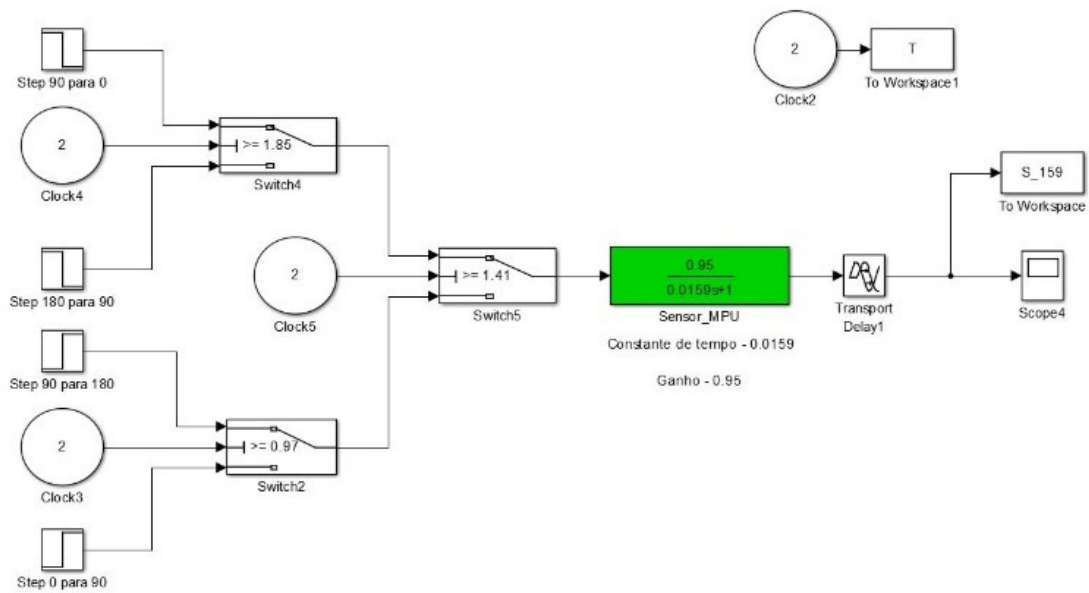


Figura 40 - Diagrama de blocos do modelo do sensor ajustado com atraso ($G=0,95$, $\tau=0,0159$ e atraso de $0,039$ s).

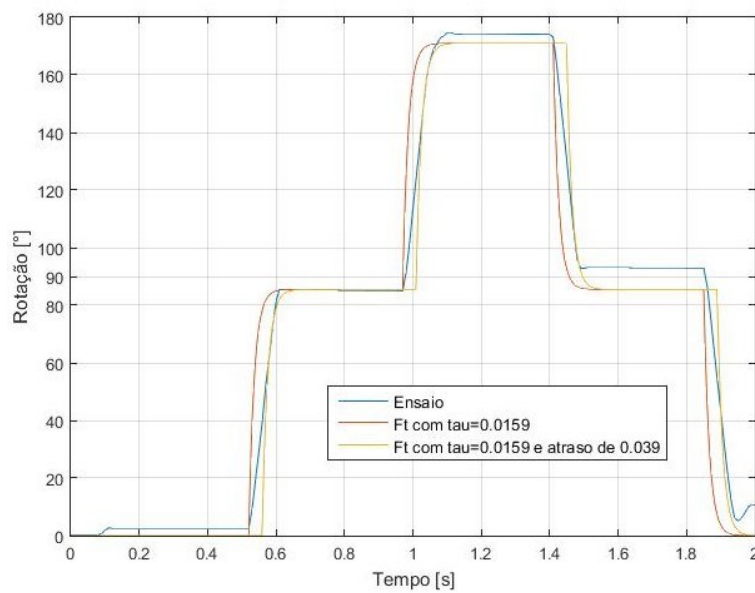


Figura 41 - Comparação entre ensaio experimental e modelo do sensor com e sem atraso ($G=0,95$, $\tau=0,0159$ e atraso $0,039$ s).

Ao observar a Figura 41, é possível verificar os resultados das iterações realizadas para uma aproximação do modelo do sensor aos resultados experimentais. Nesta figura, a linha azul representa os resultados obtidos a partir do ensaio experimental, a linha laranja representa o ensaio para o valor de τ de $0,0159$ s, e a linha amarela representa a simulação com o mesmo valor de τ com um atraso de $0,039$ s.

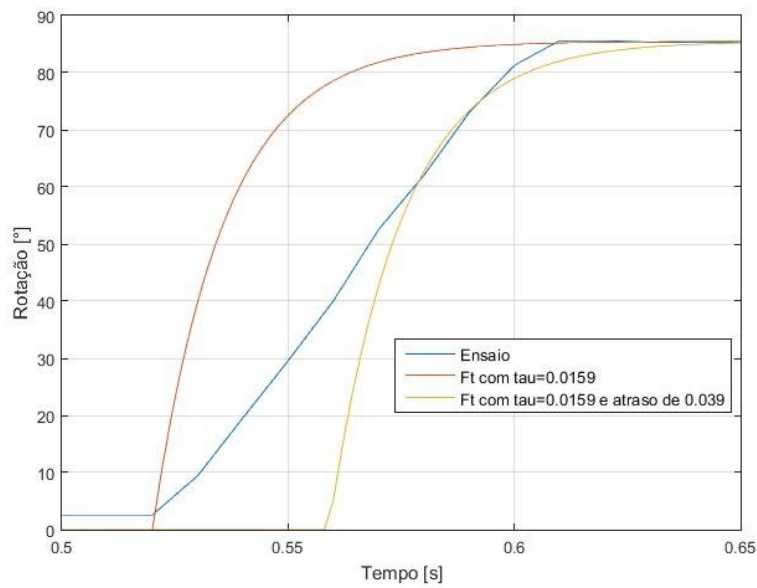


Figura 42 - Comparação entre ensaio experimental para o escalão posição dos 0° aos 90° e modelo do sensor com e sem atraso ($G=0,95$, $\tau =0,0159s$ e atraso de $0,039s$).

Após a implementação do atraso, houve uma melhoria no alinhamento com a curva obtida no ensaio experimental, porém, como já mencionado anteriormente, ao fazer-se uma aproximação do modelo teórico a uma função de transferência de 1ª ordem, é possível perder algumas características, o que pode resultar num modelo pouco preciso. Porém, apesar dessa limitação, a aproximação permite o uso de um controlador do tipo PID, o que pode ser benéfico em aplicações práticas.

4 Implementação e Ensaios Experimentais

4.1 Método de Ziegler-Nichols

Após a validação dos modelos em Matlab pela parte dos ensaios experimentais, o próximo passo é abordar o controlo do processo. Para essa tarefa, será utilizado o método de Ziegler-Nichols do ganho crítico ou *continuous cycling method*, que é amplamente utilizado na indústria, especialmente em processos em que a dinâmica de alguns dos processos ainda não é totalmente conhecida. Este método consiste em encontrar o valor limite do ganho proporcional, para o qual o sistema começa a oscilar, e em seguida, determinar os parâmetros do controlador PID com base nesse valor crítico (K_{cr}).

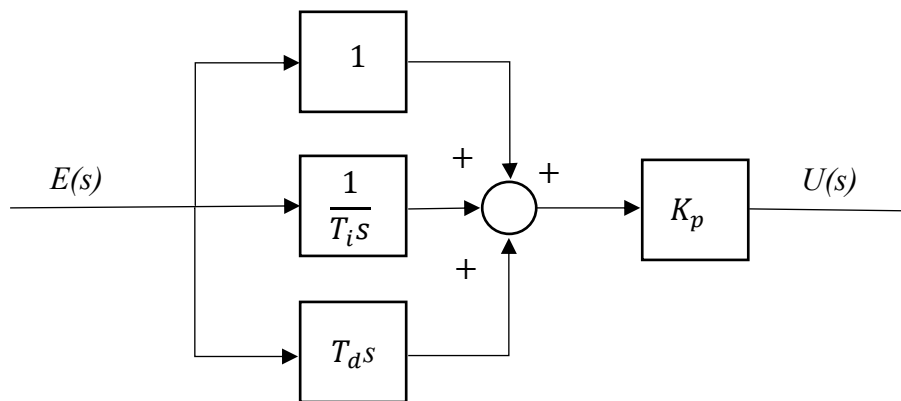


Figura 43 - Diagrama de blocos para controlador PID, usando o método de Ziegler-Nichols [23].

Este método tem como vantagens:

- Simplicidade, o facto de ser muito intuitivo é uma das características que mais abonam a favor deste método;
- Os controladores obtidos através deste método têm um bom desempenho em processos de *loops* simples;
- Fácil aplicabilidade sem ser precisos grandes conhecimentos da dinâmica do processo em si, o que no estudo desta dissertação poderá estar implícito alguma fase do processo.

Desvantagens:

- Necessidade de alguma sintonia fina depois de calculado os elementos necessários;
- Este método tem como princípio de “obrigar” o sistema a ficar instável, o que para situações que sejam práticas/reais, como motores a movimentarem grandes cargas, pode-se tornar perigoso, sendo que se está em ambiente de simulação, esta situação é evitada.

Neste método temos como conceito eliminar as ações integral (K_i) e derivativa (K_d) do controlador PID.

Com este procedimento obriga-se com que os polos complexos dominantes do sistema fiquem próximos do semiplano direito, fazendo como sua consequência a instabilidade do sistema [23].

De modo a obter o ganho crítico aumentou-se lentamente o valor do ganho proporcional (K_p) na simulação em *Matlab*, até se observar o limite que o sistema entra em instabilidade, que faz corresponder a uma oscilação permanente de amplitude constante. Posteriormente com o valor do ganho para o qual o sistema entra em instabilidade foram determinados os parâmetros de acordo com o método de Ziegler-Nichols, para isso existem umas relações propostas pelos autores para determinar os parâmetros do tipo de controlador que desejamos, como indicado na Tabela 1.

Tabela 1 - Relações de Ziegler-Nichols (Original) [24]

Tipo de Controlador	K_p	T_i	T_d
P	$0,5 K_{cr}$	∞	0
PI	$0,45 K_{cr}$	$0,83 P_{cr}$	0
PID	$0,6 K_{cr}$	$0,5 P_{cr}$	$0,125 P_{cr}$

Para os parâmetros do processo utilizou-se como base uns parâmetros de uma composição do modelo da mão humana, baseada no artigo “Active force control system for hand

tremor suppression by different actuators” [20], o parâmetro utilizado foi o do tecido subcutâneo, em que a massa é de 0,566 kg, o atrito 0,1 Ns/m e a mola é 23,9 N/m.

Para achar o ganho crítico (K_{cr}), procedeu-se ao ensaio do aumento do ganho proporcional, até atingir a instabilidade crítica, sendo depois esse o valor utilizado na Tabela 1, para achar as restantes constantes. Com o valor do ganho crítico obtido, pode-se obter o valor do período crítico (P_{cr}), que consiste em retirar o período da onda, para o qual a mesma está no ponto de instabilidade crítica.

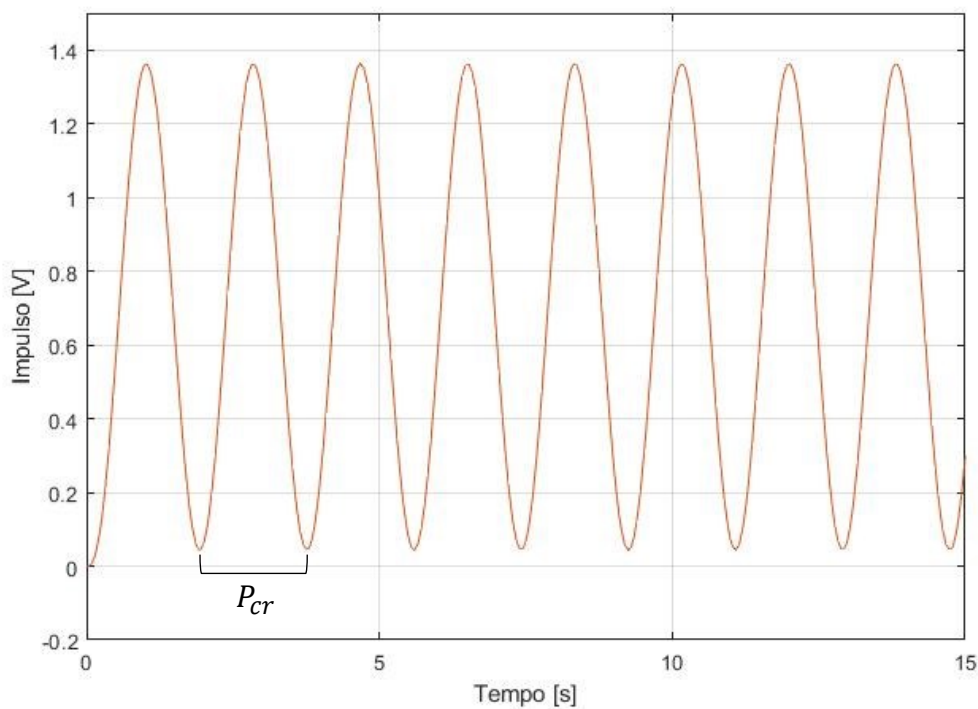


Figura 44 – Onda no limite da estabilidade para quando o ganho do controlador é $K_{cr}=0,9$.

Com o aumento do parâmetro K_p para atingir uma instabilidade do sistema, observou-se que o ganho crítico é $K_{cr} = 0,9$, como $K_{cr} = K_p$, substituiu-se.

$$K_p = 0,6K_{cr} = 0,54 \quad (19)$$

Como indicado na Figura 44, observando o sinal de tensão à saída do sistema. Da análise do gráfico para o valor de K_{cr} crítico, obteve-se um período crítico P_{cr} de 1,832s.

Com este resultado obteve-se o valor de K_i e K_d :

$$K_i = 1,2 \frac{K_{cr}}{P_{cr}} = 1,2 \frac{0,9}{1,832} = 0,589 \quad (20)$$

$$K_d = 0,075 \frac{K_{cr}}{P_{cr}} = 0,075 \frac{0,9}{1,832} = 0,0368 \quad (21)$$

Após o cálculo dos valores para o controlador PID, obteve-se o seguinte comportamento, para um impulso de 5V:

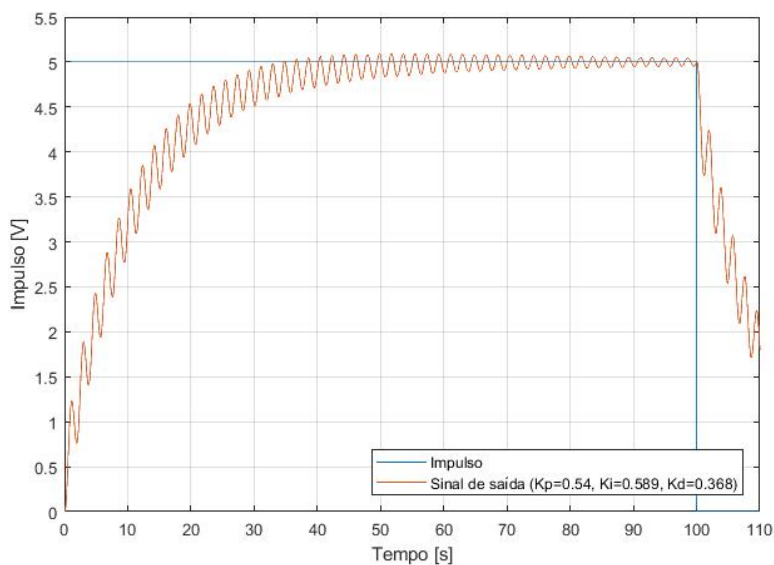


Figura 45 – Sinal de saída com controlador PID, ($K_p=0,54$, $K_i=0,589$, $K_d=0,368$).

Através da análise do sinal de tensão na saída do sistema, foi possível obter a Figura 45 como referência para ajustes mais precisos dos parâmetros do controlador. O objetivo é otimizar o comportamento do controlador PID de modo a ter um comportamento mais suave. Para realizar essa otimização, foi utilizado o recurso *tune* do *Matlab*, resultando nos valores de $K_p=0$, $K_i=0,605$, $K_d=0$. Com esses valores ajustados, o controlador apresentou um comportamento suave, conforme pode ser observado na Figura 46.

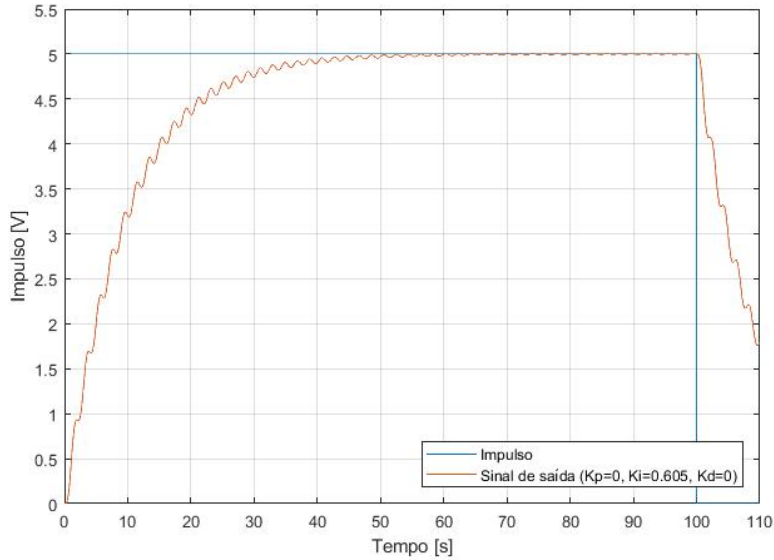


Figura 46 – Sinal de saída com controlador PID ajustado, ($K_p=0$, $K_i=0,605$, $K_d=0$).

A Figura 46 é obtida observando o sinal de tensão á saída do sistema. Após experimentar os valores obtidos para o método original de Ziegler-Nichols, decidiu-se experimentar para os valores de um controlador com pequena sobrelevação e sem sobrelevação [25].

Tabela 2 - Relações de Ziegler-Nichols (Original, pequena sobrelevação, sem sobrelevação) [25].

Tipo de Controlador (PID)	K_p	T_i	T_d
Ziegler-Nichols Original	$0,6 K_{cr}$	∞	0
Ziegler-Nichols Pequena sobrelevação	$0,33 K_{cr}$	$0,5 P_{cr}$	0
Ziegler-Nichols Sem sobrelevação	$0,2 K_{cr}$	$0,5 P_{cr}$	$0,33 P_{cr}$

Calculando os valores com base da tabela para o método de Ziegler-Nichols para pequena sobrelevação tem-se:

$$K_p = 0,33 K_{cr} = 0,297 \quad (22)$$

$$K_i = 0,66 \frac{K_{cr}}{P_{cr}} = 0,66 \frac{0,9}{1,832} = 0,324 \quad (23)$$

$$K_d = 0,11 \frac{K_{cr}}{P_{cr}} = 0,11 \frac{0,9}{1,832} = 0,054 \quad (24)$$

Substituindo esses valores no modelo do sistema, observa-se então o comportamento do controlador PID para os valores de K_p , K_i e K_d para um controlo com pequena sobrelevação.

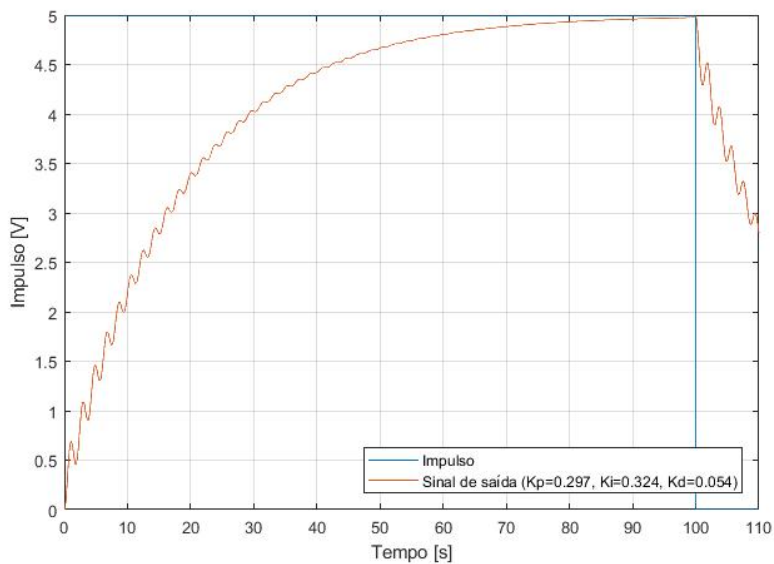


Figura 47 - Sinal de saída com controlador PID, com pequena sobrelevação ($K_p=0,297$, $K_i=0,324$, $K_d=0,054$).

A Figura 47 é obtida observando-se o sinal de tensão na saída do sistema.

Calculando os valores com base na Tabela 2 do método de Ziegler-Nichols para sem sobrelevação tem-se:

$$K_p = 0,2 \quad K_{cr} = 0,18 \quad (25)$$

$$K_i = 0,4 \frac{K_{cr}}{P_{cr}} = 0,4 \frac{0,9}{1,832} = 0,196 \quad (26)$$

$$K_d = 0,066 \frac{K_{cr}}{P_{cr}} = 0,066 \frac{0,9}{1,832} = 0,032 \quad (27)$$

Implementando-se os valores obtidos anteriormente e observou-se a resposta a um escalão de tensão de 5V:

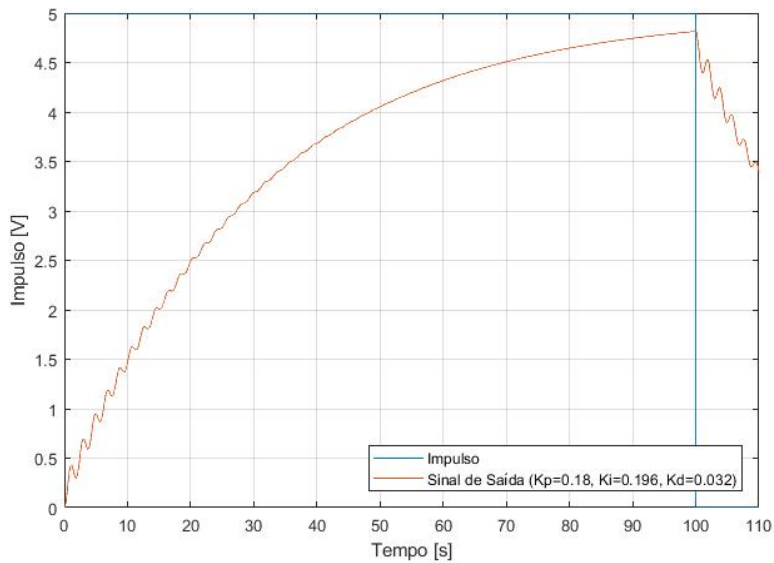


Figura 48 - Sinal de saída com controlador PID, sem sobrelevação ($K_p=0,18$, $K_i=0,196$, $K_d=0,032$).

Tanto para as curvas de pequena sobrelevação como para sem sobrelevação quando se utiliza a função *tune* do *Matlab* deu o mesmo valor para os parâmetros do controlador K_p , K_i , K_d obtidos para o método de Ziegler-Nichols original, Figura 46.

4.2 Implementação Laboratorial

Após atualizar a bancada de testes para incorporar o sistema Massa-Mola-Atrito, o próximo passo foi implementar o controlador PID no Arduino. O potenciômetro foi utilizado como referência de entrada, enquanto os dados do sensor MPU6050 foram utilizados como retroalimentação da posição do servo. O código foi escrito para implementar o algoritmo do controlador PID.

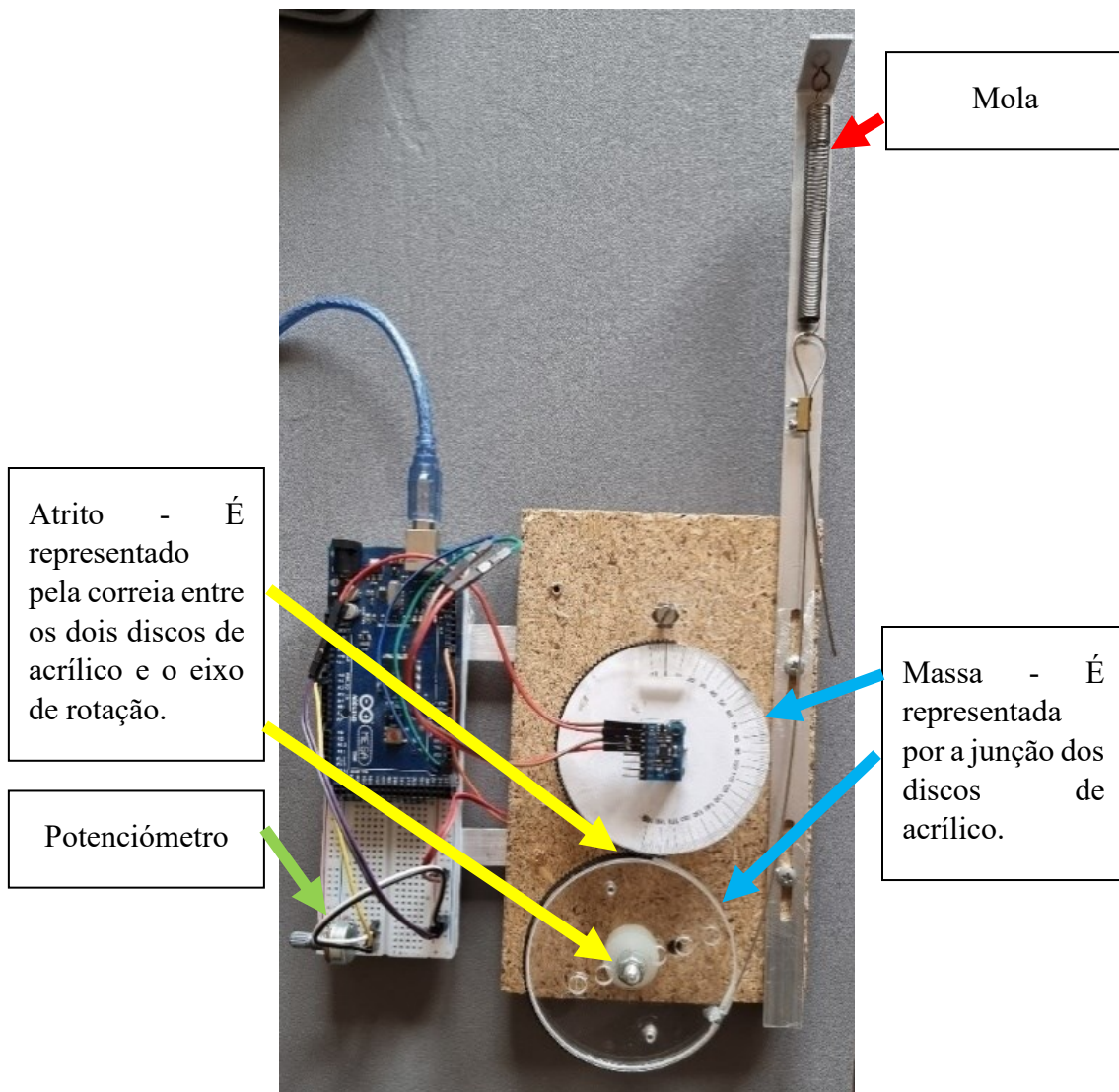


Figura 49 - Bancada de testes com implementação do sistema Massa-Mola-Atrito.

Para simular o sistema mecânico (Massa-Mola-Atrito) adicionou-se à bancada de testes, uma mola (Figura 50), uma roda de acrílico (Figura 51) e uma correia para servir de acoplador com a roda que está ligada ao servo motor e sensor, na mesma roda abriu-se

um sulco onde se encaixa o cabo de aço quando há o movimento para exercer força na mola.

Assim com os componentes representados nas Figura 50 e Figura 51, conseguimos um sistema mecânico Massa-Mola-Atrito, com a mola temos a componente K , com a adição do disco de acrílico e todos os componentes presentes deste sistema temos a massa M e o atrito A é representado pelo atrito causado na rotação da roda de acrílico com o seu eixo.



Figura 50 – Fotografia da mola utilizada.

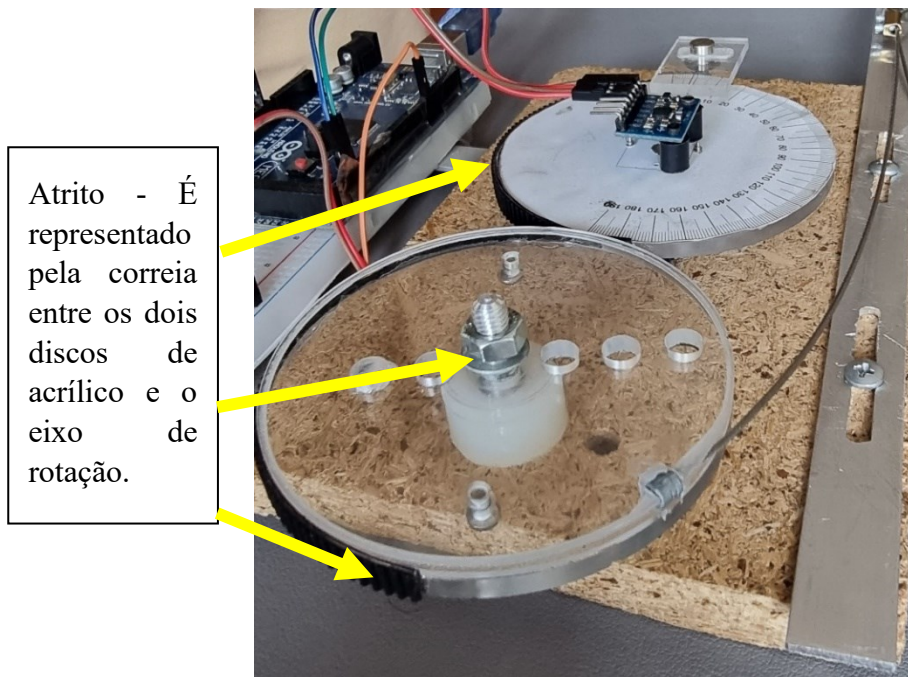


Figura 51 - Roda dentada em acrílico com cabo de aço.

Com o sistema mecânico implementado na bancada de testes, adicionou-se ainda um potenciômetro para mais facilmente poder-se alterar a referência do sistema sem ser necessário um re-upload para o Arduino.

4.3 Controlador PID

Com a bancada de testes já atualizada para conseguir simular e obter as características do sistema Massa-Mola-Atrito, procedeu-se agora a implementação no controlador do tipo PID em código no Arduino, utilizando assim o potenciômetro como referência de entrada e utilizando os dados fornecidos pelo sensor MPU6050 como retroação da posição do servo.

$$\text{Proporcional} = \text{erro} K_p \quad (28)$$

$$\text{Integral} = K_i \int_0^t \text{erro}(t) dt \quad (29)$$

$$\text{Derivativo} = K_d \frac{d\text{erro}(t)}{dt} \quad (30)$$

Ajustando as relações da Tabela 1, de maneira a possibilitar o uso na linguagem de programação, calculou-se as relações para obter-se o ganho integral K_i , e o ganho derivativo K_d .

Tabela 3 - Relações de Ziegler-Nichols.

Tipo de Controlador	K_p	K_i	K_d
P	$0,5 K_{cr}$	∞	0
PI	$0,45 K_{cr}$	$0,54 (K_{cr}/P_{cr})$	0
PID	$0,6 K_{cr}$	$1,2 (K_{cr}/P_{cr})$	$0,075 (K_{cr}/P_{cr})$

Fazendo uma conversão para linguagem de código no Arduino:

Para o cálculo do erro:

$$\text{erro} = \text{Setpoint} - \text{absangulo} \quad (31)$$

Em que temos o *erro*, a *Referência* definida pelo valor do potenciômetro, sendo o termo *absangulo* o valor dado pelo sensor MPU6050.

$$integral += erro dt \quad (32)$$

O valor integral vai guardar a multiplicação do valor do erro já calculado com o *dt* do tempo entre o valor anterior do erro com o presente.

$$derivativo = (erro - pre_{erro})/dt \quad (33)$$

O valor derivativo vai guardar a diferença dos erros associados à última iteração e à iteração presente no tempo *dt*.

$$controle = K_p erro + K_i integral + K_d derivativo \quad (34)$$

Com esta equação obtemos a variável que irá fazer o controle do servo motor.

4.4 Resultados Experimentais

Após a implementação do controlador no código (anexo D), prosseguiu-se para o estudo do controlador. Com base no que foi implementado anteriormente quando se dimensionou o controlador PID para a simulação, decidiu-se usar o método Ziegler-Nichols, tendo este método como um dos seus pontos fortes a facilidade de aplicabilidade sem ser preciso um grande conhecimento da dinâmica dos vários processos. Sendo que não se conhece os parâmetros do processo do sistema Massa-Mola-Atrito, utilizou-se como ponto de partida o ganho $K_p=1$ e as restantes constantes K_i e K_d igual a zero, aumentando lentamente o ganho do controlador K_p , de maneira a que o sistema chegue ao seu limite crítico.

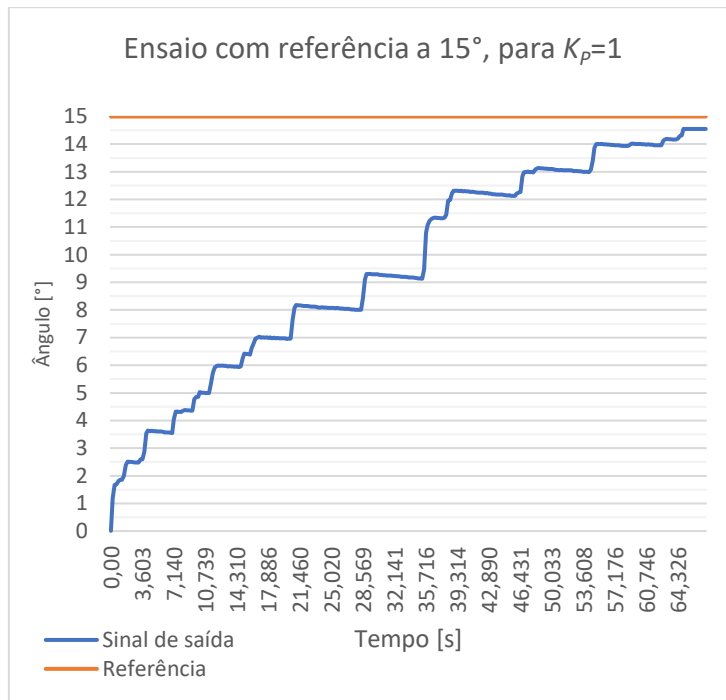


Figura 52 – Observação do sinal de saída, do ensaio experimental com referência a 15°, para $K_p=1$.

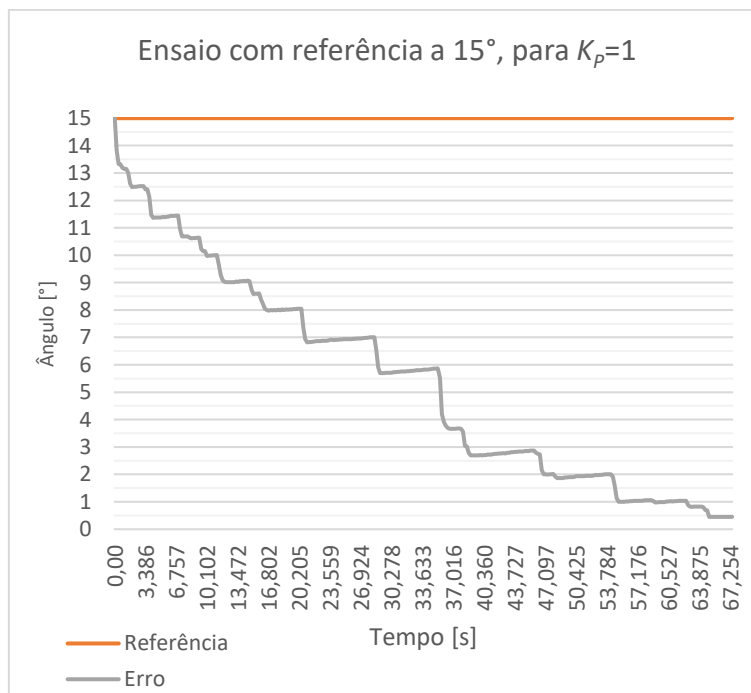


Figura 53 – Observação do erro, do ensaio experimental com referência a 15°, para $K_p=1$.

Durante o ensaio experimental do servo motor para $K_p=1$, com a referência definida em 15° e partindo do repouso, pode-se observar a aproximação do sinal de saída em relação à referência na Figura 52 e a diminuição do erro Figura 53. Foi possível perceber que o

o sistema estabilizou em torno de 67 segundos, com um erro de 0,45°. Ao observar estes resultados indicam que o controlador tem uma dinâmica lenta, tendo um erro finito, próximo de zero.

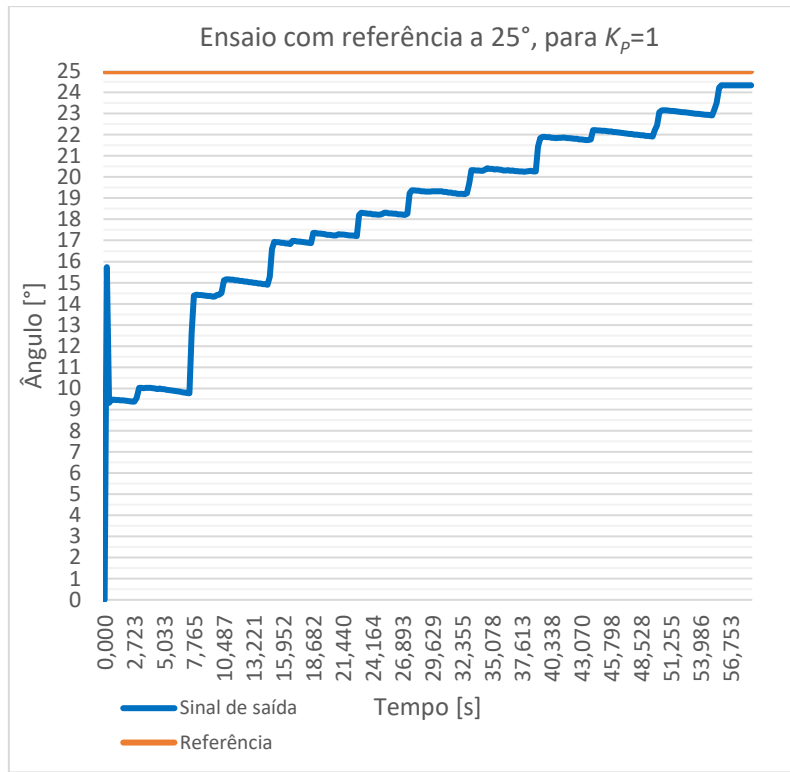


Figura 54 - Observação do sinal de saída, do ensaio experimental com referência a 25°, para $K_p=1$.

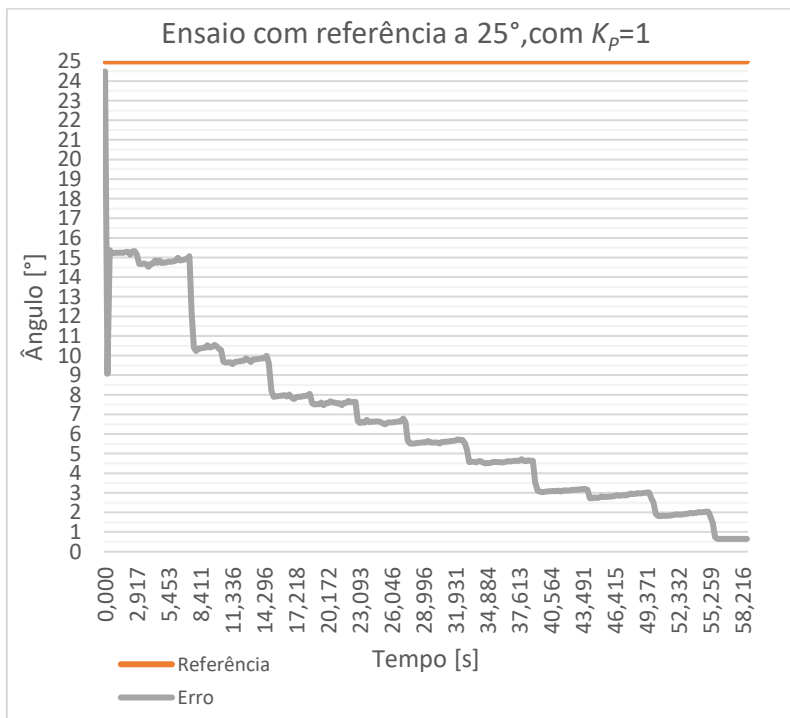


Figura 55 - Observação do erro, do ensaio experimental com referência a 25°, para $K_p=1$.

Durante o ensaio experimental do servo motor para $K_p=1$, com a referência definida em 25° e partindo do repouso, pode-se observar no início do ensaio um salto, provocado pela mudança do estado de repouso para o funcionamento, na Figura 54 observa-se a aproximação do sinal de saída em relação à referência e na Figura 55 a diminuição do erro. Foi possível perceber que o sistema estabilizou em torno de 58 segundos, sendo ligeiramente mais rápido que o ensaio anterior, com um erro de $0,67^\circ$ em relação à referência.

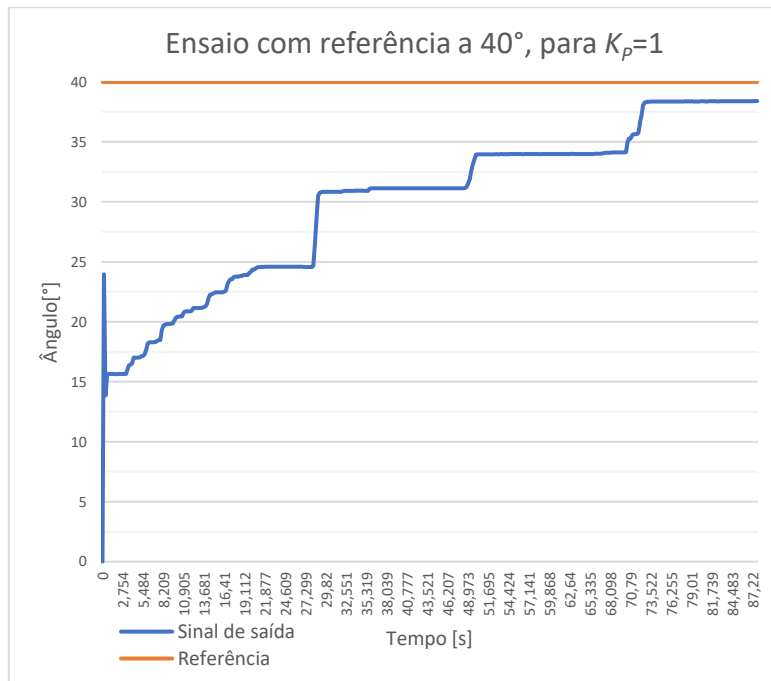


Figura 56 - Observação do sinal de saída, do ensaio experimental com referência a 40° , para $K_p=1$.

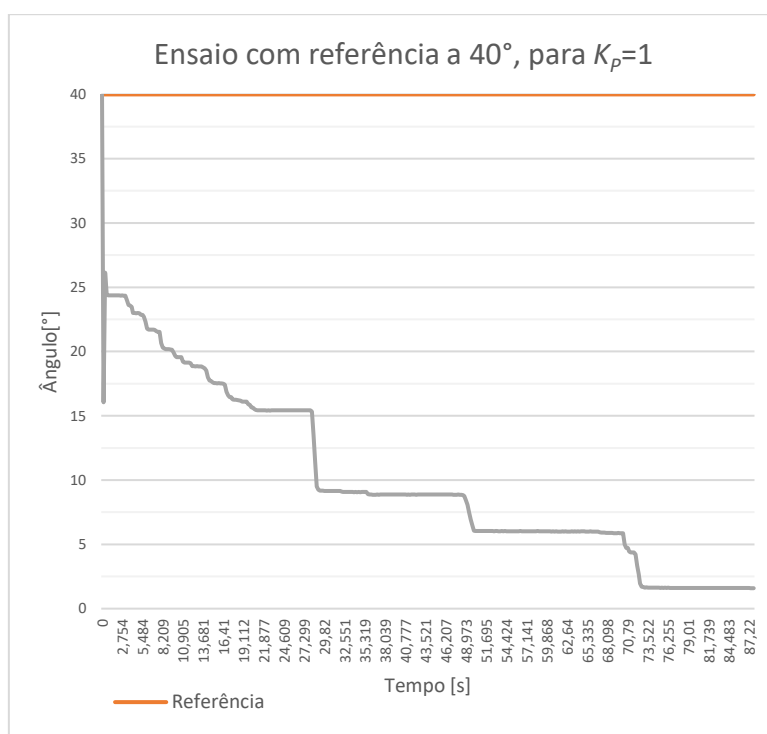


Figura 57 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1$.

Durante o ensaio experimental do servo motor com o ganho proporcional de $K_p=1$, desta vez com a referência definida em 40° e partindo novamente do repouso, pode-se observar que no início do ensaio, o motor tem o mesmo comportamento do ensaio com a referência a 25°, tendo sido provocado a mesma pela mudança do estado de repouso para o funcionamento, na Figura 56 observa-se a aproximação do sinal de saída em relação à referência e na Figura 57 a diminuição do erro em relação à referência. Foi possível perceber que demorou mais tempo que os ensaios anteriores, estabilizando em torno de 87 segundos, sendo mais lento e com graus de ajuste mais demorados terminando com o mesmo com um erro de 1,59°.

Tabela 4 - Resumo de resultados experimentais para $K_p=1$

Referência [Graus °]	Sinal de saída [Graus °]	Erro [Graus °]	Erro [%]
15	14,55	0,45	3,09
25	24,33	0,67	2,75
40	38,41	1,59	4,14

Depois de vários ensaios e terem ocorrido alguns danos em diversos motores, as referências foram escolhidas para os ensaios experimentais realizados, com algum cuidado, para evitar a perda de mais equipamentos. Ao observar o comportamento do sistema para diferentes referências, notou-se que a dinâmica do sistema muda consoante o aumento da amplitude do ângulo da referência. Isso pode ser explicado pelo aumento do esforço exercido pelo motor, que resulta em uma resposta mais lenta.

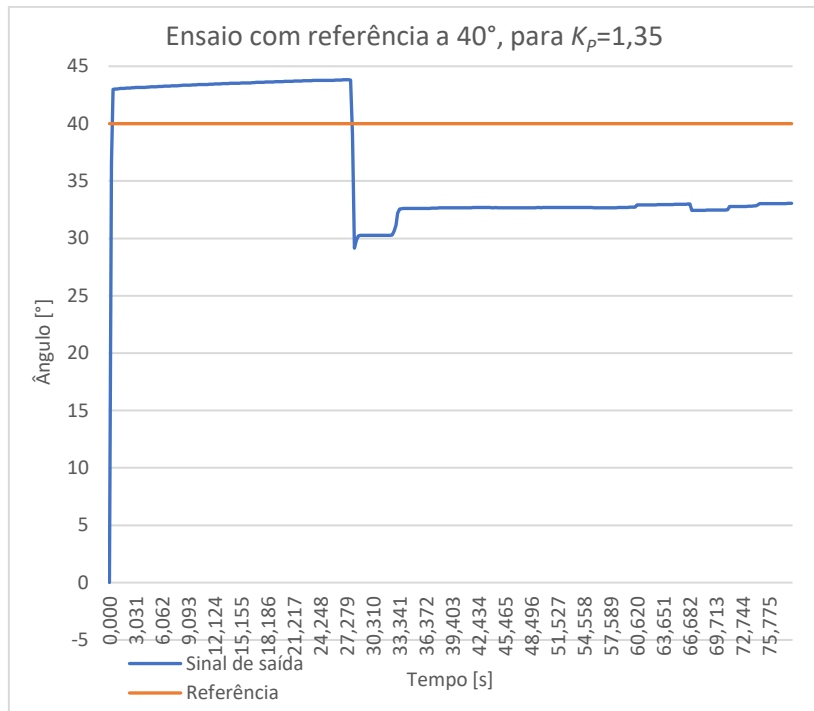


Figura 58 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=1,35$.

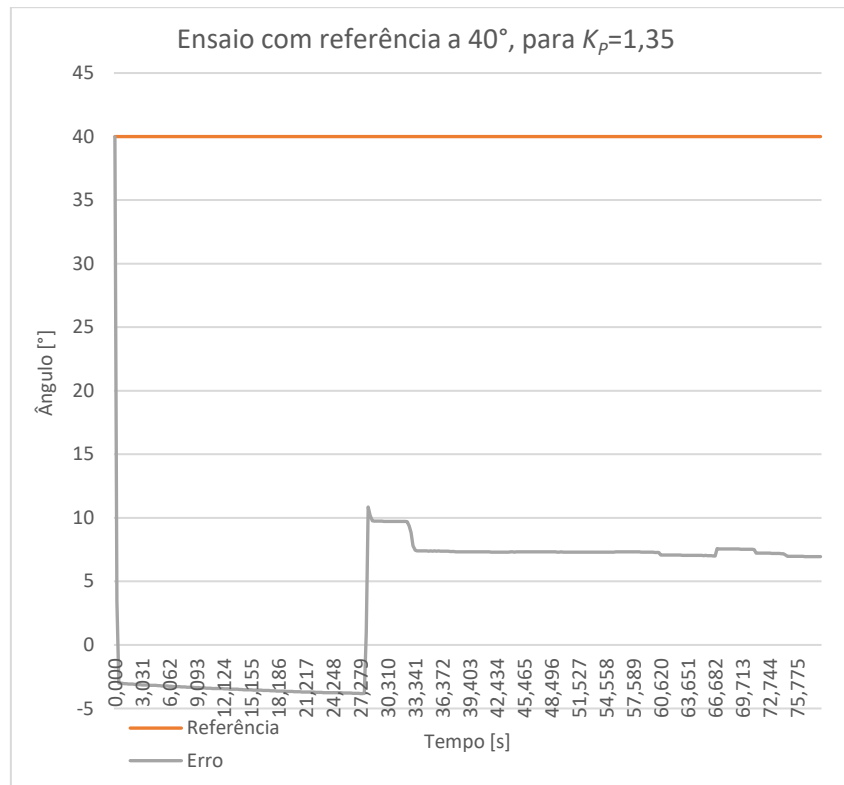


Figura 59 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1,35$.

Durante o ensaio experimental do servo motor para $K_p=1,35$, com a referência definida a 40° e partindo do repouso, existe um salto do sinal da saída passando acima da referência, podendo significar uma má leitura do sensor. Alguns segundos após o começo do ensaio experimental o sinal de saída diminui para se colocar abaixo da referência, tendo o controlador uma resposta lenta e demorada, estabilizando por volta dos 75 segundos, com um erro 6,95°.

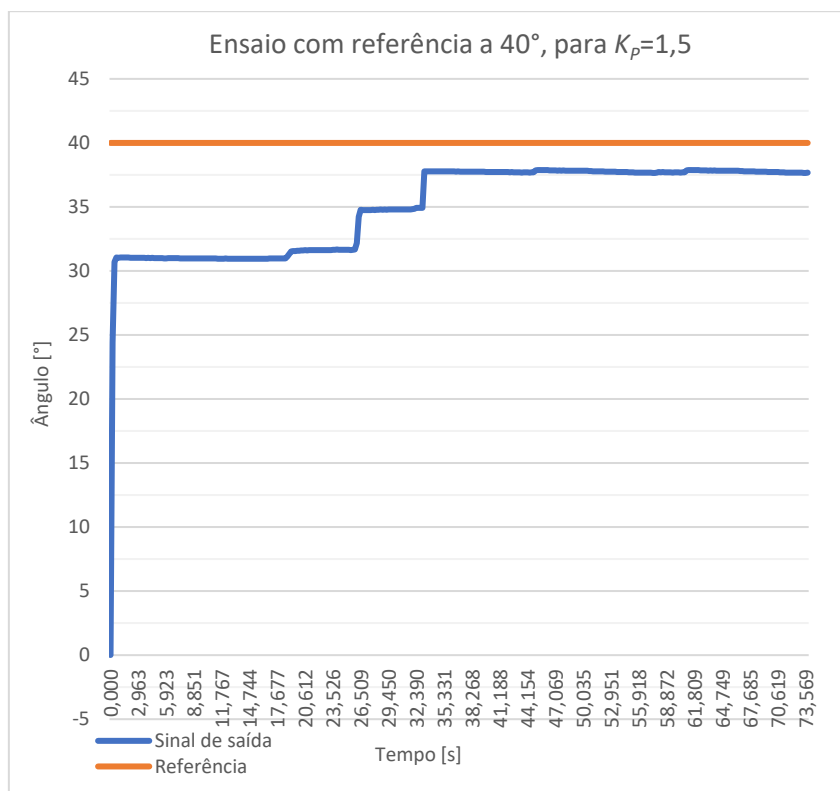


Figura 60 - Observação do sinal de saída, do ensaio experimental com referência a 40°, para $K_p=1,5$.

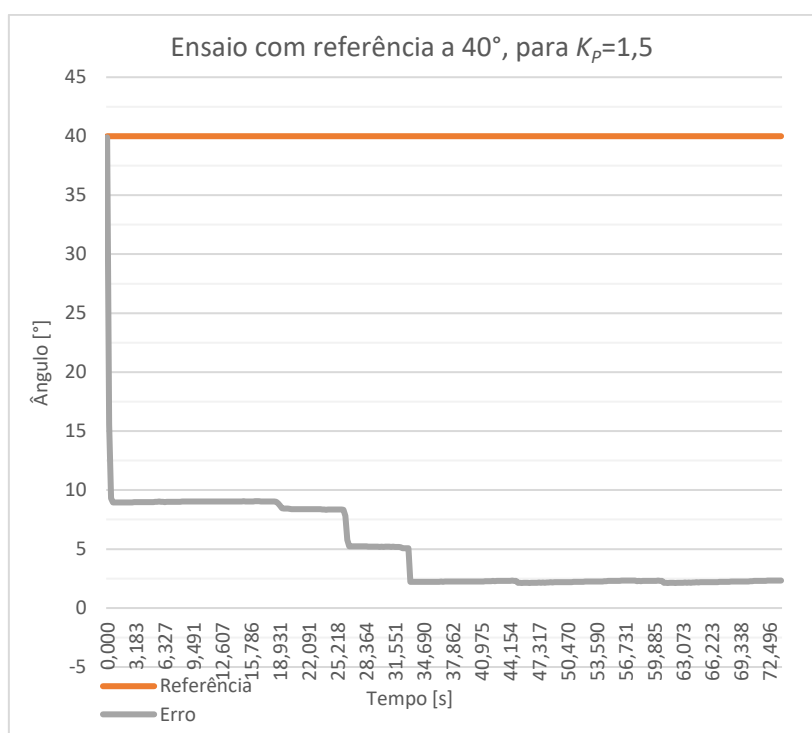


Figura 61 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=1,5$.

Durante o ensaio experimental do servo motor com o ganho proporcional de $K_p=1,5$, com a referência definida em 40° e partindo do repouso, pode-se observar a aproximação do sinal de saída em relação à referência (Figura 60), e conseqüentemente a diminuição do

erro (Figura 61). Foi possível perceber que o sistema se estabilizou em torno de 73 segundos, com um erro de $2,33^\circ$

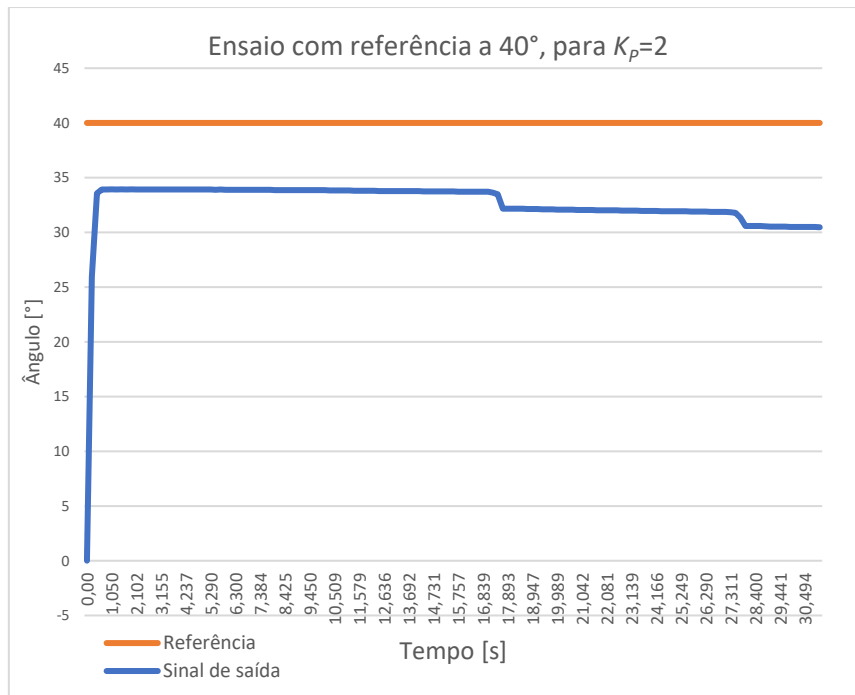


Figura 62 - Observação do sinal de saída, do ensaio experimental com referência a 40° , para $K_p=2$.

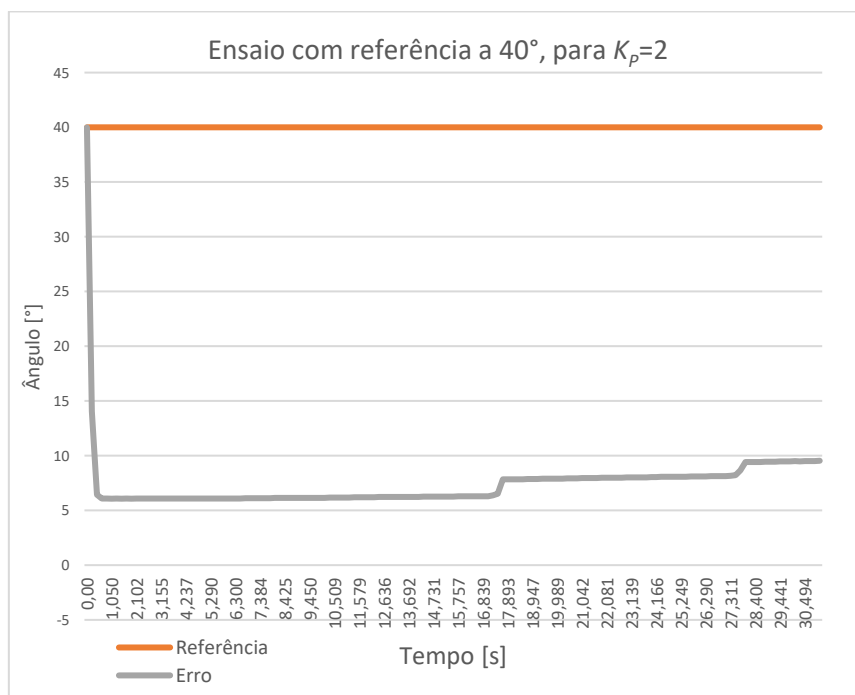


Figura 63 - Observação do erro, do ensaio experimental com referência a 40° , para $K_p=2$.

Durante o ensaio experimental com os resultados nas demonstrados nas Figura 62 e Figura 63, do servo motor para o ganho proporcional de $K_p=2$, com a referência definida a 40° e partindo do repouso, pode-se observar um afastamento do sinal de saída em relação à referência. Podendo ser devido ao *drift* por parte do sensor, passando uma leitura incorreta da posição do servo ao controlador. Foi possível perceber que o sistema estabilizou-se em torno de 30 segundos, com um erro de $9,53^\circ$.

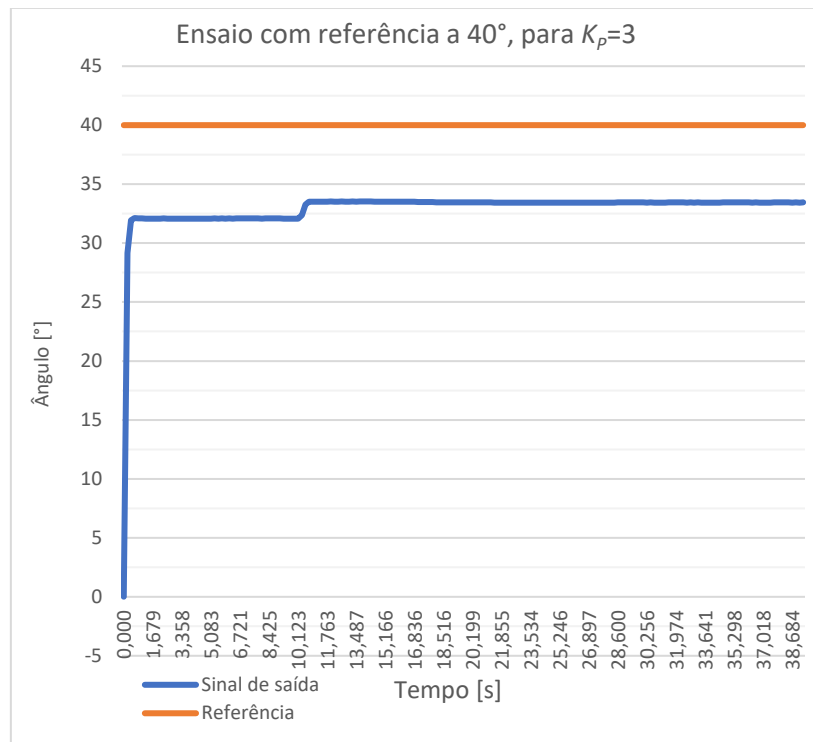


Figura 64 - Observação do sinal de saída, do ensaio experimental com referência a 40° , para $K_p=3$.

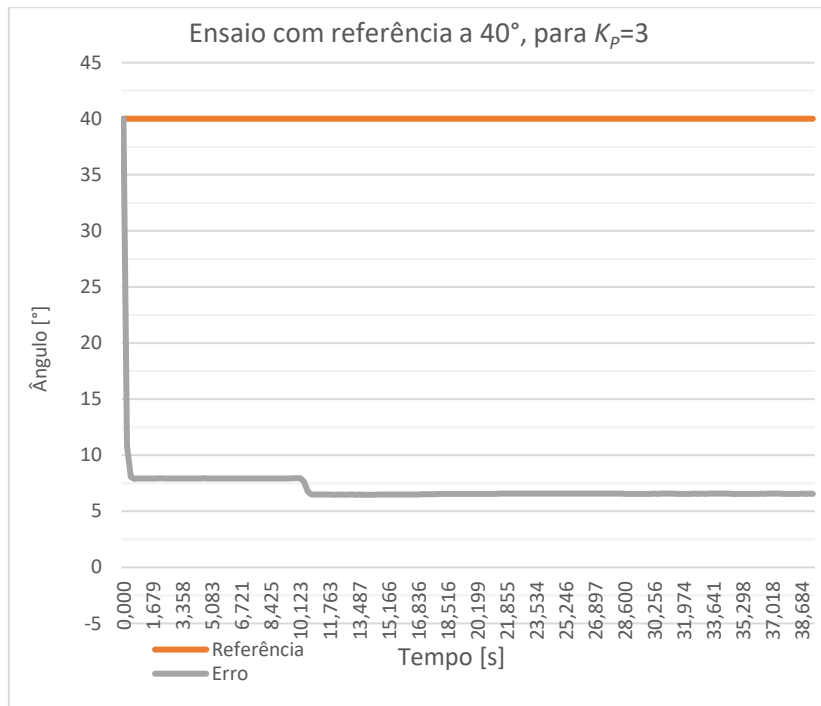


Figura 65 - Observação do erro, do ensaio experimental com referência a 40°, para $K_p=3$.

Durante o ensaio experimental do servo motor para o ganho proporcional de $K_p=3$, com a referência definida a 40° e partindo do repouso, pode-se observar a aproximação do sinal de saída em relação à referência e conseqüentemente a diminuição do erro. Foi possível perceber que o sistema estabilizou em torno de 38 segundos, com um erro de 6,56°

Tabela 5 - Resumo de resultados para referência a 40°

K_p	Referência [Graus °]	Erro [Graus °]
1	38,41	1,59
1,35	33,05	6,95
1,5	37,67	2,33
2	30,49	9,53
3	33,44	6,56

Observa-se também na Tabela 5, uma não linearização dos dados obtidos em relação à estabilização da curva e ao erro associado a cada ensaio experimental. Os resultados

esperados seriam, quanto maior fosse o ganho de K_P , menor seria o erro em relação ao ponto de referência.

5 Conclusões

5.1 Conclusões Gerais

Nesta dissertação dos vários objetivos propostos é dimensionar um sistema eletromecânico com um auxílio de uma bancada de testes e criar um modelo do sistema, analisando-se a característica dos componentes usados. No sistema em estudo, presente na dissertação não foram introduzidas perturbações na modelação, pois o mesmo tem como objetivo criar um modelo de base para estudo e demonstrar uma prova de conceito. Com o isto as componentes elétrica e mecânica do modelo no *software Matlab – Simulink*, foram validadas, com os dados extraídos dos ensaios experimentais realizados aos componentes.

A criação do modelo e a simulação do sistema em *Matlab - Simulink* em cadeia fechada usando o método de Ziegler-Nichols para obtenção das constantes K_p , K_i e K_d do controlador PID, parece ser eficaz para os diversos tipos de modelo e diferentes sobrelevações sugeridas no método de Ziegler-Nichols. Foi observado que é possível fazer o controlo através de um controlador PID com os parâmetros obtidos.

A criação de um protótipo que permita a simulação uma oscilação mecânica através da aplicação de um sistema Massa-Mola-Atrito foi bem-sucedida, possibilitando a implementação e o dimensionamento de um controlador em cadeia fechada.

Com base nos resultados experimentais obtidos com o protótipo, foi decidido dimensionar o controlador utilizando novamente o método de Ziegler-Nichols. Como não é conhecida a dinâmica e os parâmetros do processo a ser controlado, que tem como uma das suas vantagens não exigir o conhecimento por completo do processo para determinação dos parâmetros do controlador. Com base nisto e observando os resultados, observa-se que o controlo feito para 15°, tem um erro de cerca de 3,09%, para a referência de 25° observa-se um erro de 2,75° e para os 40° observa-se 4,14% de erro em relação á referência, o que dentro do intervalo testado dá cerca de 3,32% de média para a margem de erro do controlado e leitura.

No entanto, os resultados obtidos seriam passíveis de algumas melhorias. Podendo existir melhoramentos ao nível de eliminação do ruído utilizando como exemplo cabos sem *jumpers*, passar o circuito de controlo para uma placa PCB, eliminando assim desta forma

algumas correntes parasitas, que possam ter existido e uma melhoria à robustez da parte mecânica. Dessa forma, conclui-se também, que a obtenção do modelo do processo é fundamental para o sucesso da implementação do controlador e para o desempenho adequado do sistema.

5.2 Desenvolvimentos Futuros

Neste subcapítulo, descrevem-se alguns tópicos que poder vir a ser úteis para outros trabalhos a ser desenvolvidos tendo como base esta dissertação:

- A obtenção dos parâmetros através de ensaios laboratoriais para o sistema Massa-Mola-Atrito;
- O melhoramento dos modelos obtidos para o servo motor e sensor;
- Implementar modelo com os valores obtidos dos ensaios laboratoriais e fazer a validação do sistema mecânico Massa-Mola-Atrito;
- Fazer o estudo para os restantes eixos (X e Y);
- Desenvolver uma representação gráfica em Processing para fácil e interativa identificação do funcionamento do controlo;
- Desenvolver um protótipo que permita implementar a solução em pessoas com a doença de Parkinson.

Bibliografia

- [1] V. CABREIRA e J. MASSANO, “Doença de Parkinson: Revisão Clínica e Atualização,” *Revista Científica da Ordem dos Médicos*, pp. 661-670, 2019. (Artigo de opinião)
- [2] J. Needham, *Science and Civilization in China: Volume 4, Physics and Physical Technology; Part 2, Mechanical Engineering*, Taipei: Caves Books Ltd. Pág. 229, 1986.(Livro) ISBN-10: 9780521070607
- [3] J. Needham, *Science and Civilization in China: Volume 4, Physics and Physical Technology; Part 2, Mechanical Engineering*, Taipei: Caves Books Ltd. Pág. 233-234, 1986. ISBN-10: 9780521070607
- [4] J. Needham, *Science and Civilization in China: Volume 4, Physics and Physical Technology; Part 2, Mechanical Engineering*, Taipei: Caves Books Ltd. p.234, 1986. ISBN-10: 9780521070607
- [5] G. Sarton, *A History of Science: Hellenistic Science and Culture in the Last Three centuries B.C.*, Cambridge: Harvard University Press. pp. 349–350, 1959.(Livro) ISBN: 10. 9780674400528
- [6] S. Handler, *Austere Luminosity of Chinese Classical Furniture*, California: University of California Press p. 308, 2001. (Livro) ISBN-10: 0520214846
- [7] A. Wilson, “Machines, Power and the Ancient Economy,” *The Journal of Roman Studies*, pp. 1-32, 2002. (Artigo de revista)
- [8] S. L. Macey, *Encyclopedia of Time*, Taylor & Francis, 1994.(Livro) ISBN-10. 1412941644
- [9] T. E. o. E. Britannica, “Encyclopedia Britannica,” 16 June 2017. [Online]. Available: <https://www.britannica.com/technology/inertial-guidance-system>. [Acedido em 2 Fevereiro 2022].
- [10] T. E. o. E. Britannica, “Encyclopedia Britannica,” 5 November 2021. [Online]. Available: <https://www.britannica.com/technology/gyroscope>. [Acedido em 3 Fevereiro 2022].
- [11] L. Vieira, “3D gyroscope,” 4 Outubro 2006. [Online]. Available: https://commons.wikimedia.org/wiki/File:3D_Gyroscope.png. [Acedido em 10 Dezembro 2023].

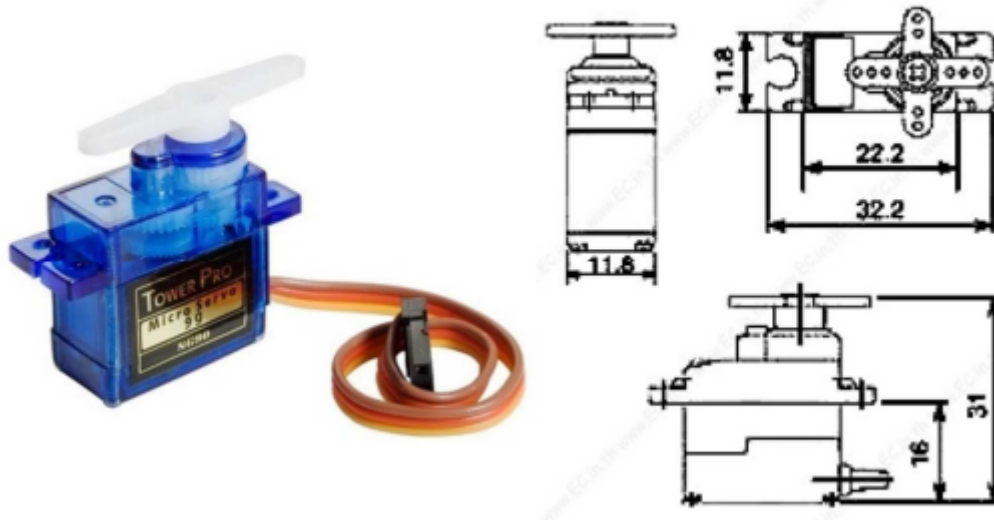
- [12] D. L. Polla, “MEMS Technology for Biomedical Applications,” Minneapolis, Minnesota, 2001. (Artigo) DOI: 10.1109/ICSICT.2001.981417
- [13] F. T. M. C. a. S. S. S. Murat Cenk C, avus,og~lu, “A Laparoscopic Telesurgical Workstation,” *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*, vol. 15, n° 4, 1999. (Artigo) DOI: 10.1109/70.782027
- [14] E. K. N. T. a. K. I. Kohji Masuda, “Three dimensional motion mechanism of ultrasound probe and its application for tele-echography system,” em *International Conference on IntelUgent Robots and Systems*, Maui, Hawaii, 2001. (Artigo de conferência) DOI: 10.1109/IROS.2001.976317
- [15] P. R. E. L. A. F. a. J. G. Wael Bachta, “Active Stabilization for Robotized Beating Heart Surgery,” *IEEE TRANSACTIONS ON ROBOTICS*, vol. 27, n° 4, pp. 757-768, 2011. (Artigo) DOI: 10.1109/TRO.2011.2137770
- [16] P. N. S. L. M. S. C. O. C. M. I. John A. Martinez, “Design of Wrist Gimbal: a Forearm and Wrist Exoskeleton for Stroke Rehabilitation,” em *IEEE International Conference on Rehabilitation Robotics*, Seattle, Washington USA, 2013. (Artigo de conferência) DOI: 10.1109/ICORR.2013.6650459
- [17] The Editors of Encyclopaedia. "governor". Encyclopedia Britannica, “Britannica,” 20 July 1998. [Online]. Available: <https://www.britannica.com/technology/governor-machine-component>. [Acedido em 20 June 2023].
- [18] J. &. N. N. B. Ziegler, Optimum settings for automatic controllers, ASME, 1942.
- [19] Electrical 4U, “Electrical 4U,” 25 June 2023. [Online]. Available: https://www.electrical4u.com/control-engineering-historical-review-and-types-of-control-engineering/?utm_content=cmp-true. [Acedido em 30 June 2023].
- [20] S. Hosseini, A. Al-Jumaily e S. A. Abboud, “Active Force Control System for Han Tremor Suppression by Different Actuators,” em 2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA), Ras Al Khaimah, United Arab Emirates, 06-08 December 2016. (Artigo) DOI: 10.1109/ICEDSA.2016.7818487
- [21] A. A. MahdiAbbasi, “Modeling and experimental study of a hand tremor suppression system,” *Elsevier*, n° 126, pp. 189-200, 2018. (Artigo) DOI: <https://doi.org/10.1016/j.mechmachtheory.2018.04.009>

- [22] Auawise, “Wiki Commons,” 10 February 2010. [Online]. Available: https://commons.wikimedia.org/wiki/File:Yaw_Axis_Corrected.svg. [Acedido em 20 February 2023].
- [23] M. A. Botto, *Controlo de Sistemas*, AEIST Press, 2008. (Livro) ISBN: 978-989-683-049-6
- [24] A. S. M. a. K. R. Godfrey, “Rule-Based Autotuning Based on Frequency Domain Identification,” *IEEE Transactions on Control Systems Technology*, vol. vol 6, 1998. (Artigo de revista) DOI: 10.1109/87.654876
- [25] G. Ellis, *Control System Design Guide*, Butterworth-Heinemann, 2012. (Livro) ISBN-10: 0128102411

Anexos

Anexo A – Características do Servo Motor

SG90 9 g Micro Servo



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but *smaller*. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

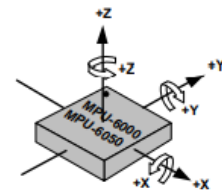
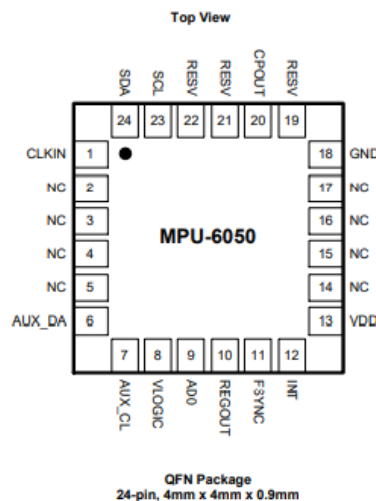
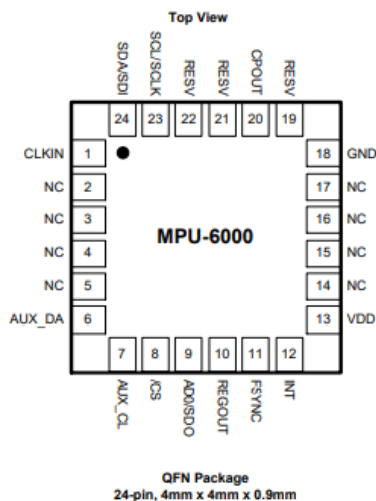
Specifications

- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10 μ s
- Temperature range: 0 $^{\circ}$ C – 55 $^{\circ}$ C

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is all the way to the left. ms pulse) is all the way to the right, ""-90" (~1ms pulse) is all the way to the left.

Anexo B – Características do Sensor MPU6050

Pin Number	MPU-6000	MPU-6050	Pin Name	Pin Description
1	Y	Y	CLKIN	Optional external reference clock input. Connect to GND if unused.
6	Y	Y	AUX_DA	I ² C master serial data, for connecting to external sensors
7	Y	Y	AUX_CL	I ² C Master serial clock, for connecting to external sensors
8	Y		/CS	SPI chip select (0=SPI mode)
8		Y	VLOGIC	Digital I/O supply voltage
9	Y		AD0 / SDO	I ² C Slave Address LSB (AD0); SPI serial data output (SDO)
9		Y	AD0	I ² C Slave Address LSB (AD0)
10	Y	Y	REGOUT	Regulator filter capacitor connection
11	Y	Y	FSYNC	Frame synchronization digital input. Connect to GND if unused.
12	Y	Y	INT	Interrupt digital output (totem pole or open-drain)
13	Y	Y	VDD	Power supply voltage and Digital I/O supply voltage
18	Y	Y	GND	Power supply ground
19, 21	Y	Y	RESV	Reserved. Do not connect.
20	Y	Y	CPOUT	Charge pump capacitor connection
22	Y	Y	RESV	Reserved. Do not connect.
23	Y		SCL / SCLK	I ² C serial clock (SCL); SPI serial clock (SCLK)
23		Y	SCL	I ² C serial clock (SCL)
24	Y		SDA / SDI	I ² C serial data (SDA); SPI serial data input (SDI)
24		Y	SDA	I ² C serial data (SDA)
2, 3, 4, 5, 14, 15, 16, 17	Y	Y	NC	Not internally connected. May be used for PCB trace routing.



Serial Interface

Pin Number	MPU-6000	MPU-6050	Pin Name	Pin Description
8	Y		/CS	SPI chip select (0=SPI enable)
8		Y	VLOGIC	Digital I/O supply voltage. VLOGIC must be \leq VDD at all times.
9	Y		AD0 / SDO	I ² C Slave Address LSB (AD0); SPI serial data output (SDO)
9		Y	AD0	I ² C Slave Address LSB
23	Y		SCL / SCLK	I ² C serial clock (SCL); SPI serial clock (SCLK)
23		Y	SCL	I ² C serial clock
24	Y		SDA / SDI	I ² C serial data (SDA); SPI serial data input (SDI)
24		Y	SDA	I ² C serial data

Anexo C – Características do Arduino



Description

Arduino® Mega 2560 is an exemplary development board dedicated for building extensive applications as compared to other maker boards by Arduino. The board accommodates the ATmega2560 microcontroller, which operates at a frequency of 16 MHz. The board contains 54 digital input/output pins, 16 analog inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header, and a reset button.

Features

- **ATmega2560 Processor**
 - Up to 16 MIPS Throughput at 16MHz
 - 256k bytes (of which 8k is used for the bootloader)
 - 4k bytes EEPROM
 - 8k bytes Internal SRAM
 - 32 × 8 General Purpose Working Registers
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Four Programmable Serial USART
 - Controller/Peripheral SPI Serial Interface
- **ATmega16U2**
 - Up to 16 MIPS Throughput at 16 MHz
 - 16k bytes ISP Flash Memory
 - 512 bytes EEPROM
 - 512 bytes SRAM
 - USART with SPI master only mode and hardware flow control (RTS/CTS)
 - Master/Slave SPI Serial Interface
- **Sleep Modes**
 - Idle
 - ADC Noise Reduction
 - Power-save
 - Power-down
 - Standby
 - Extended Standby
- **Power**
 - USB Connection
 - External AC/DC Adapter
- **I/O**
 - 54 Digital
 - 16 Analog
 - 15 PWM Output

Anexo D – Código para Implementação no Arduíno

```
/* =====  
  I2Cdev device library code is placed under the MIT (Massachusetts  
Institute of Technology) license  
  Copyright (c) 2012 Jeff Rowberg  
  
  Permission is hereby granted, free of charge, to any person obtaining a  
copy  
  of this software and associated documentation files (the "Software"),  
to deal  
  in the Software without restriction, including without limitation the  
rights  
  to use, copy, modify, merge, publish, distribute, sublicense, and/or  
sell  
  copies of the Software, and to permit persons to whom the Software is  
furnished to do so, subject to the following conditions:  
  
  The above copyright notice and this permission notice shall be included  
in  
  all copies or substantial portions of the Software.  
  
  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS  
OR  
  IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF  
MERCHANTABILITY,  
  FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL  
THE  
  AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER  
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING  
FROM,  
  OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS  
IN  
  THE SOFTWARE.  
  =====  
*/  
  
#include "I2Cdev.h" //Biblioteca protocolo de comunicação
```

```

#include "MPU6050_6Axis_MotionApps_V6_12.h" //Biblioteca para usar com o
grafismo e usar o Yaw, Pitch e o Roll
#include <Servo.h> //Biblioteca Servo motor
#if I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
#include "Wire.h"
#endif
#define pot_pin A1 // Define o Pin analógico da board do Arduino
MPU6050 mpu;
#define LER_YAWPITCHROLL // Define o output para o Yaw, Pitch e Roll
#define INTERRUPT_PIN 2
#define LED_PIN 13
bool blinkState = false; // O Led deixa de piscar se não trabalhar

// Variáveis de controlo e estado
bool dmpReady = false; // Se o DMP (Digital Motion Processor do MPU6050)
esta variável passa a (true/verdadeiro) se ele for iniciado com sucesso
uint8_t mpuIntStatus; // Bit de interrupção do MPU
uint8_t devStatus; // Faz return do estado depois de cada operação
do dispositivo (0 = sucesso, !0 = erro)
uint16_t packetSize; // Armazena o pacote do DMP (default é 42 bytes),
caso contrário não está a fazer o envio todo
uint16_t fifoCount; // Contagem dos bytes em FIFO
uint8_t fifoBuffer[64]; // FIFO buffer

// Orientação/movimento
Quaternion q; // [w, x, y, z] contentor de quaternions
VectorInt16 aa; // [x, y, z] Variáveis de aceleração
VectorInt16 gy; // [x, y, z] Variáveis de giroscópio
VectorInt16 aaReal; // [x, y, z] Variáveis de aceleração
sem a gravidade
VectorInt16 aaWorld; // [x, y, z] Variáveis de aceleração
em mundo real
VectorFloat gravity; // [x, y, z] Vetor da gravidade
float euler[3]; // [psi, theta, phi] Ângulos de Euler
float ypr[3]; // [yaw, pitch, roll] Ângulos yaw/pitch/roll
com vetor de gravidade

Servo myservo; // Iniciação do objeto servo

```

```

int valor, ax, ay, az;          // Variáveis para ler o valor do
potenciometro, acc X, acc Y e acc Z
double erro;                  // Erro entre o Referência e a posição atual
double Kp = 0;                // Ganho proporcional PID (alterar de acordo com o
pretendido)
double Ki = 0;                // Ganho integral PID (alterar de acordo com o
pretendido)
double Kd = 0;                // Ganho derivativo PID (alterar de acordo com o
pretendido)
double pre_erro;              // Erro anterior à iteração
double integral;              // Integral do erro para a parte do ganho integral
double Referência,angulo,dt;
double accX, accY, accZ,x,y,z;
double gyroX, gyroY, gyroZ;
unsigned long previousTime;
double controlo;

// Pacote para utilizar o grafismo mais tarde
uint8_t teapotPacket[14] = { '$', 0x02, 0, 0, 0, 0, 0, 0, 0, 0, 0x00,
0x00, '\r', '\n' };

// Rotina de interrupção
volatile bool mpuInterrupt = false;    // Se o pin de interrupção for a
high é sinalizado
void dmpDataReady() {
    mpuInterrupt = true;
}

// Definição de configurações (Setup)

void setup() {
#ifdef I2CDEV_IMPLEMENTATION == I2CDEV_ARDUINO_WIRE
    Wire.begin(); //Começo de comunicação
    Wire.setClock(400000); // 400kHz I2C clock (Este clock é o necessário
para 400kHz e para a visualização gráfica mais tarde)
#elif I2CDEV_IMPLEMENTATION == I2CDEV_BUILTIN_FASTWIRE
    Fastwire::setup(400, true);
#endif
}

```

```

// Inicio do da comunicação série, o baud rate é o 115200 porque o
Processing necessita dessa atualização
Serial.begin(115200);
while (!Serial);

// Inicio do dispositivo
Serial.println(F("Inicio do dispositivo I2C..."));
mpu.initialize();
pinMode(INTERRUPT_PIN, INPUT);

// Verificação da conexão
Serial.println(F("Teste de conexão..."));
Serial.println(mpu.testConnection() ? F("MPU6050 conexão feita") :
F("MPU6050 conexão falhou"));

// Esperar pelo input do utilizador !!!!!!!!!!!!!!!ATENÇÃO!!!!!!!!!!!!!! O
POTENCIOMETRO TEM QUE ESTAR NO ZERO PARA COMEÇAR
Serial.println("\n!!!!!!!!!!!!!!ATENÇÃO!!!!!!!!!!!!!! O POTENCIOMETRO TEM
QUE ESTAR NO ZERO PARA COMEÇAR");
Serial.println(F("\nEnviar qualquer caracter para começar o programa:
"));
while (Serial.available() && Serial.read()); // Limpa o buffer
while (!Serial.available()); // Espera pelos pacotes
while (Serial.available() && Serial.read()); // Limpa o buffer
novamente

// Carrega e configura o DMP
Serial.println(F("Iniciando o DMP..."));
devStatus = mpu.dmpInitialize();

// Os offsets abaixo foram retirados com outro programa para ajuste da
sensibilidade do MPU
mpu.setXGyroOffset(94);
mpu.setYGyroOffset(49);
mpu.setZGyroOffset(-19);
mpu.setXAccelOffset(231);
mpu.setYAccelOffset(75);

```

```

mpu.setZAccelOffset(1481);

// Verificar se funcionou (returns 0 se sim)
if (devStatus == 0) {
    // Calibração para zero do MPU6050
    mpu.CalibrateAccel(6);
    mpu.CalibrateGyro(6);
    Serial.println();
    mpu.PrintActiveOffsets();
    // Ativar o DMP
    Serial.println(F("Ativar o DMP..."));
    mpu.setDMPEnabled(true);

    // Ativar o interrupt
    Serial.print(F("Ativar o Interrupt detection (Arduino external
interrupt "));
    Serial.print(digitalPinToInterrupt(INTERRUPT_PIN));
    Serial.println(F(")..."));
    attachInterrupt(digitalPinToInterrupt(INTERRUPT_PIN), dmpDataReady,
RISING);
    mpuIntStatus = mpu.getIntStatus();

    // Meter o DMP Ready flag para que o ciclo de loop() saiba que está
tudo ok
    Serial.println(F("DMP pronto! A esperar pelo primeiro
interrupt..."));
    dmpReady = true;

    // comparação de pacotes esperados do DMP (42 bytes)
    packetSize = mpu.dmpGetFIFOPacketSize();
} else {
    // ERRO!
    // 1 = Inicio do load da memoria falhou
    // 2 = A configuração do DMP falhou

    Serial.print(F("DMP Iniciação falhou (code "));
    Serial.print(devStatus);
    Serial.println(F(")"));
}

```

```

}
myservo.attach(9); // Pin do servo motor
myservo.write(0); //Mete o servo a 0 graus
// Configura o LED da board para piscar em atividade
pinMode(LED_PIN, OUTPUT);
}

void loop() {
// Se o programa falhou não faz nada
if (!dmpReady) return;
// Lê os pacotes do FIFO
if (mpu.dmpGetCurrentFIFOPacket(fifoBuffer)) { // Obtém o ultimo pacote
do buffer

#ifdef LER_YAWPITCHROLL
mpu.dmpGetQuaternion(&q, fifoBuffer);
mpu.dmpGetGravity(&gravity, &q);
mpu.dmpGetYawPitchRoll(ypr, &q, &gravity);

unsigned long currentTime = millis();
dt = (currentTime - previousTime)/ 1000.0; // Se for 1000 passa para
s
previousTime = currentTime;
Serial.print(dt);Serial.print(" -dt \t\t\t");
angulo=ypr[0] * 180 / M_PI;
double absangulo=abs(angulo);

//Referência=0;
//delay(500);
//Referência = map(analogRead(pot_pin), 0, 1023, 0, 180); //
Uncomment para ler o valor do potenciometro e converte em ângulo
//Kp = map(analogRead(pot_pin), 0, 1023, 0, 800)/100.00; //
Uncomment para ler o valor do potenciometro fazer uma alteração do Kp
Serial.print(Kp,3);Serial.print(" -Kp \t\t\t");
Referência=32*0.95;
Serial.print(absangulo);Serial.print(" -Graus Sensor \t\t\t");
Serial.print(Referência);Serial.print(" -Graus Potenciometro
\t\t\t");

```

```

    erro = Referência - absangulo; // Faz a diferença entre o ângulo do
Referência (potenciometro) e o ângulo dado no MPU6050 no eixo Z
    double abserro=abs(erro);
    Serial.print(abserro);Serial.print(" -Graus erro \t\t\t");
    //Serial.print(angulo);Serial.print(" Graus Sensor \t\t\t");
    integral += erro*dt; // Atualiza o erro no integral
    Serial.print(integral);Serial.print(" -integral \t\t\t");
    double derivativo = (abserro - pre_erro)/dt; // Calcula a derivativa
no erro
    // O sinal de controlo é dado pela soma das componentes do PID
(Kp,Ki,Kd)
    controlo = Kp * abserro + Ki * integral; // + Kd * derivativo;
if (controlo >= 180)
{

    Serial.print("Limite máximo atingido\t\t\t");
    //return controlo=180;
}
if (controlo <=0){

    Serial.print("Limite mínimo atingido\t\t\t");
    //return controlo=0;
}

    Serial.print(controlo);Serial.print("-Graus PID \t\t\t ");
    // Atualiza a posição do servo com o sinal de controlo
myservo.write(controlo);
    pre_erro = erro; // Substitui o erro anterior para se usar no
próximo cálculo
    delay(200); // Delay para a próxima iteração

    Serial.println();

#endif

#ifdef OUTPUT_TEAPOT

    teapotPacket[2] = fifoBuffer[0];
    teapotPacket[3] = fifoBuffer[1];

```

```
teapotPacket[4] = fifoBuffer[4];
teapotPacket[5] = fifoBuffer[5];
teapotPacket[6] = fifoBuffer[8];
teapotPacket[7] = fifoBuffer[9];
teapotPacket[8] = fifoBuffer[12];
teapotPacket[9] = fifoBuffer[13];
Serial.write(teapotPacket, 14);
teapotPacket[11]++;
#endif

    blinkState = !blinkState;
    digitalWrite(LED_PIN, blinkState);
}
}
```