

UNIVERSIDADE DE SÃO PAULO  
ESCOLA DE ENGENHARIA DE SÃO CARLOS

HENRIQUE DE ABREU RANGEL AGUIRRE

Implementação de controle digital de  
posição do motor CC em tempo real  
utilizando MATLAB e Simulink

São Carlos  
2012



HENRIQUE DE ABREU RANGEL AGUIRRE

# Implementação de controle digital de posição do motor CC em tempo real utilizando MATLAB e Simulink

Trabalho de Conclusão de Curso apresentado à Escola de Engenharia de São Carlos, da Universidade de São Paulo

Curso de Engenharia Elétrica com ênfase em Eletrônica

ORIENTADOR: Manoel Luis de Aguiar

São Carlos  
2012

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,  
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS  
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

A284i Aguirre, Henrique de Abreu Rangel  
Implementação de controle digital de posição do  
motor CC em tempo real utilizando MATLAB e Simulink /  
Henrique de Abreu Rangel Aguirre; orientador Manoel  
Luis de Aguiar. São Carlos, 2012.

Monografia (Graduação em Engenharia Elétrica com  
ênfase em Eletrônica) -- Escola de Engenharia de São  
Carlos da Universidade de São Paulo, 2012.

1. Controle de posição. 2. Motor CC. 3. Real-time  
Workshop. 4. MATLAB. 5. Simulink. I. Título.

# FOLHA DE APROVAÇÃO

Nome: Henrique de Abreu Rangel Aguirre

Título: "Implementação de controle digital de posição do motor CC em tempo real utilizando MATLAB e Simulink"

*Trabalho de Conclusão de Curso defendido e aprovado  
em 03 / 12 / 2012,*

*com NOTA 8,0 (oitos, zero), pela Comissão Julgadora:*

*Prof. Dr. Manoel Luís de Aguiar (Orientador)*  
*SEL/EESC/USP*

*Prof. Dr. Rodrigo Andrade Ramos*  
*SEL/EESC/USP*

*M.Sc. Eduardo Sacogne Fraccaroli*  
*ICMC/USP*

Coordenador da CoC-Engenharia Elétrica - EESC/USP:  
Prof. Associado Homero Schiabel



## **Dedicatórias**

Dedico este trabalho aos meus pais, Hugo e Lêda, por terem se dedicado à minha educação de forma sempre presente e motivante.

À tia Estela, pelo exemplo de vida.

Aos amigos de Rio Preto, que para sempre lembrarei das boas risadas e momentos nostálgicos.

Aos amigos de São Carlos, que fizeram deste lugar o melhor lugar do mundo e dos últimos 5 anos os melhores da minha vida.

À minha namorada, Aline, pelo companheirismo, amor e carinho.





## **Agradecimentos**

Agradeço ao Professor Manoel pelas dicas valiosas e pelas várias horas dedicadas a me ajudar na conclusão deste trabalho.

Aos técnicos de laboratório Alessandro e César pela grande ajuda e confiança cedidas para utilizar o Laboratório de Ensino do SEL.

Aos professores do campus da USP em São Carlos, em especial aos professores do Departamento de Engenharia Elétrica e de Computação, pela dedicação e comprometimento à sólida formação dos alunos da Escola de Engenharia de São Carlos da USP.



## Resumo

AGUIRRE, H. A. R. **Implementação de controle digital de posição do motor CC em tempo real utilizando MATLAB e Simulink**. 2012. A284i. Trabalho de Conclusão de Curso – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2012.

O trabalho aqui desenvolvido tem por objetivo a análise do desempenho de controladores digitais disponíveis na literatura aplicados ao controle de posição do motor CC, e a eficiência do controle em tempo real utilizando o *toolbox Real-Time Workshop* do Simulink.

Em uma primeira etapa, serão feitos estudos dos recursos do *toolbox Real-Time Workshop*. Na segunda etapa, será implementado o circuito tipo ponte H de acionamento do motor, bem como os circuitos de isolamento óptico entre a placa de aquisição de dados National Instruments AT-MIO-16E-10 e o circuito de potência. Na terceira etapa, com a interface microcomputador-motor concluída, o modelo dinâmico do motor CC será encontrado a partir de testes de degrau de tensão e o *System Identification Tool* do MATLAB.

Finalmente, com a dinâmica do motor modelada, poderão ser investigados diferentes tipos de controladores, tais como Proporcional (P), Proporcional-Integral (PI), Proporcional-Integral-Derivativo (PID) com auxílio do MATLAB e *toolbox Real-Time Workshop* do Simulink.

**Palavras-chave:** Controle digital de posição. Motor CC. *Real-Time Workshop*. MATLAB. Simulink.



## **Abstract**

AGUIRRE, H. A. R. **Implementation of DC motor digital position control in real time with MATLAB and Simulink**. 2012. A284i. Trabalho de Conclusão de Curso – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2012.

The document here presented aims to analyze the performance of digital controllers available in the literature applied to DC motor position control, as well as real-time control efficiency with Simulink Real-Time Workshop.

On a first step, studies of the resources of Real-Time Workshop toolbox will be made. On a second step, the H-bridge circuit will be implemented in order to drive the motor, as well as the optical isolation between the National Instruments AT-MIO-16E-10 data acquisition board and the power circuitry. On a third step, with the PC-Motor interface already designed and implemented, the DC motor dynamic model will be found with voltage step tests and MATLAB System Identification Tool.

Finally, with the known DC motor dynamics, different types of controllers, such as Proportional (P), Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) will be investigated and implemented with MATLAB and Simulink Real-Time Workshop.

**Keywords:** Digital position control. DC motor. Real-Time Workshop. MATLAB. Simulink.



## Lista de ilustrações

Figura 1.1 - Visão geral do sistema de controle.....	20
Figura 2.1 – Processo de rápida prototipação. Adaptado de (THE MATHWORKS, INC., 1999). .....	24
Figura 2.2 - <i>Real-time Windows Target</i> .....	24
Figura 2.3 - Sistema elétrico do motor CC (KARA e EKER, 2003).....	26
Figura 2.4 - Sistema mecânico do motor CC (KARA e EKER, 2003) .....	27
Figura 2.5 - Diagrama de blocos do motor CC.....	28
Figura 2.6 - Diagrama de blocos do modelo não-linear do motor CC .....	28
Figura 2.7 - Diferentes modelos para efeitos não-lineares (KARA e EKER, 2003) ...	29
Figura 2.8 - Diagrama do Simulink para identificação dos parâmetros do motor CC	32
Figura 2.9 - Subsistema de aquisição da corrente elétrica.....	33
Figura 2.10 - Subsistema do motor.....	33
Figura 2.11 - Subsistema do <i>encoder</i> .....	34
Figura 2.12 - System Identification Tool.....	35
Figura 2.13 - Import Data.....	35
Figura 2.14 - Process Models.....	36
Figura 2.15 – Esquema de controle digital (AGUIAR, 2011).....	37
Figura 2.16 - Configuração básica de um sistema de controle digital realimentado (AGUIAR, 2011).....	38
Figura 2.17 - Controlador PID (GUERRA, GONÇALVES JÚNIOR e LAGES, 2002). 38	
Figura 2.18 - Controlador PID Digital (GUERRA, GONÇALVES JÚNIOR e LAGES, 2002) .....	39
Figura 2.19 - Sistema de controle em malha fechada – tempo discreto.....	42
Figura 2.20 - Região de estabilidade. Adaptado de (FRANKLIN, POWELL e EMAMI-NAEINI, 2002).....	43
Figura 2.21 – Página inicial - rltool.....	43
Figura 2.22 - Aba "Compensator Editor" - rltool.....	44
Figura 2.23 - Visão geral da placa de aquisição de dados (NATIONAL INSTRUMENTS CORPORATION, 1996).....	45
Figura 2.24 - Pinagem da placa de aquisição de dados (NATIONAL INSTRUMENTS CORPORATION, 1996) .....	46
Figura 2.25 - Configuração interna do CI L298 (STMICROELECTRONICS, 2000) ..	46
Figura 2.26 - (a) Tensão do motor sem ponte H; (b) tensão do motor com ponte H; resolução de 2V/div e 50ms/div. ....	47

Figura 2.27 - Motor acionado por PWM com duty cycle de 50%.....	48
Figura 2.28 - Configuração interna do CI TIL111 (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000) .....	48
Figura 2.29 - Decodificação de quadratura .....	49
Figura 2.30 - Flip-flop tipo D para decodificação da quadratura.....	49
Figura 2.31 - (a) Configuração interna do CI 74LS74; (b) lógica de funcionamento (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000) .....	50
Figura 2.32 - (a) Configuração interna do CI 74LS86; (b) lógica de funcionamento (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000) .....	50
Figura 2.33 - Multiplicação da frequência do <i>encoder</i> .....	51
Figura 3.1 – Laboratório de Controle .....	53
Figura 3.2 - Resposta ao degrau - Corrente elétrica .....	55
Figura 3.3 - Comparação entre o modelo levantado e o sistema real – sistema elétrico .....	56
Figura 3.4 - Resposta ao degrau - velocidade angular.....	57
Figura 3.5 – Comparação entre o modelo levantado e o sistema real – sistema mecânico .....	57
Figura 3.6 - Controlador Proporcional - variação de $K_p$ .....	59
Figura 3.7 - Resposta ao degrau - Controle PI .....	60
Figura 3.8 - Lugar das raízes.....	61
Figura 3.9 - Detalhe do lugar das raízes .....	61
Figura 3.10 - Simulação de controlador PI.....	63
Figura 3.11 - Implementação de controlador PI .....	63
Figura 3.12 - Comparação entre a resposta ao degrau de posição do sistema real e do sistema ideal .....	64
Figura 3.13 - Comparação entre a ação de controle do sistema real e do sistema ideal.....	65
Figura 3.14 - Resposta à rampa .....	66
Figura A.1 - Diagrama esquemático do projeto.....	71



## **Lista de tabelas**

Tabela 3.1 - Resistência da armadura - Motor Electrocraft.....	54
Tabela 3.2 - Obtenção das constantes do motor .....	54
Tabela 3.3 - Parâmetros do motor de corrente contínua Electrocraft.....	58



## Sumário

1.	Introdução .....	19
1.1.	Objetivos do trabalho .....	20
1.2.	Motivação .....	20
1.3.	Visão geral .....	20
1.4.	Organização do documento .....	21
2.	Materiais e métodos .....	23
2.1.	<i>Real-Time Workshop</i> .....	23
2.2.	Motor de corrente contínua .....	25
2.2.1.	Modelagem do sistema dinâmico .....	25
2.2.2.	Efeitos não-lineares .....	28
2.2.3.	Estimação dos parâmetros do sistema.....	30
2.3.	Controle digital .....	37
2.3.1.	O método do lugar das raízes no tempo discreto com MATLAB .....	41
2.4.	Placa de aquisição de dados .....	44
2.5.	Circuito de acionamento.....	46
2.6.	Circuitos auxiliares .....	48
2.6.1.	Circuito de isolamento óptica .....	48
2.6.2.	Circuito decodificador de quadratura.....	49
2.6.3.	Circuito multiplicador de resolução.....	50
3.	Resultados .....	53
3.1.	Obtenção do modelo do motor CC.....	53
3.2.	Investigação do controlador proporcional.....	58
3.3.	Investigação do controlador PI.....	59
3.4.	Investigação do controlador PID .....	62
3.5.	Implementação e resultados .....	62
3.6.	Discussão .....	66
4.	Conclusão .....	67
	Referências Bibliográficas .....	69
	Apêndice A – Diagrama esquemático do projeto .....	71



## 1. Introdução

A primeira vez em que foi proposto o uso de computadores operando em tempo real como parte de um sistema de controle se deu em 1950 por Brown e Campbell (BENNETT, 1994). Ainda no final desta década, despontaram as primeiras aplicações industriais de controle digital. Não por iniciativa das indústrias de processo ou de manufatura, mas pela iniciativa de fabricantes de sistemas eletrônicos e de computação que procuravam pela expansão de seus mercados, bem como para encontrar aplicação a equipamentos que falharam na adoção pelos militares (WILLIAMS, 1977). A primeira instalação industrial de um sistema de controle digital em tempo real foi feita pela Texaco Company, que em 1959 instalou um sistema em malha fechada utilizando os recém-lançados RW-300 (Ramo-Wooldridge Company, atual TRW) na sua refinaria em Port Arthur, no Texas (ANON, 1959). Este esquema, no entanto, bem como cerca de outros 40 sistemas baseados no RW-300, não agiam diretamente nos atuadores das plantas. O primeiro sistema de controle digital direto (DDC, *Direct Digital Control*), i.e, que tinha ação direta sobre os atuadores, foi instalado em 1962 em uma planta da Imperial Chemical Industries, utilizando um sistema baseado nos computadores Ferranti Argus 200.

O rápido progresso da microeletrônica e dos microprocessadores possibilitou a aplicação de técnicas de controle digital em tempo real não somente nas indústrias de processo e de manufatura, mas em praticamente todos os meios de transporte modernos e cada vez mais nos bens de consumo e nas residências, com o advento da automação predial e residencial.

Uma maneira interessante de se projetar e simular sistemas de controle digital de sistemas dinâmicos é utilizando a suíte MATLAB/Simulink/*Real-Time Workshop*, ou simplesmente RTW. Com o Simulink, é possível modelar e simular sistemas dinâmicos simplesmente conectando blocos funcionais utilizando a notação matemática usual (QUARANTA e MANTEGAZZA, 2001). Esta facilidade, aliada ao fato de estar inteiramente integrado ao MATLAB, possibilita ao usuário a se concentrar na modelagem e em estratégias de controle, dispensando-o de aspectos de programação. O RTW gera automaticamente códigos em C/C++ a partir de diagramas do Simulink. O código gerado pode ser utilizado para aplicações em tempo-real ou não-tempo-real, incluindo prototipação rápida e simulações *hardware-in-the-loop* (THE MATHWORKS, INC., 1999).

### 1.1. Objetivos do trabalho

Este trabalho tem por objetivo a implementação de técnicas de controle digital aplicadas ao posicionamento de motores de corrente contínua utilizando o *toolbox Real-Time Workshop* do Simulink, em conjunto com a placa de aquisição de dados National Instruments AT-MIO-16E-10.

### 1.2. Motivação

A principal motivação para o desenvolvimento deste trabalho é o uso das técnicas aqui expostas no ensino de sistemas de controle digital para cursos de Engenharia, aliando teoria e experimentos inteiramente integrados ao ambiente MATLAB/Simulink. Este documento dá continuidade aos trabalhos de conclusão de curso realizados por (NEWTON, 2009) e (GUIMARÃES, 2011), contribuindo com novas abordagens para o controle de motores de corrente contínua.

### 1.3. Visão geral

O diagrama da Figura 1.1 fornece uma visão geral sob o aspecto construtivo do sistema desenvolvido.

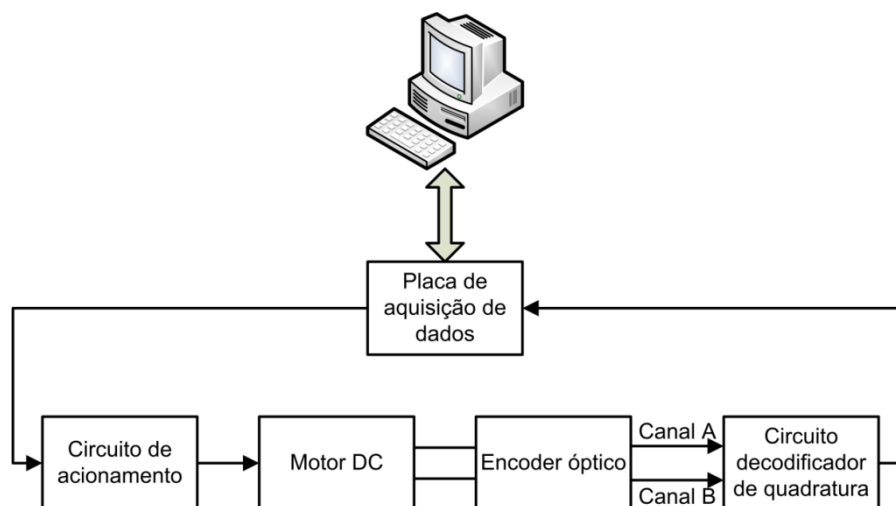


Figura 1.1 - Visão geral do sistema de controle

O *loop* de controle é representado pelo conjunto computador/placa de aquisição, onde as variáveis físicas são transformadas em código binário e as ações de controle são computadas. O circuito de acionamento consiste de um *drive* de corrente contínua na

topologia ponte H, o qual fornece ao motor a potência necessária exigida pelo controlador. Por fim, a posição e velocidade angular do motor CC podem ser calculadas a partir dos pulsos fornecidos pelo *encoder* incremental em quadratura acoplado ao seu eixo. No Apêndice A pode ser encontrado o diagrama esquemático geral do projeto.

#### **1.4. Organização do documento**

O documento foi organizado em uma estrutura contendo quatro capítulos.

O Capítulo 2 trata dos materiais e métodos utilizados para se implementar um controlador em tempo real utilizando o *toolbox Real-Time Workshop*, incluindo considerações sobre efeitos não-lineares que afetam o funcionamento de um motor de corrente contínua. Traz ainda informações sobre o circuito de acionamento escolhido, a placa de aquisição de dados e os circuitos auxiliares que fazem a interface entre os componentes do sistema. Neste capítulo, ainda é feita uma breve introdução sobre as estratégias de controle utilizadas e como implementá-las no ambiente Simulink.

O Capítulo 3 traz os principais resultados do projeto, e tem como principal objetivo comparar os resultados experimentais com os resultados obtidos através de simulações. Uma discussão sobre estes resultados é feita ao final do capítulo.

Por fim, o Capítulo 4 descreve as conclusões que puderam ser obtidas deste trabalho realizado ao longo do ano.





## 2. Materiais e métodos

### 2.1. *Real-Time Workshop*

O Simulink é um pacote do MATLAB para modelagem, simulação e análise de sistemas dinâmicos. Para a modelagem, o Simulink dispõe de uma interface gráfica ao usuário (GUI, *Graphical User Interface*) para construção de modelos a partir de diagrama de blocos. A partir do modelo construído, é possível iniciar uma simulação utilizando diferentes métodos de integração disponíveis no menu do Simulink. Devido ao fato de que o Simulink e o MATLAB serem inteiramente integrados, é possível analisar e revisar o modelo em qualquer um dos ambientes.

O *Real-Time Workshop* é um produto multi-plataforma desenvolvido para ser usado com MATLAB e Simulink. Este produto foi projetado para atender a diversas aplicações, dentre as quais podem ser citadas (THE MATHWORKS, INC., 1999):

- Controle em tempo-real
- Processamento de sinais em tempo real
- Simulação *hardware-in-the-loop*
- Sintonia de parâmetros em tempo real
- Geração de código em C/C++ portátil para outros programas de simulação

Neste trabalho, o controle em tempo-real e a sintonia de parâmetros em tempo-real serão extensamente abordados ao longo da identificação e projeto de controladores digitais.

Desta forma, o *Real-Time Workshop* é uma poderosa ferramenta para *rápida prototipação*, método que permite a implantação de soluções usando um ambiente de diagrama de blocos e testar a performance do sistema sem estar comprometido a um *design* específico. Com a rápida prototipação, pode-se refinar os modelos dos sistemas iterativamente sintonizando os parâmetros *on-the-fly*, i.e., diretamente sobre o diagrama de blocos, dispensando o projetista de atividades relacionadas ao desenvolvimento de *softwares*. A Figura 2.1 ilustra as vantagens do processo de rápida prototipação em relação ao método tradicional de implantação de um sistema.

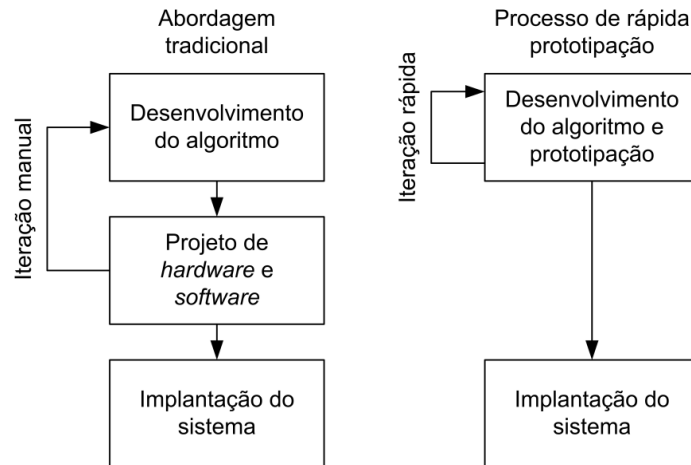


Figura 2.1 – Processo de rápida prototipação. Adaptado de (THE MATHWORKS, INC., 1999).

O *Real-Time Workshop* utiliza arquivos do tipo *target* (Figura 2.2) para traduzir os modelos construídos em códigos fonte e executáveis. Estes arquivos contêm as informações do ambiente em que o programa será executado, ou seja, especifica o *hardware* e sobre quais configurações o mesmo será utilizado.

Dentre os modelos de *targets* disponíveis, deve-se atentar quais deles o *hardware* efetivamente suporta. A placa de aquisição de dados National Instruments AT-MIO-16E-10 utilizada neste trabalho possui I/O digitais e analógicas e 2 I/O digitais de alta velocidade que podem ser usadas como contadores ou geradores de sinais PWM (*Pulse Width Modulation*). Embora fosse necessária uma porta de leitura de sinais de *encoders* de quadratura, a placa da National Instruments não contém este tipo de interface. Consequentemente, a leitura da velocidade e posição do motor deve ser feita utilizando um contador incremental tradicional.

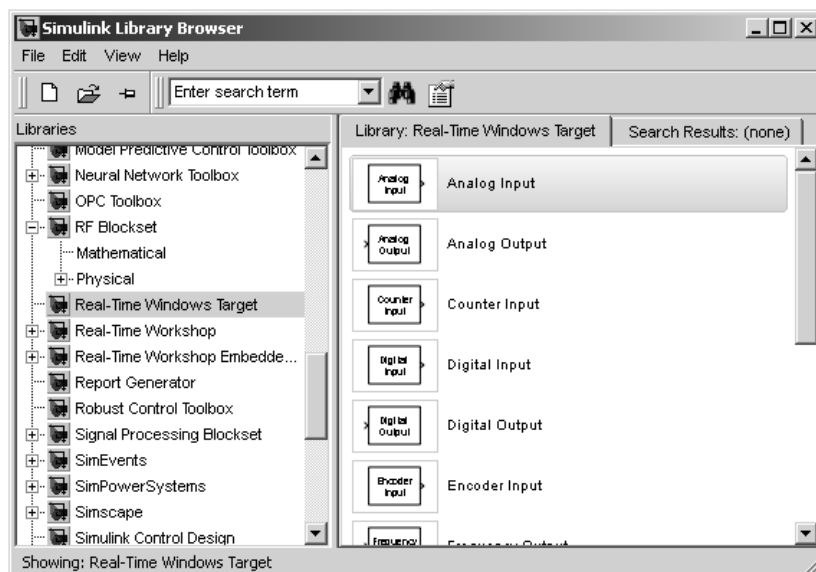


Figura 2.2 - Real-time Windows Target

## 2.2. Motor de corrente contínua

Motores de corrente contínua são muito utilizados em aplicações industriais devido à relativa facilidade de se implementar controladores de velocidade e posição, aliada à vasta região de operação que os mesmos podem trabalhar (KARA e EKER, 2003). Usualmente, a modelagem de motores CC não considera efeitos não-lineares, e se constrói uma função de transferência linear que relaciona a tensão aplicada em sua armadura à sua velocidade ou posição angular. Este modelo possui boa precisão quando o motor é operado próximo à sua região de operação nominal, porém comportamentos não-lineares podem afetar a precisão do controle quando o mesmo é operado a baixas velocidades ou quando é posto a mudar de direção.

### 2.2.1. Modelagem do sistema dinâmico

A identificação e modelagem de sistemas dinâmicos são essenciais no estágio de desenvolvimento de sistemas de controle. Desta forma, o uso de técnicas apropriadas de identificação de parâmetros é crucial para o projeto de controladores de grande precisão.

Os motores de corrente contínua são regidos por fenômenos elétricos e mecânicos que interagem entre si através de realimentação. Não se pode, portanto, modelá-los de forma independente. O passo que se deve tomar é na utilização de técnicas que identifiquem seus parâmetros levando em conta suas dependências.

O sistema elétrico do motor pode ser descrito pelas equações (2.1), (2.2) e (2.3), as quais descrevem o circuito elétrico da Figura 2.3.

$$v_a(t) = R_a i_a(t) + L_a \left( \frac{di_a(t)}{dt} \right) + e_a(t) \quad (2.1)$$

$$e_a(t) = \omega_m(t) k_e \quad (2.2)$$

$$T_m(t) = k_T i_a(t) \quad (2.3)$$

Onde  $v_a$  (V) é a tensão aplicada na armadura,  $R_a$  ( $\Omega$ ) é a resistência da armadura,  $i_a$  (A) é a corrente da armadura,  $L_a$  (H) é a indutância da armadura,  $e_a$  (V) é a força contra-eletromotriz,  $\omega_m$  (rad/s) é a velocidade angular do motor,  $k_e$  (Vs/rad) é a constante de força contra-eletromotriz,  $T_m$  (Nm) é o torque desenvolvido pelo motor e  $k_T$  (Nm/A) é a constante de torque.

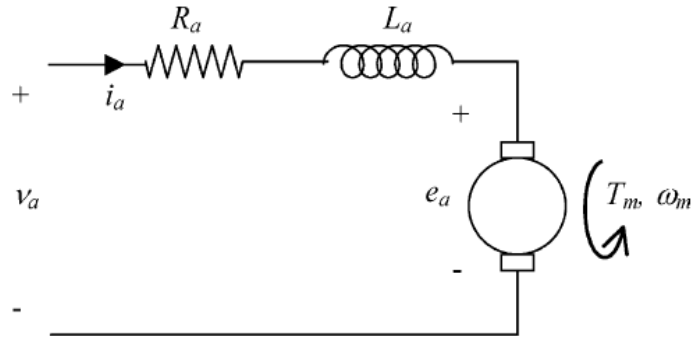


Figura 2.3 - Sistema elétrico do motor CC (KARA e EKER, 2003)

O sistema mecânico do motor pode ser modelado como duas massas girantes acopladas por um eixo, o qual pode ter sua massa e momento de inércia desprezados. A primeira massa representa o motor CC e a segunda massa representa a carga total aplicada ao eixo do motor. As equações (2.4), (2.5) e (2.6) descrevem o sistema mecânico ilustrado na Figura 2.4.

$$J_m \left( \frac{d\omega_m(t)}{dt} \right) = k_T i_a(t) - k_B \omega_m(t) - T_s(t) \quad (2.4)$$

$$J_L \left( \frac{d\omega_L(t)}{dt} \right) = T_s(t) - k_{BL} \omega_L(t) - T_d(t) \quad (2.5)$$

$$T_s(t) = k_s [\theta_m(t) - \theta_L(t)] + k_{Bs} [\omega_m(t) - \omega_L(t)] \quad (2.6)$$

Com:

$$\frac{d\theta_m(t)}{dt} = \omega_m(t), \quad \frac{d\theta_L(t)}{dt} = \omega_L(t) \quad (2.7)$$

Onde  $J_m$  e  $J_L$  (kg.m<sup>2</sup>) são os momentos de inércia do motor e da carga,  $\omega_m$  e  $\omega_L$  (rad/s) são as velocidades angulares do motor e da carga,  $k_B$  e  $k_{BL}$  (Nm/rad(s)) são as constantes de atrito viscoso do motor e da carga,  $T_s$  (Nm) é o torque transmitido pelo eixo,  $T_d$  (Nm) é o torque externo aplicado à carga,  $k_s$  (Nm/rad) é a constante elástica do eixo,  $k_{Bs}$  (Nm/rad(s)) é a constante de amortecimento do eixo e  $\theta_m$  e  $\theta_L$  (rad) são as posições angulares do motor e da carga.

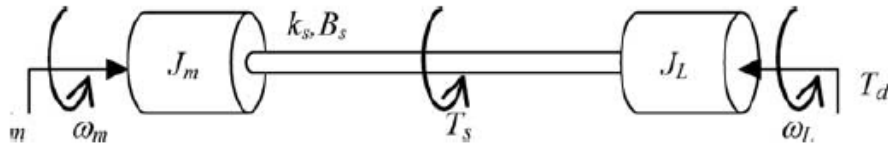


Figura 2.4 - Sistema mecânico do motor CC (KARA e EKER, 2003)

Considerando que o motor seja operado sem carga e livre de distúrbios externos, o sistema mecânico fica reduzido às equações (2.8) e (2.9).

$$J_m \left( \frac{d\omega_m(t)}{dt} \right) = k_T i_a(t) - k_B \omega_m(t) \quad (2.8)$$

$$\frac{d\theta_m(t)}{dt} = \omega_m(t) \quad (2.9)$$

Aplicando a transformada de Laplace às equações (2.1), (2.8) e (2.9), temos:

$$V_a(s) = R_a I_a(s) + L_a I_a(s)s + \Omega_m(s)k_e \quad (2.10)$$

$$s\Omega_m(s)J_m = k_T I_a(s) - \Omega_m(s)k_B \quad (2.11)$$

$$s\Theta_m(s) = \Omega_m(s) \quad (2.12)$$

Agrupando as equações (2.10) e (2.11), pode-se encontrar a função de transferência que relaciona a tensão aplicada na armadura do motor à velocidade angular desenvolvida.

$$\frac{\Omega_m(s)}{V_a(s)} = \frac{k_T / J_m L_a}{s^2 + s \left( \frac{k_B}{J_m} + \frac{R_a}{L_a} \right) + \frac{k_B R_a + k_e k_T}{J_m L_a}} \quad (2.13)$$

Com esta função de transferência, e a partir da equação (2.12), pode-se encontrar a função de transferência que relaciona a tensão aplicada na armadura à posição angular do motor (Equação 2.14).

$$\frac{\Theta_m(s)}{V_a(s)} = \frac{k_T / J_m L_a}{s \left( s^2 + s \left( \frac{k_B}{J_m} + \frac{R_a}{L_a} \right) + \frac{k_B R_a + k_e k_T}{J_m L_a} \right)} \quad (2.14)$$

Estas funções de transferência podem ser facilmente visualizadas a partir do diagrama de blocos da Figura 2.5.

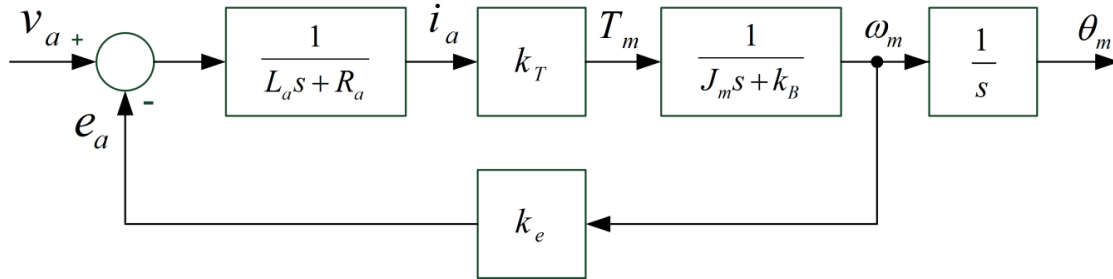


Figura 2.5 - Diagrama de blocos do motor CC

### 2.2.2. Efeitos não-lineares

Comportamentos não-lineares, como atrito de Coulomb e zona morta, influenciam significativamente na operação do motor quando este é posto a mudar de direção, ou quando é operado a baixas velocidades. Estes comportamentos não serão considerados no modelo geral do motor CC deste trabalho, porém serão devidamente descritos ao passo que justificam uma série de fenômenos observados na etapa de implantação do sistema de controle em tempo-real.

A Figura 2.6 ilustra em forma de diagrama de blocos um motor de corrente contínua em que se consideram forças de atrito não lineares, onde  $T_f(\omega_m)$  (Nm) é a força de atrito não-linear resultante e  $T_d$  (Nm) é o torque resultante de distúrbios externos.

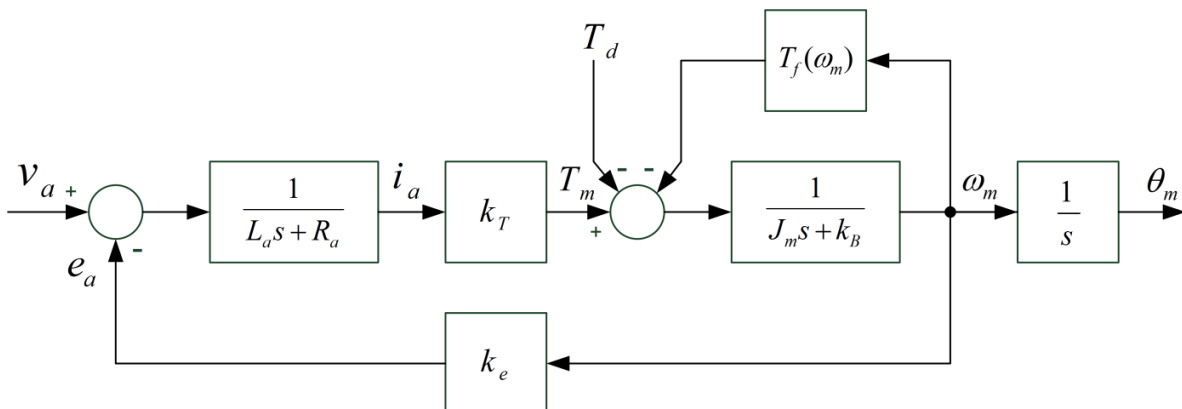


Figura 2.6 - Diagrama de blocos do modelo não-linear do motor CC

Para motores que operam exclusivamente a baixas velocidades, pode-se considerar que existe uma força de atrito estático que se opõe ao movimento quando  $\omega = 0$ . Enquanto

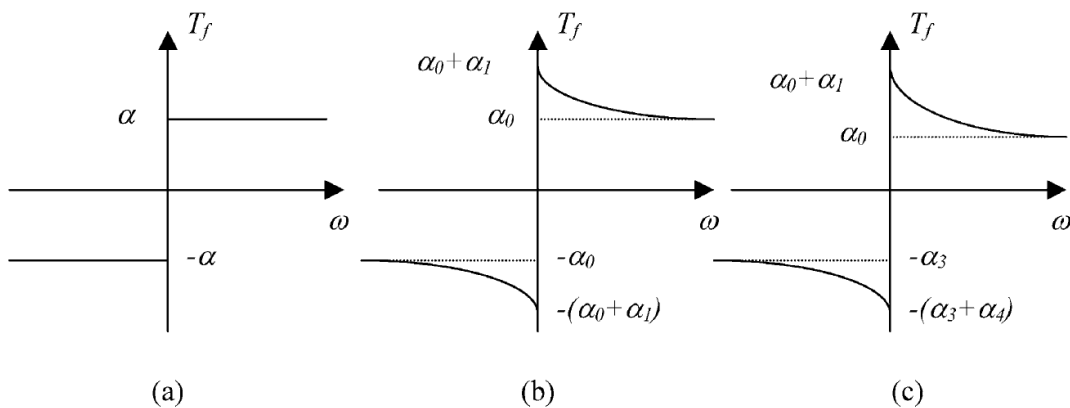
o torque fornecido pelo motor não for maior do que esta força de atrito estático, o motor permanece em repouso.

Para motores operando a altas velocidades, i.e., caso a operação predominante do motor não inclua baixas velocidades, exceto no momento inicial de aceleração, o modelo da Figura 2.7a representa bem a força de atrito de Coulomb predominante. Neste caso, a força de atrito pode ser representada por uma função *senal* dependente da velocidade:

$$T_f(\omega) = \alpha \operatorname{sgn}(\omega) \quad (2.15)$$

Onde:

$$\operatorname{sgn}(\omega) = \begin{cases} 1 & , \omega > 0 \\ 0 & , \omega = 0 \\ -1 & , \omega < 0 \end{cases} \quad (2.16)$$



**Figura 2.7 - Diferentes modelos para efeitos não-lineares (KARA e EKER, 2003)**

Para os casos em que motor opera intermitentemente entre períodos de alta e baixa velocidade, incluindo mudanças no sentido de rotação, o modelo matemático ilustrado pela Figura 2.7b representa a transição entre forças de atrito estático e cinético por um termo exponencial dependente da velocidade.

$$T_f(\omega) = \alpha_0 \operatorname{sgn}(\omega) + \alpha_1 e^{-\alpha_2 |\omega|} \operatorname{sgn}(\omega) \quad (2.17)$$

Onde  $\alpha_i$  são constantes positivas,  $i = 0, 1, 2$ .

O modelo da Figura 2.7c é uma generalização do modelo da Figura 2.7b pois considera a assimetria do movimento do motor entre as duas direções de rotação.

Usualmente, sistemas de controle de posição exigem que o motor opere transitando entre períodos de alta velocidade - nos casos em que o motor se encontra afastado de seu *setpoint* - e baixas velocidades - nos casos em que motor se aproxima da sua posição de referência.

A correta identificação dos parâmetros não-lineares está diretamente relacionada à precisão com que o controlador poderá atuar. Por outro lado, a identificação destes parâmetros não é uma tarefa trivial, e mesmo que o modelo linear do motor não seja suficiente para descrever todos os fenômenos envolvidos, neste trabalho as não-linearidades serão *desprezadas*.

Embora os efeitos não-lineares predominantes estejam identificados e modelados, procedimentos convencionais para identificação de parâmetros não são suficientemente acurados para esta finalidade. Tendo em vista que técnicas avançadas de modelagem de sistemas não cabem no escopo deste trabalho, a precisão dos controladores, bem como a similaridade entre as respostas provenientes do modelo teórico e do sistema real serão ligeiramente prejudicadas.

### **2.2.3. Estimação dos parâmetros do sistema**

Os parâmetros do motor de corrente contínua apresentados na seção 2.2.1 podem ser estimados de diversas maneiras. Quando não se dispõe de uma placa de aquisição de dados, é necessário que os parâmetros sejam identificados individualmente a partir de uma sequência de experimentos, conforme sugerido por (AGUIAR, OLIVEIRA e VARGAS, 2005)

Tendo em vista que neste projeto se dispõe de uma placa de aquisição de dados, pode-se utilizar o *System Identification Tool* do MATLAB, o qual constrói modelos matemáticos de sistemas dinâmicos a partir de dados de entrada e saída do sistema.

Utilizando-se desta abordagem, poder-se-ia considerar o sistema dinâmico de 2ª ordem como uma “caixa-preta”, bastando somente um ensaio para adquirir os dados da velocidade desenvolvida pelo motor *versus* tempo a partir de um ensaio em degrau de tensão. Este método é bastante preciso para se encontrar a função de transferência do motor, mas ainda não é possível que os parâmetros físicos sejam identificados. Uma maneira prática e rápida de se encontrar todos os parâmetros do sistema a partir de poucos ensaios é aplicando o mesmo degrau de tensão, mas adquirindo os dados de velocidade e corrente desenvolvidos pelo motor.

Preliminarmente, efetuam-se dois ensaios (GUIMARÃES, 2011):



- Obtenção da resistência de armadura

Com o rotor do motor bloqueado, aplica-se uma pequena tensão em seus terminais e mede-se a corrente elétrica com um multímetro. A partir da lei de Ohm é possível encontrar o valor da resistência. É importante que este ensaio seja feito com o rotor bloqueado, pois em caso contrário, a força contra-eletromotriz ( $e_a$ ) gerada deveria ser considerada no cálculo. Esta, por sua vez, ainda não pode ser determinada pois não se sabe o valor da constante de força contra-eletromotriz ( $k_e$ ).

- Obtenção da constante de força contra-eletromotriz ( $k_e$ ) e constante de torque ( $k_T$ ):

Para a obtenção da constante de força contra-eletromotriz ( $k_e$ ), baseia-se na equação (2.18), referente ao sistema elétrico do motor.

$$v_a(t) = R_a i_a(t) + L_a \left( \frac{di_a(t)}{dt} \right) + k_e \omega(t) \quad (2.18)$$

Em regime permanente, i.e., quando a corrente e velocidade assumem valores constante, temos que:

$$\frac{di_a(t)}{dt} = 0 \quad (2.19)$$

Então a equação (2.18) pode ser reduzida a:

$$v_a(t) = R_a i_a(t) + k_e \omega(t) \quad (2.20)$$

Desta forma:

$$k_e = \frac{v_a(t) - R_a i_a(t)}{\omega(t)} \quad (2.21)$$

Portanto, basta aplicar uma tensão constante nos terminais e medir a corrente e velocidade desenvolvidos pelo motor para identificar o valor da constante  $k_e$ . Se expressas em unidades do sistema internacional, a constante de torque ( $k_T$ ) possui o mesmo valor da constante de força contra-eletromotriz ( $k_e$ ).

Com estas constantes, é possível agora identificar os parâmetros restantes do motor aplicando simplesmente um ensaio em degrau e adquirindo os valores de velocidade angular e corrente elétrica.

- Obtenção da função de transferência do sistema elétrico

Para se obter a função de transferência de 1º grau referente ao sistema elétrico do motor, aplica-se um degrau de tensão em sua armadura com o objetivo de adquirir os dados referentes à velocidade e corrente do motor. Estes dados são utilizados pelo *System Identification Tool* para estimação dos parâmetros de uma função de transferência linear de primeira ordem.

Para tanto, constrói-se um diagrama de blocos no Simulink conforme a Figura 2.8.

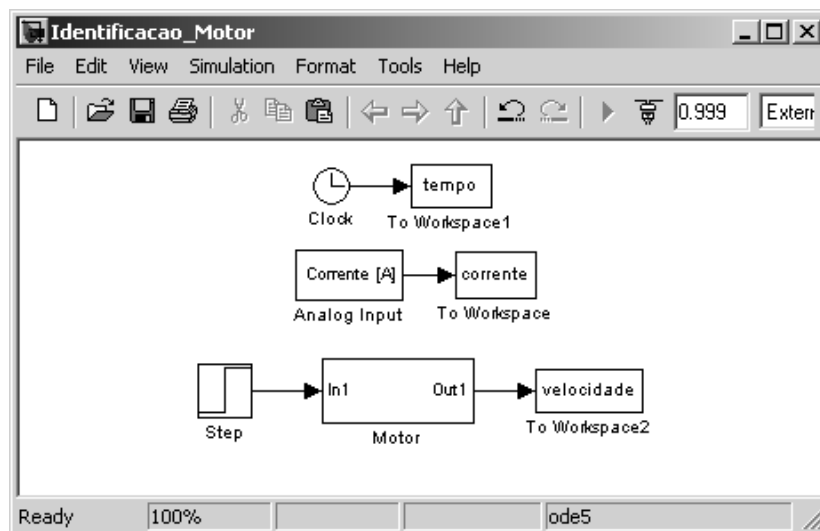


Figura 2.8 - Diagrama do Simulink para identificação dos parâmetros do motor CC

Este diagrama armazena os dados adquiridos do ensaio em degrau com auxílio dos blocos *to workspace*. Diferentemente do diagrama do Simulink utilizado para simulações convencionais, este diagrama é utilizado para uma aplicação em tempo real contendo arquivos do tipo *target*.

O subsistema intitulado *Analog Input* (Figura 2.9) é responsável pela aquisição dos dados da corrente elétrica do motor. A partir de uma associação paralela de dois resistores de  $1,2\Omega$  ligada em série com a armadura do motor (resistência equivalente de  $0,6\Omega$ ), a corrente elétrica é adquirida a partir da queda de tensão neste resistor. Antes do valor de tensão ser digitalizado pelo ADC (*Analog-to-digital converter*) da placa, um atenuador de razão 2:1 atua sobre este sinal, tanto para proteger o circuito interno do computador contra sobretensões, como para permitir que uma tensão de até 10V possa ser adquirida sem

danificar o ADC, que suporta somente 5V. Desta forma, o valor que a entrada analógica lê é multiplicado pelo fator de correção 2/0,6.

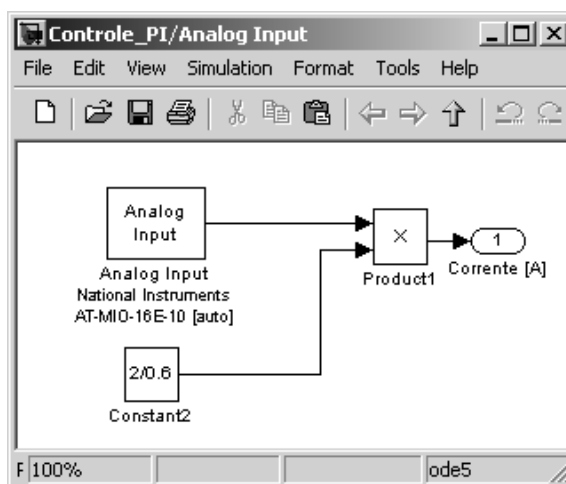


Figura 2.9 - Subsistema de aquisição da corrente elétrica

O subsistema intitulado *Motor* (Figura 2.10) contém os arquivos *target* necessários para geração do sinal de PWM, geração do sinal de sentido de rotação do motor, aquisição dos pulsos provenientes do *encoder* e leitura do sentido de rotação do motor.

O sinal de PWM é gerado pelo *target Frequency Output*. É necessário, pois, indicar a frequência e *duty cycle* que o mesmo deve operar. Primeiramente, para a determinação do *duty cycle*, o sinal de entrada é normalizado dividindo-o por 12, que é a tensão nominal da bateria que aciona o motor. Após a normalização, satura-se o mesmo para que não exiba valores superiores a 1 ou inferiores a -1 e se extrai seu módulo. A frequência de operação é fixada em 2 kHz pois esta é a taxa máxima de operação da placa de aquisição de dados.

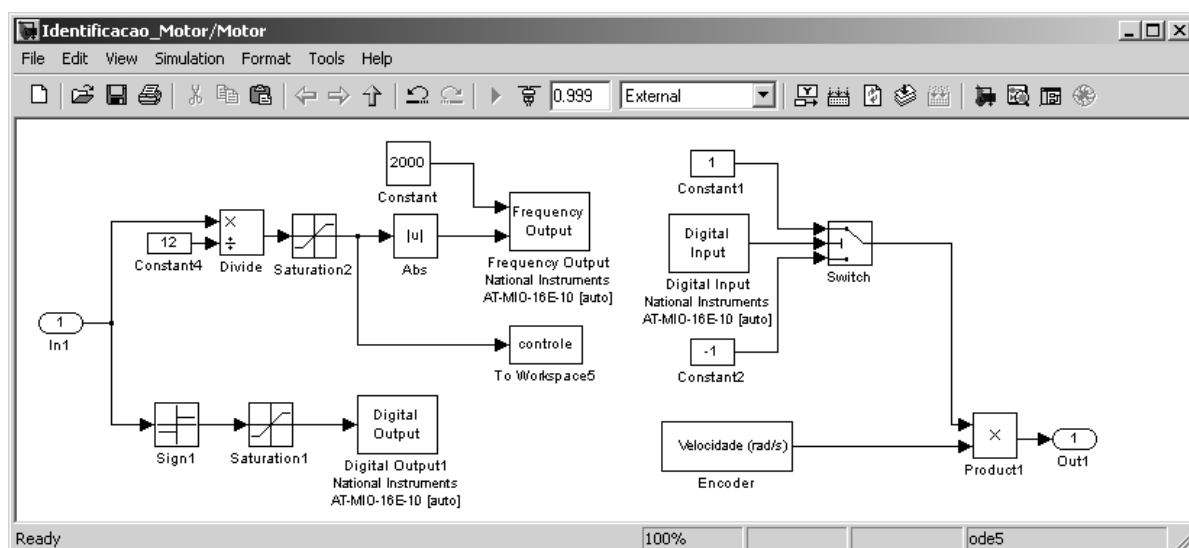


Figura 2.10 - Subsistema do motor

Deste mesmo sinal de entrada utilizado para definir o valor do *duty cycle*, extrai-se sua polaridade a partir do bloco *sign* e uma saturação de valores 1 e 0 transforma o sinal de entrada em 1 caso a polaridade seja positiva, e 0 caso a polaridade seja negativa. Pode-se então enviar este sinal via target *Digital Output* para o ambiente externo, o qual acionará o motor no devido sentido de rotação

A aquisição da velocidade e sentido de rotação do motor são feitos respectivamente pelo subsistema intitulado *Encoder* e pelo target *Digital Input*.

O sinal de sentido de rotação do motor é gerado via *hardware* pelo circuito decodificador de quadratura, e será explicado na seção 2.6.2. Como este sinal digital exhibe os valores 0 e 1, é necessário convertê-lo para -1 e 1. O bloco *switch* é responsável por esta operação. Caso o sinal do target *Digital Input* seja maior do que zero, a chave é ligada na posição superior, ligando o sistema à constante 1. Caso contrário, i.e., caso o sinal seja menor ou igual a zero, a chave é conectada na posição inferior, ligando o sistema à constante -1.

O subsistema intitulado *Encoder* está detalhado na Figura 2.11. Este subsistema é composto pelo target *Counter Input*, responsável pela contagem dos pulsos provenientes do *encoder*. Para cálculo de velocidade a partir de sinais provenientes de *encoder*, não há outra alternativa senão contar quantos pulsos ocorreram em espaços pré-determinados de tempo e fazer a devida conversão. No caso deste trabalho, o espaço pré-determinado de tempo é fixado em 1/2000 segundos, o qual é o tempo de amostragem utilizado. O *encoder* acoplado ao eixo do motor gera 1024 pulsos a cada volta. Na seção 2.6.3, um circuito multiplicador de resolução é utilizado para dobrar a frequência do *encoder* usando seus sinais em quadratura. Desta forma, a quantidade de pulsos ocorridos em 1/2000 segundos é multiplicado por uma constante para que a velocidade seja exibida em rad/s.

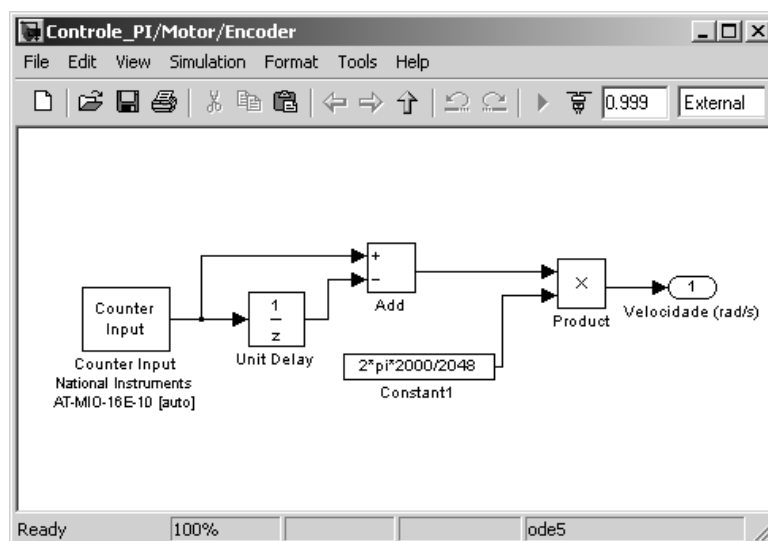


Figura 2.11 - Subsistema do *encoder*

A partir dos dados de corrente, velocidade e tensão, pode-se finalmente estimar o modelo do sistema elétrico. Conforme indica a Figura 2.5 - Diagrama de blocos do motor, o sistema elétrico do motor de corrente contínua é modelado pela função de transferência:

$$G_e(s) = \frac{I_a(s)}{V(s)} = \frac{1}{L_a s + R_a} \quad (2.22)$$

Onde:

$$V(s) = V_a(s) - E_a(s) \quad (2.23)$$

Por fim, basta usar o *System Identification Tool* para encontrar os parâmetros desconhecidos, onde o vetor de dados de entrada do sistema é igual a  $v_a(t) - \omega_m(t)k_e$ , e o vetor de dados de saída é igual a  $i(t)$ .

Nas janelas indicadas pelas Figura 2.12 e Figura 2.13, pode-se importar os dados que serão utilizados para estimação do modelo.

Basta então adicionar os dados de entrada e saída do sistema elétrico que foram adquiridos a partir do ensaio em degrau e definir o grau da função de transferência a ser modelada. Neste caso, o sistema elétrico é representado por uma função de transferência de primeiro grau com um pólo real, conforme Figura 2.14.

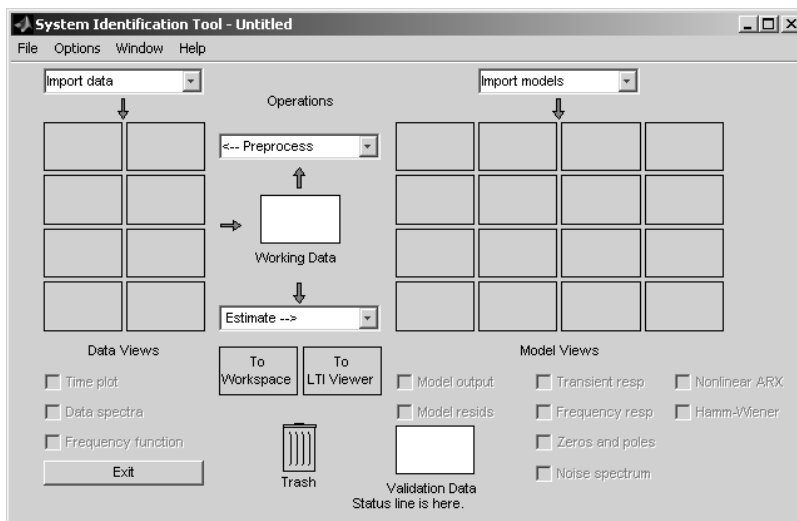


Figura 2.12 - System Identification Tool

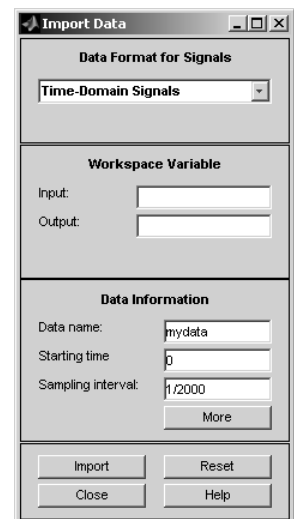


Figura 2.13 - Import Data

**Figura 2.14 - Process Models**

- Obtenção da função de transferência do sistema mecânico

Similarmente ao método proposto para estimação dos parâmetros do sistema elétrico, usa-se o *System Identification tool* para levantamento do modelo do sistema mecânico do motor.

A Figura 2.5 mostra que a função de transferência do modelo mecânico é:

$$G_m(s) = \frac{\Omega(s)}{T_m(s)} = \frac{1}{J_m s + k_B} \quad (2.24)$$

Onde:

$$T_m(s) = k_T I_a(s) \quad (2.25)$$

Portanto, basta repetir o método para estimação do modelo elétrico, fazendo as devidas alterações nos dados de entrada e saída do sistema. Desta vez, os dados de entrada do sistema são referentes ao torque,  $k_T i(t)$ , e os dados de saída são referentes à velocidade angular,  $\omega_m(t)$ .

### 2.3. Controle digital

A implementação de controle digital a sistemas contínuos no tempo traz inúmeras vantagens sobre os procedimentos de controle ditos analógicos. Os processos digitais trazem flexibilidade ao projeto, além de possibilitarem que outras tarefas sejam executadas além do controle propriamente, como supervisão do processo, coordenação e proteção (AGUIAR, 2011). Conceitos de teoria de controle moderno, como controle adaptativo e controle ótimo podem ser extremamente custosos ou irrealizáveis utilizando-se somente de dispositivos analógicos. Desta forma, o microprocessador como parte de um sistema de controle é amplamente utilizado nas indústrias.

O esquema geral de um sistema de controle digital é representado pela Figura 2.15.

Este esquema pode ser representado pela estrutura de controle da Figura 2.16. A partir de um determinado algoritmo, o microcomputador gera uma ação de controle entre dois instantes de amostragem. Os conceitos de amostragem e discretização podem ser estudados a partir da Transformada Z, a qual converte um sinal discreto no domínio do tempo em uma representação no domínio da frequência.

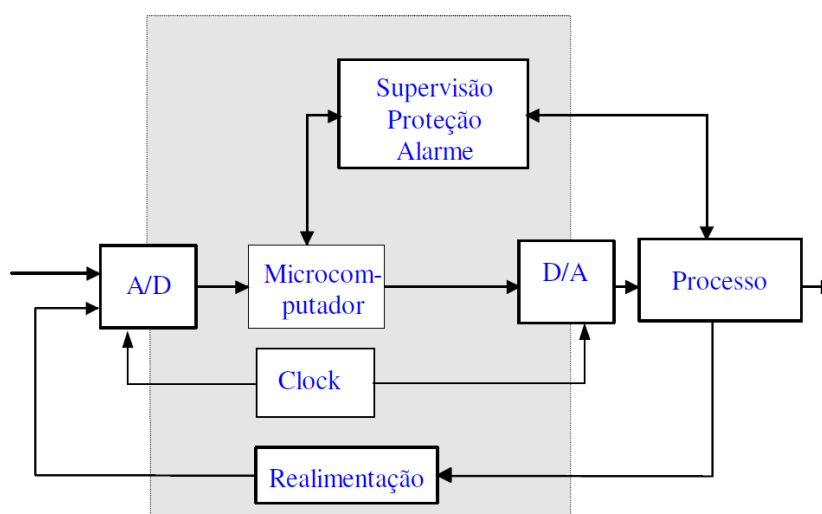


Figura 2.15 – Esquema de controle digital (AGUIAR, 2011)

O uso de controladores digitais possibilita o uso das mesmas estruturas utilizadas no controle clássico, como controlador proporcional (P), proporcional-integral (PI), proporcional-integral-derivativo (PID), avanço-atraso, controle por espaço de estados, bem como técnicas avançadas de controle ótimo, adaptativo ou difuso (*fuzzy*).

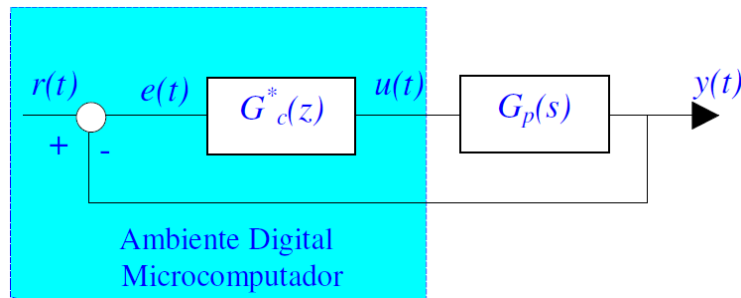


Figura 2.16 - Configuração básica de um sistema de controle digital realimentado (AGUIAR, 2011)

A maioria dos sistemas de controle realimentados determinam sua ação a partir da diferença entre um sinal de referência, aqui denotado  $r(t)$ , e o valor real da variável a ser controlada,  $y(t)$ . Uma estrutura muito utilizada para se realizar o controle de sistemas lineares é o controlador PID. Este controlador atua sobre o valor atual do erro, a integral do erro sobre um intervalo de tempo recente e a derivada do erro, para se determinar não somente quanta correção será aplicada, mas por quanto tempo (GUERRA, GONÇALVES JÚNIOR e LAGES, 2002). Cada uma destas três quantidades são multiplicadas por um fator ( $k_p$ ,  $k_i$  e  $k_d$  respectivamente) e então somadas. Portanto, a saída do controlador é o somatório indicado na Figura 2.17.

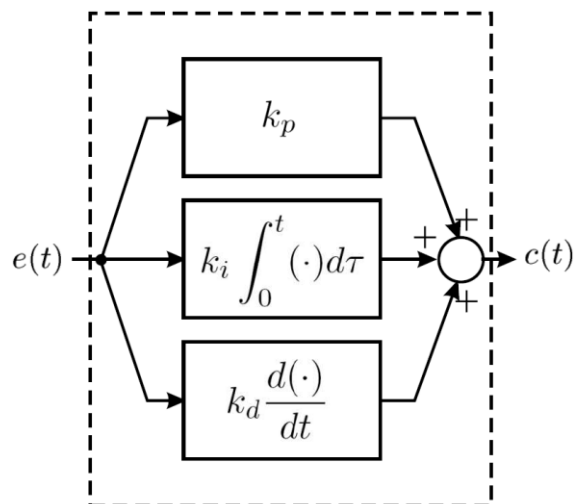


Figura 2.17 - Controlador PID (GUERRA, GONÇALVES JÚNIOR e LAGES, 2002)

O ajuste destas constantes determina quão rápida será a resposta do sistema. Esta velocidade, por sua vez, está correlacionada com a quantidade de sobressinal que a variável controlada apresentará durante a ação de controle. Desta forma, o ajuste deve ser feito considerando a performance desejável ou aceitável.

O modelo matemático do controlador PID contém três termos: (a) o termo proporcional P, correspondente à ação de controle proporcional, (b) o termo integral I,



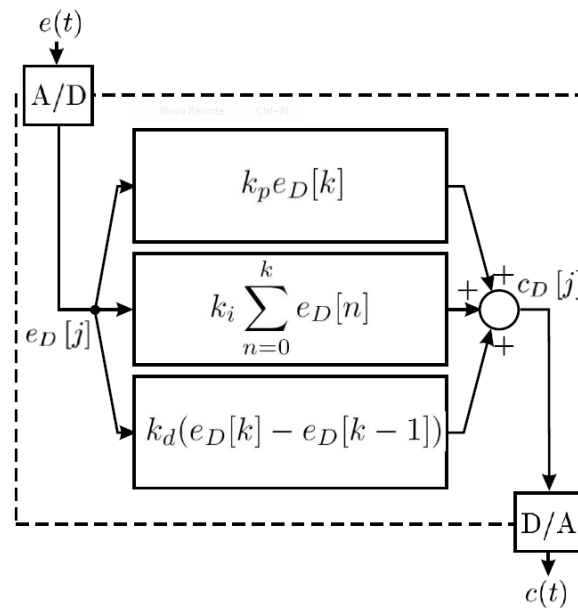
correspondente à ação de controle integral, e (c) o termo derivativo D, correspondente à ação de controle derivativa. A ação de controle pode ser calculada da seguinte forma:

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{de(t)}{dt} \quad (2.26)$$

É comum na literatura definir  $k_p = K$ ,  $k_i = K/T_i$  e  $k_d = KT_d$ . Desta forma, a ação de controle pode ser escrita como:

$$u(t) = K \left( e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (2.27)$$

Esta estrutura de controle PID considera variáveis analógicas no domínio do tempo contínuo. Uma versão digital deste controlador é representado pela Figura 2.18. Nesta versão, a ação integral é substituída por uma soma, e a derivativa por uma diferença.



**Figura 2.18 - Controlador PID Digital (GUERRA, GONÇALVES JÚNIOR e LAGES, 2002)**

O sinal contínuo no tempo é amostrado em intervalos fixos de tempo ( $T_0$ ). Um conversor analógico-digital (AD) faz a interface com o sinal de entrada, e um conversor digital-analógico (DA) faz a interface com o sinal de saída.

Discretizando o controlador representado pela equação 2.27, i.e., aproximando a ação de controle integral e derivativa para termos discretos, temos:

$$\frac{de(t)}{dt} \approx \frac{e(t) - e(t - T_0)}{T_0} \quad (2.28)$$

$$\int_0^t e(\tau) d\tau \approx T_0 \sum_{n=0}^{k-1} e(nT_0) \quad (2.29)$$

Com isto, a equação 2.27 pode ser aproximada por:

$$u(kT_0) = K \left( e(kT_0) + \frac{T_0}{T_i} \sum_{n=0}^{k-1} e(nT_0) + T_d \frac{e(kT_0) - e((k-1)T_0)}{T_0} \right) \quad (2.30)$$

Substituindo  $f(kT_0)$  por  $f[k]$ , a equação discreta do controlador PID fica:

$$u[k] = K \left( e[k] + \frac{T_0}{T_i} \sum_{n=0}^{k-1} e[n] + T_d \frac{e[k] - e[k-1]}{T_0} \right) \quad (2.31)$$

Subtraindo a equação 2.32 da equação 2.31:

$$u[k-1] = K \left( e[k-1] + \frac{T_0}{T_i} \sum_{n=0}^{k-2} e[n] + T_d \frac{e[k-1] - e[k-2]}{T_0} \right) \quad (2.32)$$

Temos:

$$u[k] - u[k-1] = K \left( e[k] - e[k-1] + \frac{T_0}{T_i} e[k-1] + T_d \frac{e[k] - 2e[k-1] + e[k-2]}{T_0} \right) \quad (2.33)$$

A equação 2.33 pode ser escrita de uma forma mais compacta:

$$u[k] - u[k-1] = K_1 e[k] + K_2 e[k-1] + K_3 e[k-2] \quad (2.34)$$

Onde:

$$K_1 = K \left( 1 + \frac{T_d}{T_0} \right) \quad (2.35)$$

$$K_2 = K \left( -1 + \frac{T_0}{T_i} - 2 \frac{T_d}{T_0} \right) \quad (2.36)$$

$$K_3 = K \frac{T_d}{T_0} \quad (2.37)$$

Por fim, aplicando a transformada Z à equação 2.34, temos:

$$U(z)(1 - z^{-1}) = E(z)(K_1 + K_2 z^{-1} + K_3 z^{-2}) \quad (2.38)$$

Portanto, a função de transferência discreta para o controlador PID digital é:

$$G_{PID}(z) = \frac{U(z)}{E(z)} = \frac{(K_1 + K_2 z^{-1} + K_3 z^{-2})}{(1 - z^{-1})} \quad (2.39)$$

### 2.3.1. O método do lugar das raízes no tempo discreto com MATLAB

A resposta transitória de um sistema em malha fechada é determinada a partir dos pólos da malha fechada (OGATA, 1985). O método do lugar das raízes possibilita determinar os pólos em malha fechada a partir dos pólos e zeros de malha aberta em função de um parâmetro variante. É frequentemente necessário ajustar um ou mais parâmetros para se obter uma localização desejável das raízes. Logo, é desejável determinar como as raízes da equação característica de um dado sistema se movimentam no plano Z quando os parâmetros variam; isto é, é usual determinar o lugar das raízes no plano Z (VILLAÇA, 2007).

Todas as regras derivadas para o domínio s são aplicáveis e podem ser usadas para se obter o lugar das raízes no domínio Z. O sistema de controle realimentado toma a forma da Figura 2.19, onde a planta e o compensador assumem suas representações discretas.

Variando o valor de um ganho  $k$  conforme a malha de controle da Figura 2.19, o diagrama do lugar das raízes mostra a trajetória dos pólos em malha fechada do sistema no plano Z. Tipicamente, o método do lugar das raízes é usado para sintonizar o ganho da malha de um sistema de controle especificando o lugar do conjunto de pólos do sistema.

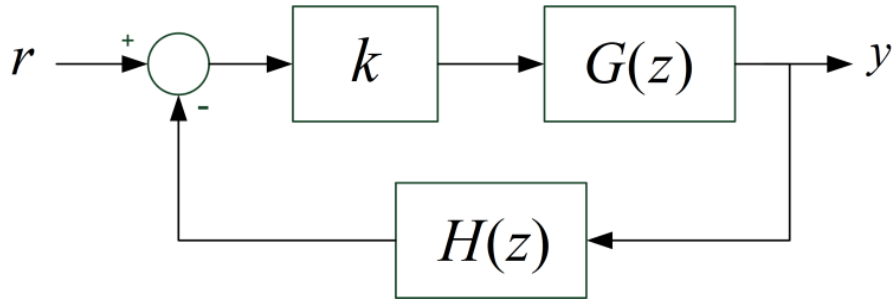


Figura 2.19 - Sistema de controle em malha fechada – tempo discreto

Considerando como exemplo o sistema da Figura 2.19, os pólos do sistema são caracterizados pelas raízes da equação característica (2.40), analogamente à equação característica no domínio  $s$ .

$$q(z) = 1 + kG(z)H(z) \quad (2.40)$$

A técnica do lugar das raízes consiste em analisar graficamente as trajetórias do pólos no plano  $Z$  de acordo com a variação do ganho  $k$ .

**Exemplo 1.1.** Considere a função de transferência discreta do sistema como proposto por (FRANKLIN, POWELL e EMAMI-NAEINI, 2002):

$$G(z) = \frac{1 - \alpha}{z - \alpha}$$

E  $C(z) = k$ . O lugar das raízes com respeito à variação do parâmetro  $k$  é indicado na Figura 2.20.

No domínio discreto, sistemas serão estáveis se os pólos da função de transferência em malha fechada estiverem contidos no círculo unitário  $|z| = 1$ . A Figura 2.20 indica, pois, um sistema que inevitavelmente se tornará instável para uma variação crescente de  $k$ .

Um lugar das raízes mais preciso pode ser obtido empregando-se o MATLAB. O MATLAB oferece uma ferramenta poderosa para projetos de controladores via método do lugar das raízes: *rltool*. Esta ferramenta permite o projeto interativo de compensadores de sistemas SISO (*Single-Input/Single-Output*).

Usando a interface gráfica ao usuário, é possível importar modelos de sistemas dinâmicos contínuos e discretos conforme tela indicada pela Figura 2.21.

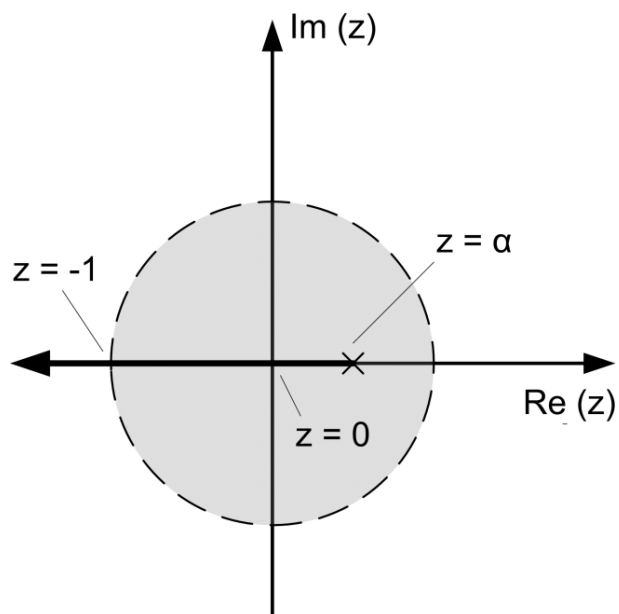


Figura 2.20 - Região de estabilidade. Adaptado de (FRANKLIN, POWELL e EMAMI-NAEINI, 2002)

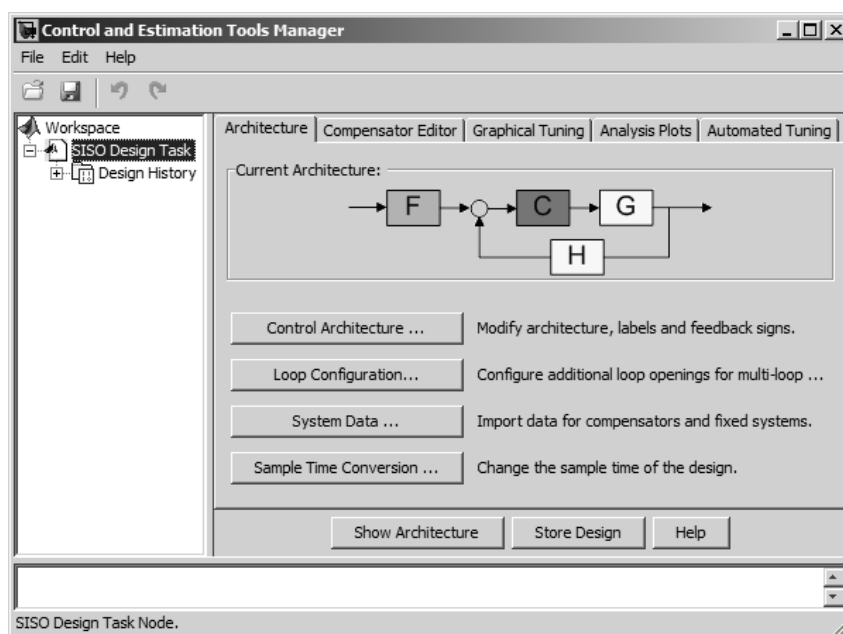


Figura 2.21 – Página inicial - rltool

Além disto, é possível adicionar ou deletar pólos e zeros para se projetar o compensador, e definir suas respectivas localizações no plano  $Z$  conforme mostra a Figura 2.22.

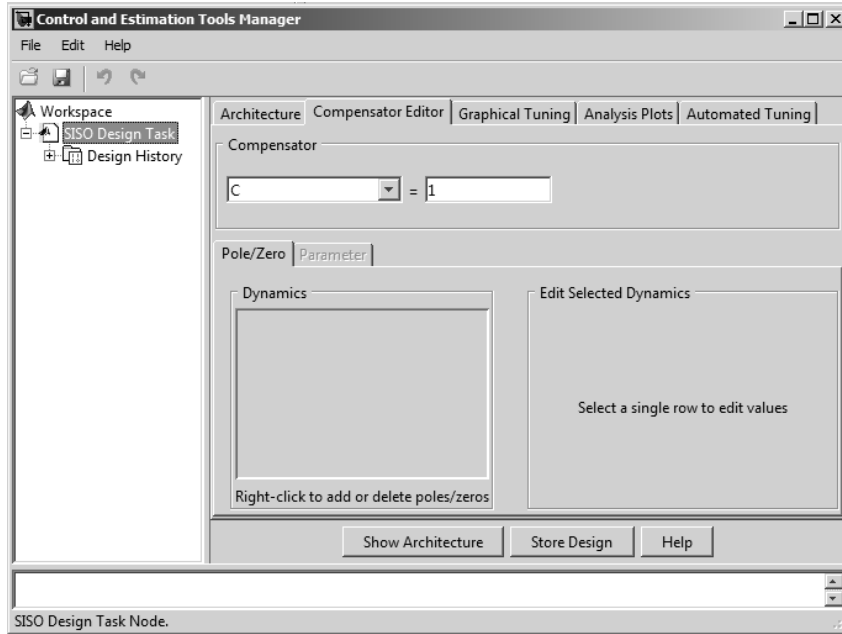


Figura 2.22 - Aba "Compensator Editor" - rltool

Para o caso de um compensador PID digital, usa-se a equação 2.41 para se definir a localização dos pólos e dos zeros que atendam aos requisitos de projeto.

$$G_{PID}(z) = \frac{U(z)}{E(z)} = \frac{(K_1 + K_2 z^{-1} + K_3 z^{-2})}{(1 - z^{-1})} \quad (2.41)$$

Com a localização dos pólos e zeros do compensador definidas, parte-se para a etapa de sintonia gráfica, i.e., os pólos do sistema em malha fechada podem ser movidos manualmente no plano Z e a resposta do sistema analisada em tempo real. O deslocamento dos pólos do sistema em malha fechada sobre o caminho do lugar das raízes altera o valor do ganho do compensador, conforme o método do lugar das raízes exposto nesta seção.

## 2.4. Placa de aquisição de dados

Fabricada pela National Instruments Corporation, a placa de aquisição AT-MIO-16E-10 possui 8 I/O digitais configuráveis por *software*, até 16 entradas analógicas com ADCs (*analog-to-digital converters*) de 12 bits, dois timers/counters de 24 bits e duas saídas analógicas com DACs (*digital-to-analog converters*) de 12 bits.

Figura 2.23 ilustra uma visão geral da placa de aquisição de dados e suas ligações internas. A Figura 2.24 ilustra a pinagem das portas de entrada e saída da placa. Os pinos utilizados neste projeto foram:



AIGND	1	2	AIGND
ACH0	3	4	ACH8
ACH1	5	6	ACH9
ACH2	7	8	ACH10
ACH3	9	10	ACH11
ACH4	11	12	ACH12
ACH5	13	14	ACH13
ACH6	15	16	ACH14
ACH7	17	18	ACH15
AISENSE	19	20	DAC0OUT
DAC1OUT	21	22	EXTREF
AOGND	23	24	DGND
DIO0	25	26	DIO4
DIO1	27	28	DIO5
DIO2	29	30	DIO6
DIO3	31	32	DIO7
DGND	33	34	+5 V
+5 V	35	36	SCANCLK
EXTSTROBE*	37	38	PFI0/TRIG1
PFI1/TRIG2	39	40	PFI2/CONVERT*
PFI3/GPCTR1_SOURCE	41	42	PFI4/GPCTR1_GATE
GPCTR1_OUT	43	44	PFI5/UPDATE*
PFI6/WFTRIG	45	46	PFI7/STARTSCAN
PFI8/GPCTR0_SOURCE	47	48	PFI9/GPCTR0_GATE
GPCTR0_OUT	49	50	FREQ_OUT

Figura 2.24 - Pinagem da placa de aquisição de dados (NATIONAL INSTRUMENTS CORPORATION, 1996)

## 2.5. Circuito de acionamento

O circuito integrado L298 foi utilizado para se fazer o acionamento do motor CC. Este circuito conta com duas pontes-H, acionadas via níveis convencionais tensão da lógica TTL, podendo acionar cargas indutivas, bem como relés, solenoides, motores CC e de passo.

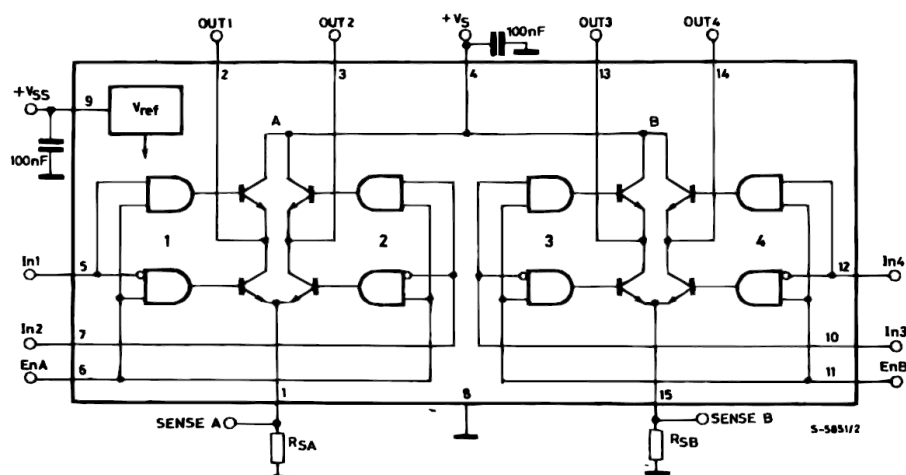


Figura 2.25 - Configuração interna do CI L298 (STMICROELECTRONICS, 2000)



A Figura 2.25 mostra as ligações internas do CI, bem como sua pinagem correspondente. Cada uma das pontes H existentes no circuito suporta picos não repetitivos de até 3 A ( $t = 100 \mu s$ ). O motor CC utilizado neste trabalho exige correntes próximas ao limite do dispositivo no instante de partida, portanto, para uma maior segurança, optou-se por conectar os dois canais da ponte H em paralelo para diminuir a corrente de pico em cada um deles.

Adicionalmente, a ponte H insere efeitos não lineares, principalmente na partida do motor. A Figura 2.26 demonstra o afundamento da tensão do motor quando se aplica um degrau diretamente da bateria (a); e quando um degrau é aplicado utilizando-se da ponte H um PWM com *duty cycle* de 100% (b). A queda de tensão de aproximadamente 2 V em regime permanente pode ser considerada quando se fazem simulações de respostas do motor, porém o afundamento de tensão presente no início da operação possui um comportamento não-linear de difícil modelagem, e que novamente foge do escopo do trabalho e não será considerado.

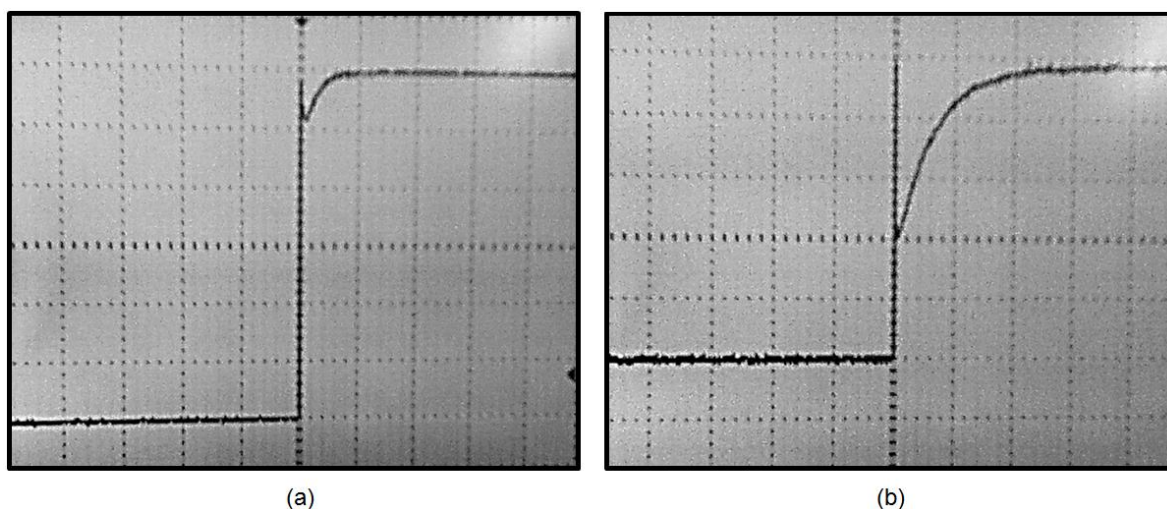


Figura 2.26 - (a) Tensão do motor sem ponte H; (b) tensão do motor com ponte H; resolução de 2V/div e 50ms/div.

Devido à relativa baixa frequência de acionamento do sinal de PWM de 2 kHz, o motor opera em condução descontínua conforme mostra a Figura 2.27. A curva superior é referente à tensão do motor para um sinal de PWM com *duty cycle* de 50%. A curva inferior mostra a corrente descontínua do motor para este mesmo sinal de PWM.

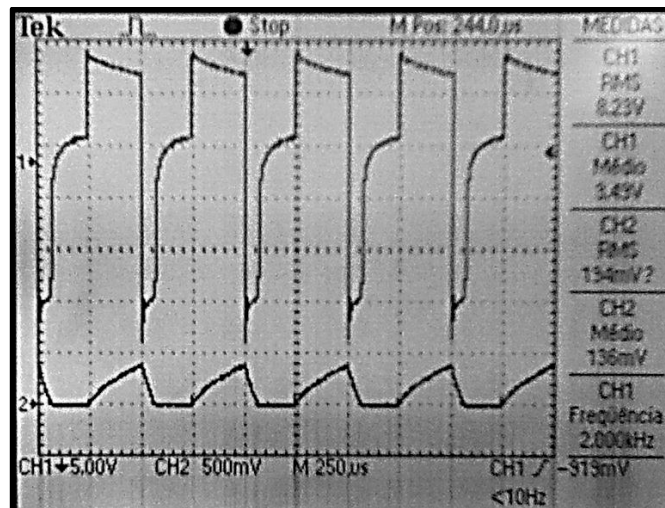


Figura 2.27 - Motor acionado por PWM com duty cycle de 50%

## 2.6. Circuitos auxiliares

### 2.6.1. Circuito de isolamento óptica

Para se fazer o isolamento óptico entre a placa de aquisição e o circuito de potência, foi utilizado o circuito integrado TIL111, constituído por um fototransistor acoplado a um diodo emissor de infravermelho.

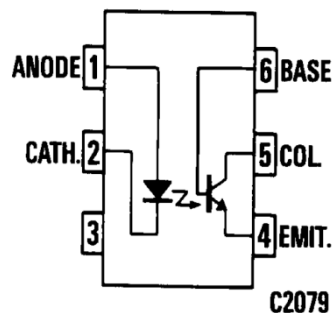


Figura 2.28 - Configuração interna do CI TIL111 (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000)

Este isolamento é necessário para se proteger a placa contra níveis de tensão espúrios que podem comprometer seu funcionamento. Para tanto, o transistor é ligado nas configurações coletor-comum e emissor-comum para condicionar os sinais de PWM e direção do sentido de rotação do motor.

### 2.6.2. Circuito decodificador de quadratura

O *encoder* óptico acoplado ao eixo do motor fornece dois sinais em quadratura, os quais podem ser usados tanto para medição de velocidade/posição do motor, como para se detectar a direção em que o motor está girando. Para tanto, são gerados dois sinais em forma de onda quadrada defasados  $90^\circ$  entre si, conforme mostra a Figura 2.29. Para cada direção de rotação do eixo do motor, o sinal A pode estar atrasado ou adiantado em relação ao sinal B.

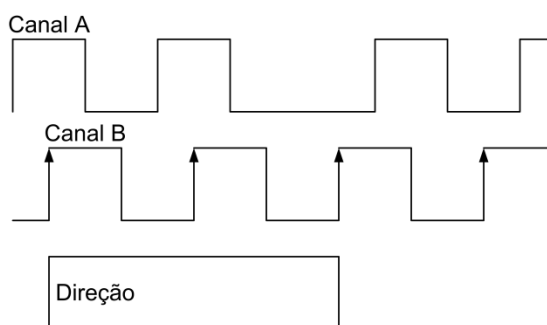


Figura 2.29 - Decodificação de quadratura

Para se decodificar a quadratura, i.e., detectar a direção de rotação do motor, utilizou-se o circuito integrado 74LS74 – Dual Flip Flop tipo D na configuração ilustrada na Figura 2.30.

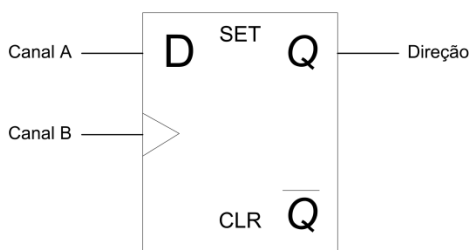


Figura 2.30 - Flip-flop tipo D para decodificação da quadratura

Conforme a tabela verdade da Figura 2.31 (b), e utilizando o canal B como sinal de *clock*, a saída do sistema é atualizada para o sinal de entrada a cada borda de subida do *clock*. Isto significa que caso o sinal A esteja adiantado em relação ao sinal B, a borda de subida do *clock* coincidirá sempre com o estado alto de A. Em caso contrário, onde o sinal A esteja atrasado em relação ao sinal B, a borda de subida do *clock* coincidirá com o estado baixo de A.

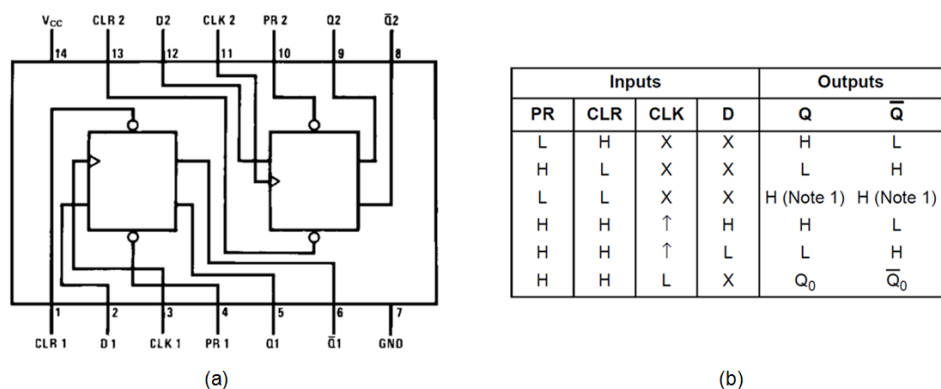


Figura 2.31 - (a) Configuração interna do CI 74LS74; (b) lógica de funcionamento (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000)

### 2.6.3. Circuito multiplicador de resolução

Embora os *encoders* de quadratura possibilitem que a frequência dos sinais seja multiplicada por quatro, utilizando-se para isto de todas as bordas de subida de ambos os sinais A e B, a placa de aquisição de dados utilizada neste projeto suporta uma frequência de até 100 kHz em suas portas de *timing*. Considerando que o *encoder* óptico do motor utilizado possui uma resolução de 1024 pulsos por volta, isto significa que caso seja usada a máxima resolução possível de ser obtida com os sinais em quadratura (4096 pulsos por volta), a máxima velocidade possível de ser medida será de apenas aproximadamente 154 rad/s.

Desta forma, optou-se por multiplicar a frequência do sinal do *encoder* por dois, ao invés de quatro. Para se dobrar a frequência do sinal do *encoder*, foi utilizado o circuito integrado 74LS86 – Porta lógica XOR (Ou-exclusivo).

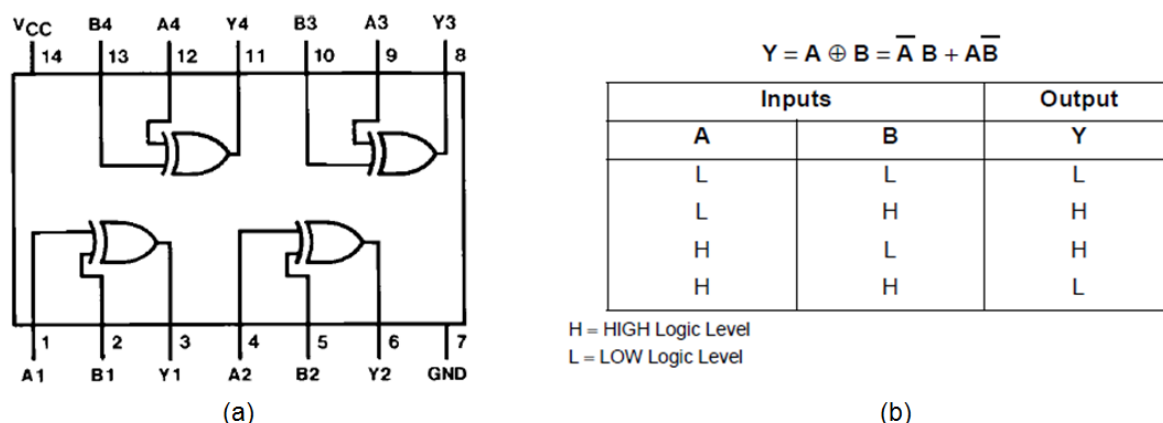
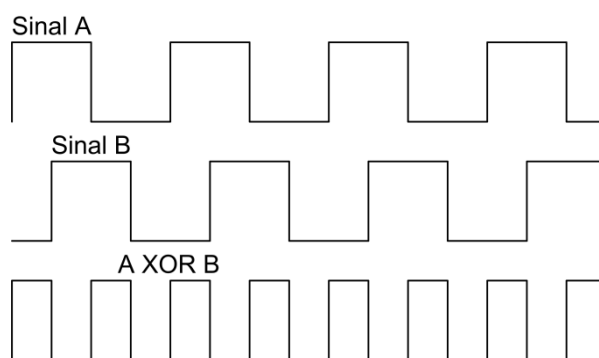


Figura 2.32 - (a) Configuração interna do CI 74LS86; (b) lógica de funcionamento (FAIRCHILD SEMICONDUCTOR CORPORATION, 2000)

A configuração interna do circuito integrado e sua lógica de operação podem ser vistas na Figura 2.32. Conectando o sinal A do *encoder* à porta A, e o sinal B à porta B, tem-se a saída ilustrada na Figura 2.33. Com isto, pode-se medir velocidades de rotação de até aproximadamente 308 rad/s com uma resolução de 2048 pulsos por volta.

Para que seja utilizada a máxima resolução possível de ser obtida com os sinais em quadratura sem comprometer a máxima velocidade medida, é necessário que o *hardware* de aquisição de dados suporte a frequência elevada proveniente desta operação.



**Figura 2.33 - Multiplicação da frequência do *encoder***



### 3. Resultados

Os resultados apresentados neste projeto foram obtidos através da integração dos métodos expostos na seção 2. A Figura 3.1 mostra os principais componentes utilizados no projeto, como a placa de bornes da placa de aquisição de dados, os circuitos auxiliares e o motor CC no local em que foram realizados os experimentos.

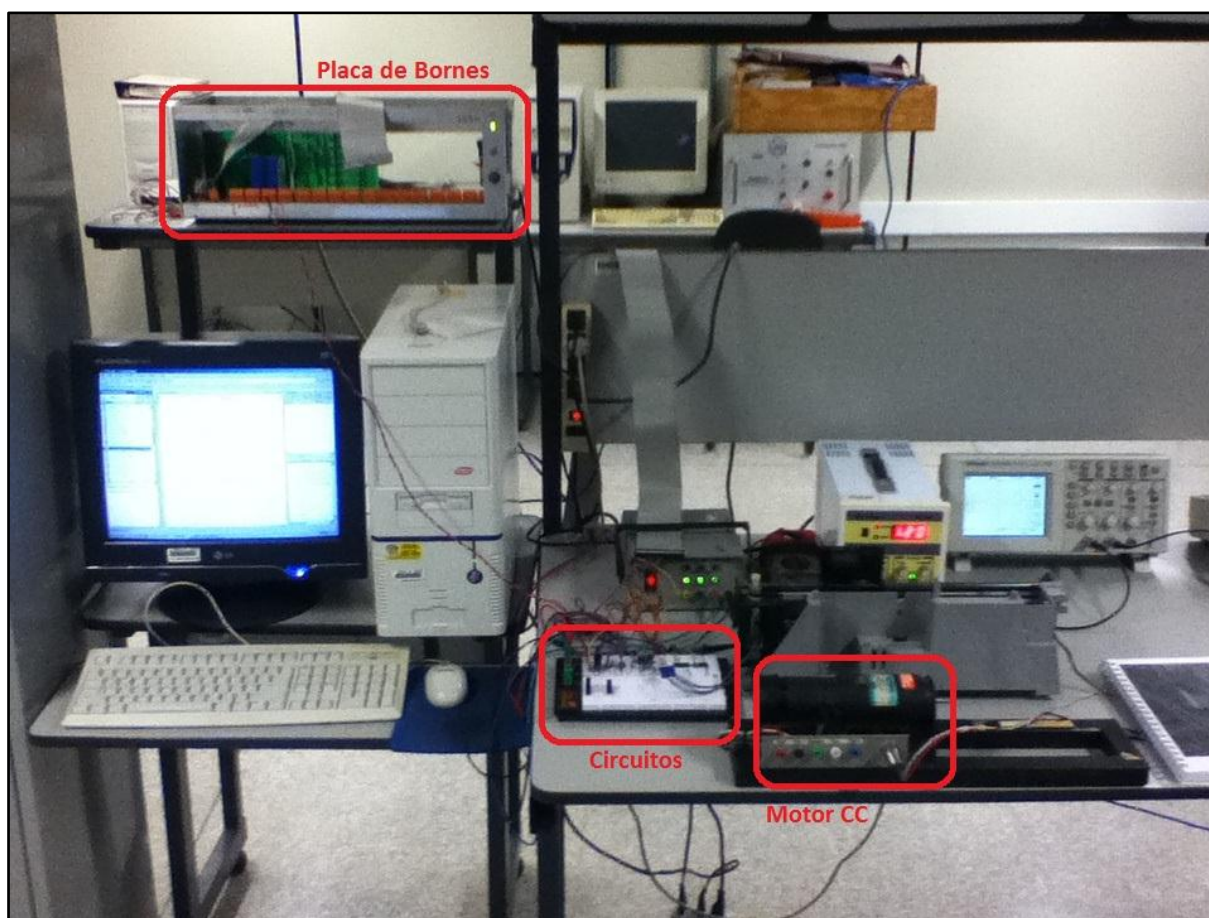


Figura 3.1 – Laboratório de Controle

#### 3.1. Obtenção do modelo do motor CC

Utilizando-se do método apresentado na seção 2.2.3, o modelo do motor de corrente contínua Electrocraft pôde ser obtido a partir de uma sequência de três ensaios.

- Resistência de armadura

A partir de uma pequena tensão aplicada nos terminais do motor, e mantendo-o com o rotor bloqueado em várias posições diferentes, a resistência da armadura para cada uma dessas posições pôde ser determinada conforme a Tabela 3.1. Conforme proposto por (AGUIAR, OLIVEIRA e VARGAS, 2005), a menor resistência encontrada é considerada para se encontrar os outros parâmetros do motor ( $R_a = 3,65 \, \Omega$ ).

**Tabela 3.1 - Resistência da armadura - Motor Electrocrafft**

<b>Resistência da armadura (<math>\Omega</math>)</b>
3,72
3,76
3,68
3,65
3,7
3,81

- Constante de força contra-eletromotriz e constante de torque

Com o motor mantido a uma velocidade constante, e a partir da Equação 3.1, pode-se obter os valores das constantes de força contra-eletromotriz e constante de torque.

$$k_e = \frac{v_a(t) - R_a i_a(t)}{\omega(t)} \quad (3.1)$$

Considerando a resistência da armadura o menor valor encontrado para as diferentes posições do rotor ( $R_a = 3,65 \, \Omega$ ), a Tabela 3.2 indica os valores de tensão, corrente e velocidade medidos, bem como o valor calculado da constante de força contra-eletromotriz.

**Tabela 3.2 - Obtenção das constantes do motor**

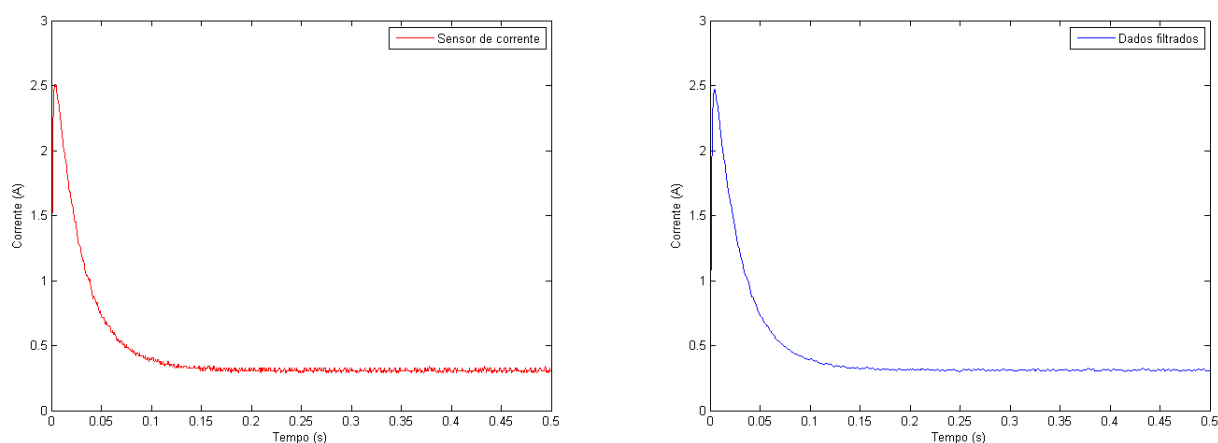
<b>Parâmetros</b>	<b>Valores medidos</b>
Tensão da armadura (V)	10,5
Corrente da armadura (A)	0,36
Velocidade angular (rad/s)	108
$k_e = \frac{v_a - R_a i_a}{\omega}$	0,085

Como  $k_e = k_t$ , ambas constantes são iguais a 0,085.



- Sistema elétrico do motor

A função de transferência do sistema elétrico do motor pode ser obtida através do método exposto na seção 2.2.3. Aplicando um degrau de tensão de 12V, desconsiderando o efeito de queda de tensão no circuito de acionamento, o motor apresenta uma resposta de corrente vs tempo conforme indicado na Figura 3.2. A curva vermelha representa os dados conforme adquiridos diretamente pela porta analógica da placa de aquisição. Para que o modelo a ser levantado não considere os ruídos da leitura, aplicou-se um filtro Butterworth para eliminar estes efeitos. A curva azul representa a corrente após a aplicação do filtro.

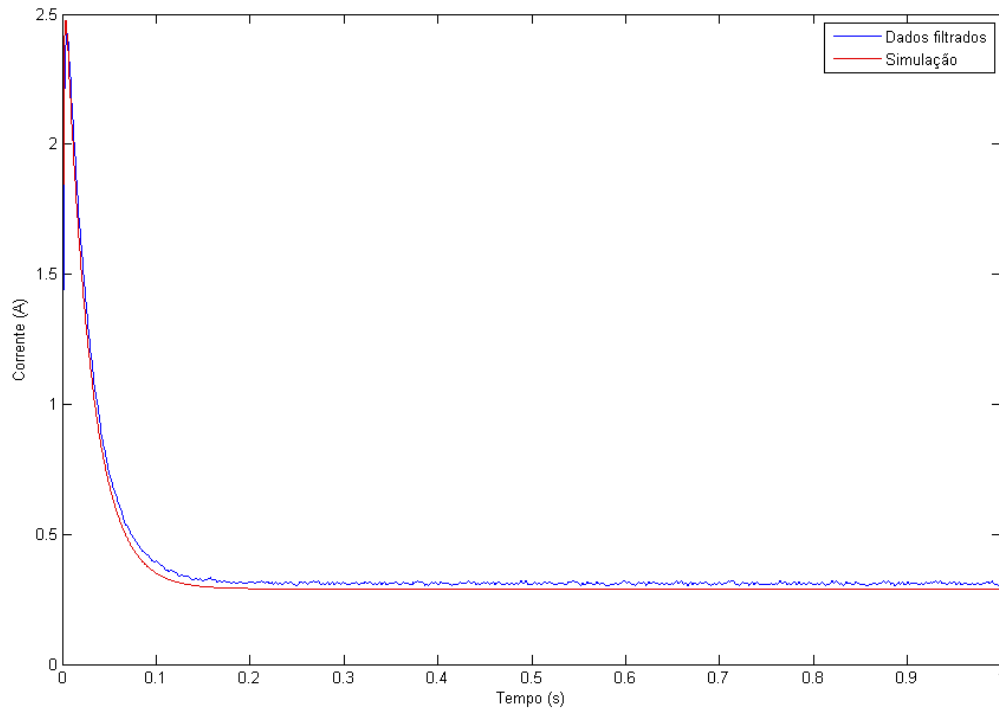


**Figura 3.2 - Resposta ao degrau - Corrente elétrica**

Utilizando o vetor de dados filtrados, e utilizando o comando *ident* do MATLAB, um modelo de primeira ordem foi obtido para o sistema elétrico:

$$G_e(s) = \frac{0,27371}{0,0010302s + 1} \quad (3.2)$$

Para efeito de comparação, a resposta ao degrau deste modelo foi confrontada com a resposta ao degrau real do motor conforme Figura 3.3.



**Figura 3.3 - Comparação entre o modelo levantado e o sistema real – sistema elétrico**

- Sistema mecânico do motor

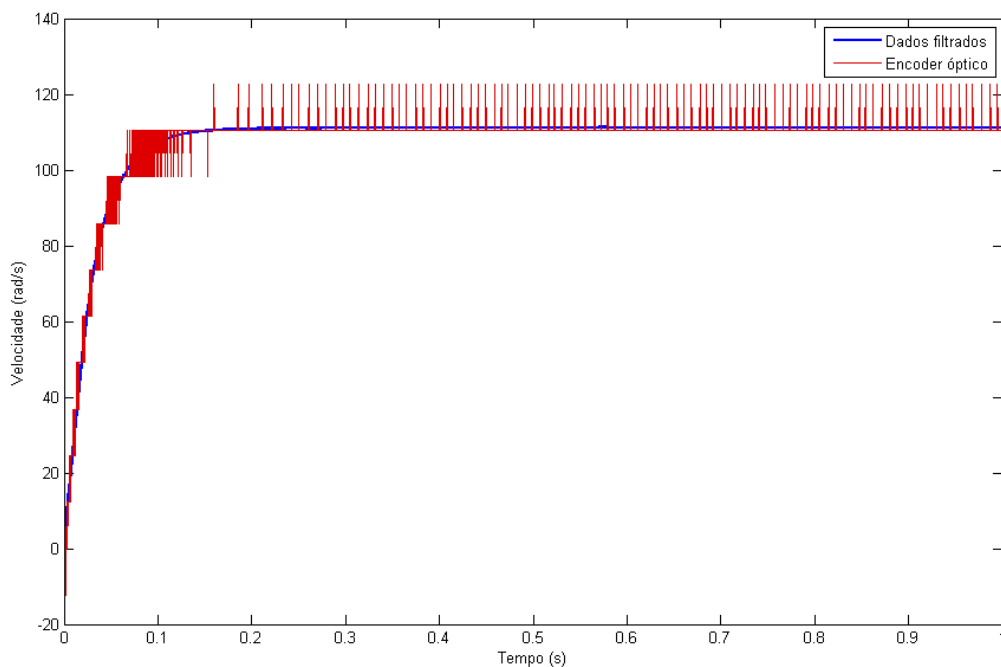
Analogamente ao método utilizado para se levantar o modelo do sistema elétrico do motor, o modelo do sistema mecânico foi obtido a partir do mesmo ensaio ao degrau. Como a velocidade do motor é adquirida a partir da contagem de pulsos do *encoder* ocorrida em intervalos fixos de tempo (1/2000 s), i.e., a velocidade não é adquirida diretamente a partir de um tacogerador, existe um ruído inerente desta conversão. Para eliminar este ruído, aplica-se novamente um filtro de Butterworth de forma que o modelo não o considere, conforme Figura 3.4.

A partir dos dados de velocidade angular filtrados, utiliza-se novamente a ferramenta de identificação de sistemas do MATLAB para encontrar os parâmetros do modelo de primeira ordem:

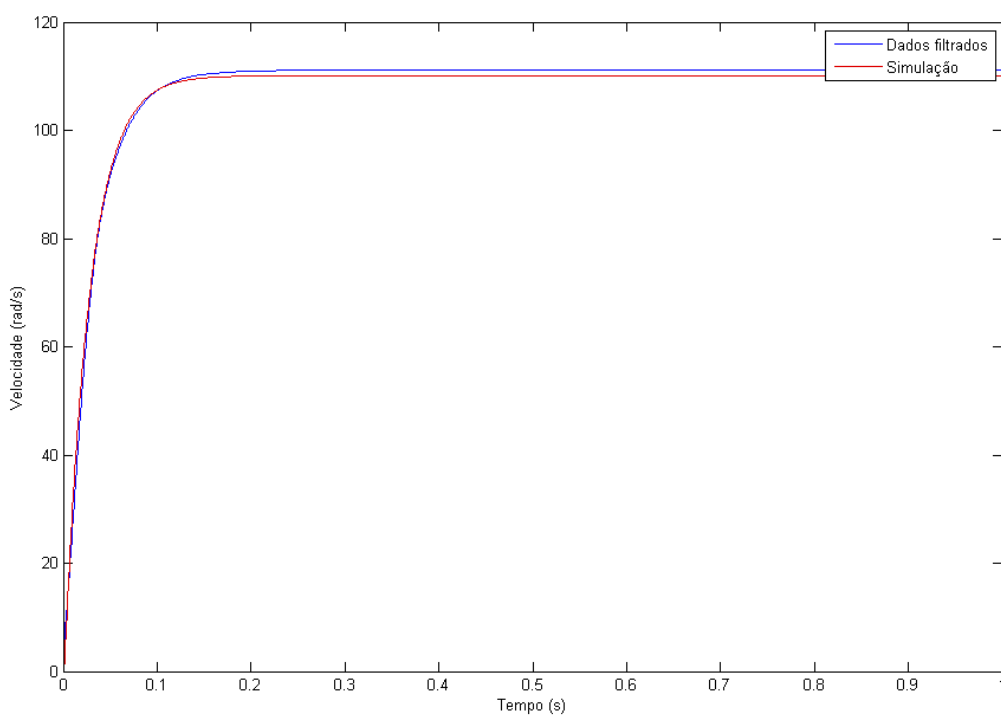
$$G_m(s) = \frac{4467,5}{0,25773s + 1} \quad (3.3)$$

Confrontando a resposta ao degrau deste modelo com a resposta do sistema real, tem-se a Figura 3.5.

Por fim, a partir de uma simples comparação polinomial entre as equações (3.2), (3.3) e as equações dispostas na Figura 2.5, os parâmetros físicos do motor são evidenciados conforme Tabela 3.3.



**Figura 3.4 - Resposta ao degrau - velocidade angular**



**Figura 3.5 – Comparação entre o modelo levantado e o sistema real – sistema mecânico**

**Tabela 3.3 - Parâmetros do motor de corrente contínua Electrocraft**

<b>Parâmetros</b>	<b>Valores</b>
Resistência da armadura ( $\Omega$ )	3,65
Indutância da armadura (mH)	3,76
Momento de inércia ( $\text{kgm}^2$ )	57,69E-06
Constante de atrito viscoso ( $\text{Nm/rad(s)}$ )	223,8E-06
Constante de fcm ( $\text{Vs/rad}$ )	0,085
Constante de torque ( $\text{Nm/A}$ )	0,085

- Função de transferência do motor

A partir das funções de transferência dos sistemas elétrico e mecânico, a função de transferência completa do motor, relacionando a posição angular com tensão de armadura é:

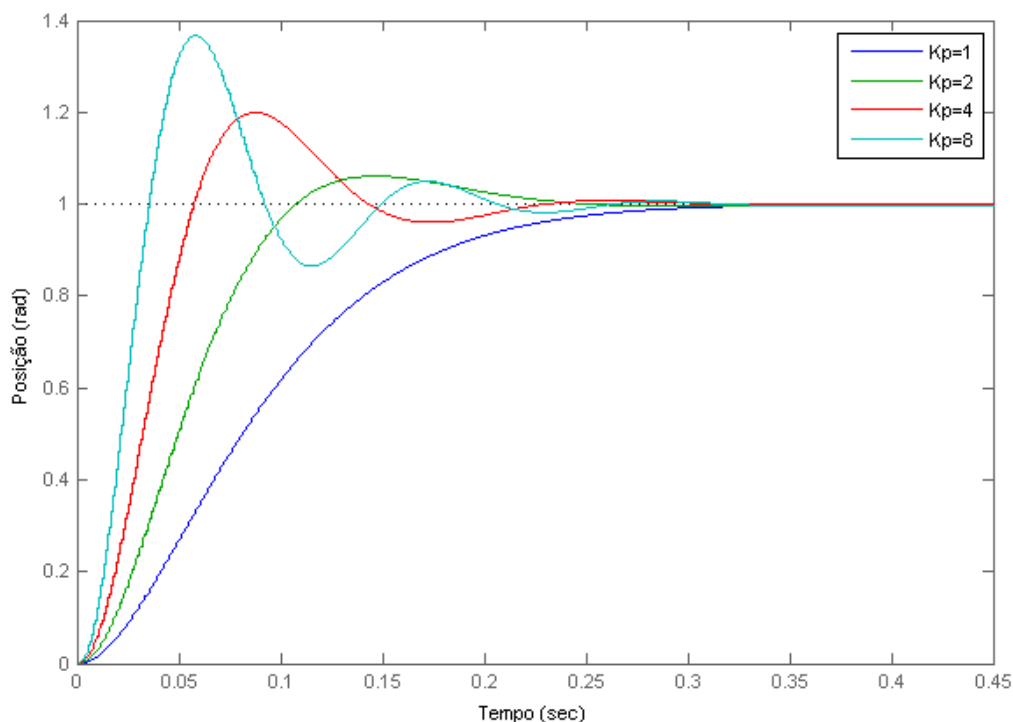
$$\frac{\theta_m(s)}{V_a(s)} = \frac{391460,2}{s(s + 934,9)(s + 39,62)} \quad (3.4)$$

Discretizando esta função de transferência com auxílio do comando *c2d* do MATLAB, para um tempo de amostragem de 1/2000 segundos, tem-se:

$$\frac{\theta_m(z)}{V_a(z)} = \frac{7,248e - 6z^2 + 2,577e - 5z + 5,682e - 6}{z^3 - 2,607z^2 + 2,221z - 0,6143} \quad (3.5)$$

### 3.2. Investigação do controlador proporcional

Sabe-se que um controlador puramente proporcional não zera o erro em regime permanente quando se faz controle de velocidade de motores CC. No caso de controle de posição, este fato não é observado pois a planta possui naturalmente um integrador. A resposta do sistema ideal para vários ganhos do controlador proporcional pode ser vista na Figura 3.6.



**Figura 3.6 - Controlador Proporcional - variação de  $K_p$**

Independente do ganho proporcional  $k_p$  escolhido, tem-se erro de regime permanente igual a zero. Para valores crescentes de  $k_p$ , respostas cada vez mais rápidas podem ser obtidas, com um efeito adverso de aumento de sobressinal. Com isto, quanto maior seja o ganho, mais oscilatória será a resposta do sistema. Isto é válido para um motor CC ideal e sem aplicação de distúrbios. Quando se adicionam distúrbios, o erro de regime deixa de ser nulo e a ação de controle proporcional não será mais capaz de anulá-lo.

Como detalhado na seção 2.2.2, os efeitos não-lineares como atrito de Coulomb e zona morta adicionam distúrbios significativos no sistema quando operado a baixas velocidades. Desta forma, um controlador puramente proporcional não será capaz de atender o requisito básico de controle de anulação do erro de regime. Um método bastante eficaz para se atingir o erro de regime igual a zero é a adição de um termo integral à ação de controle.

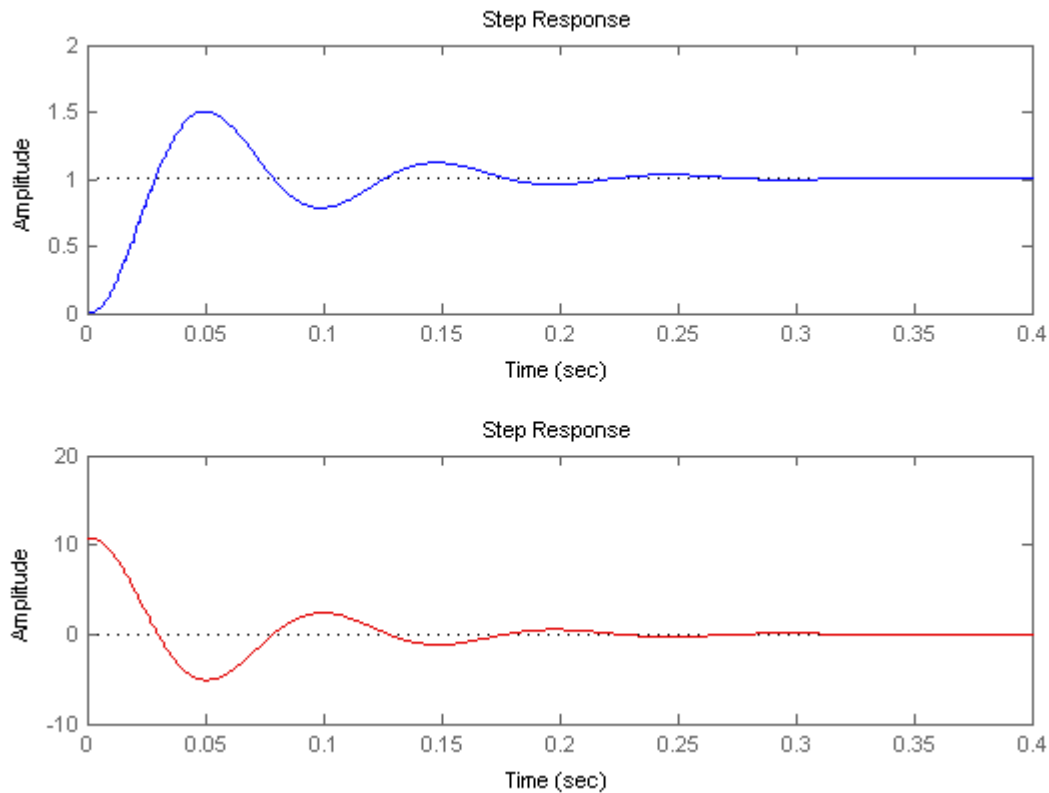
### 3.3. Investigação do controlador PI

O termo integral do controlador PI é capaz de zerar o erro de regime permanente causado pelos efeitos não-lineares intrínsecos à operação do motor CC. Investigando-se um controlador PI com auxílio da ferramenta *rltool*, um controlador foi projetado para que sua

máxima ação de controle fosse exatamente igual à máxima tensão fornecida pelo sistema. Este requisito foi escolhido para que o efeito de saturação do controlador fosse evitado, e a resposta do sistema pudesse ser comparada com o sistema ideal.

Utilizando o método do lugar das raízes, o controlador PI que fornece a resposta mais rápida sem que haja efeito de saturação foi (Figura 3.7):

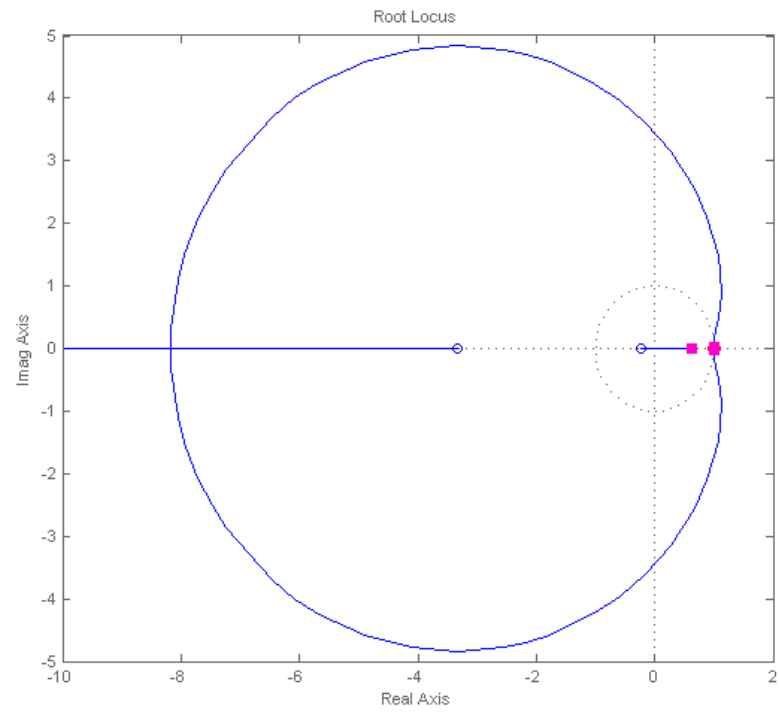
$$C(z) = 10,698 \frac{z - 0,998}{z - 1} \quad (3.6)$$



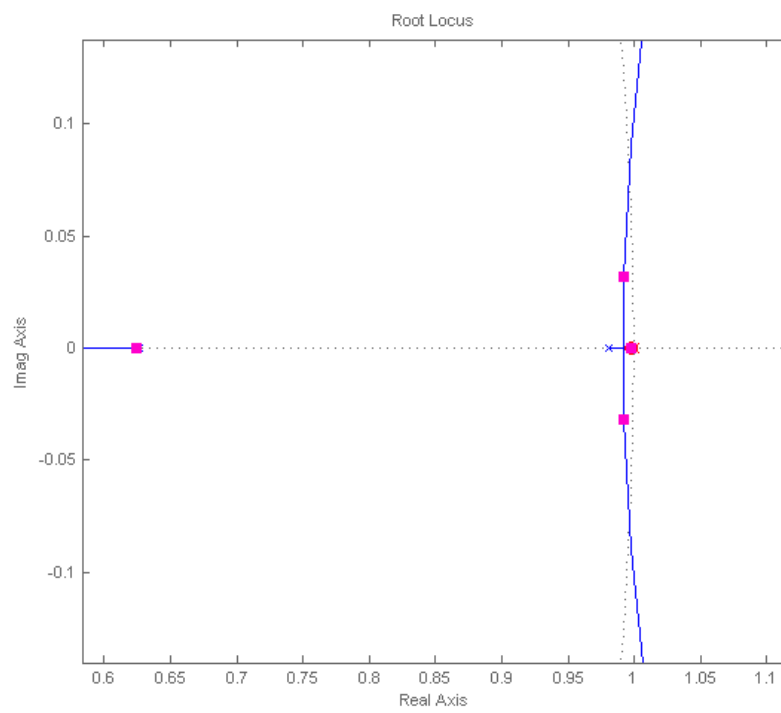
**Figura 3.7 - Resposta ao degrau - Controle PI**

O lugar das raízes pode ser visualizado nas Figura 3.8 e Figura 3.9.

A Figura 3.9 em especial mostra como um controlador PI é difícil de ser projetado para este sistema pois a adição de mais um pólo na origem faz com que o lugar das raízes rapidamente saia do círculo unitário do plano complexo.



**Figura 3.8 - Lugar das raízes**



**Figura 3.9 - Detalhe do lugar das raízes**

Muito embora este controlador não seja o mais indicado devido ao elevado sobressinal causado, é ainda muito mais eficiente que o controlador proporcional, pois sua ação zera efetivamente o erro de regime, independente do distúrbio causado pelos efeitos não-lineares, ou por efeitos externos, como adição de torque devido ao acoplamento mecânico do eixo com uma carga.

Por fim, a resposta deste sistema desconsiderando-se os efeitos não-lineares pode ser observada com o auxílio do *rttool* (Figura 3.7). A curva azul indica a resposta do sistema quando aplicada uma referência em degrau de 1 rad, e a curva verde indica a ação de controle desenvolvida. Nota-se que a máxima ação de controle é aproximadamente 12 V, acontecendo no início da operação, causada principalmente pelo ganho proporcional. À medida que a resposta do sistema se aproxima de seu valor de referência, a ação de controle decresce até que a mesma atinja o valor nulo, no mesmo instante em que o sistema atinge o valor de referência.

### **3.4. Investigação do controlador PID**

O controlador proporcional-integral-derivativo exige uma energia muito grande devido à ação derivativa. Por limitações físicas do sistema, o mesmo não é capaz de fornecer toda a energia requerida pelo compensador, saturando o mesmo e atrasando a resposta em comparação ao que se esperaria se o sistema fosse ideal. Desta forma, o controlador mais indicado para o sistema de posicionamento do motor CC é um controlador PI, que alia a inexistência do erro em regime permanente proporcionado pela ação integral, e a rápida resposta proporcionada pela ação de controle proporcional.

### **3.5. Implementação e resultados**

O controlador PI projetado na seção anterior será testado em dois ensaios. O primeiro, trata-se de uma resposta a um degrau de referência de 1 rad. A simulação para este ensaio, considerando-se um motor CC ideal, pode ser realizada a partir do diagrama Simulink indicado na Figura 3.10.

A implementação deste controlador no sistema físico pode ser realizada com o diagrama do Simulink indicado na Figura 3.11. O sub-sistema intitulado motor está descrito na seção 2.2.3. Este bloco contém todos os arquivos *target* necessários para aquisição dos dados de velocidade do motor, e dos arquivos necessários para envio dos sinais de PWM e direção de rotação do motor.



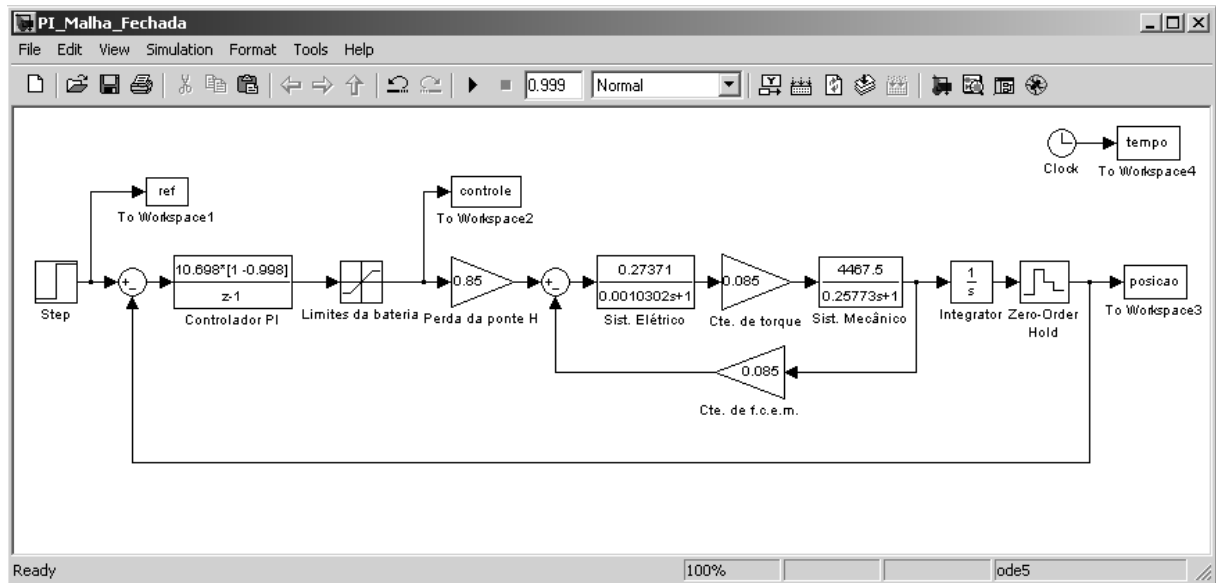


Figura 3.10 - Simulação de controlador PI

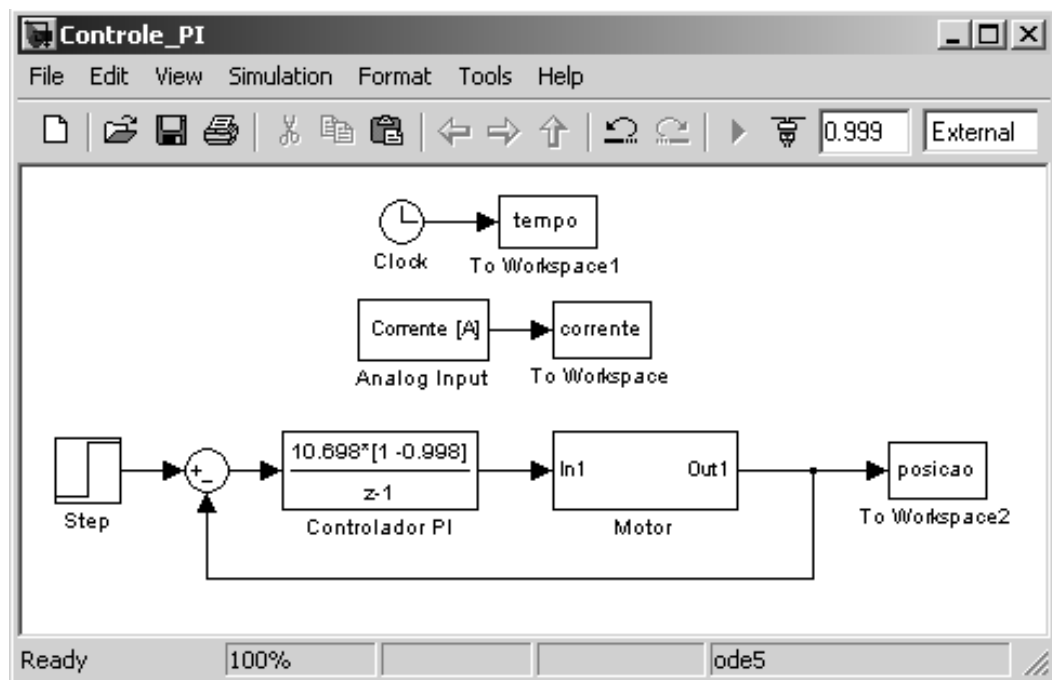
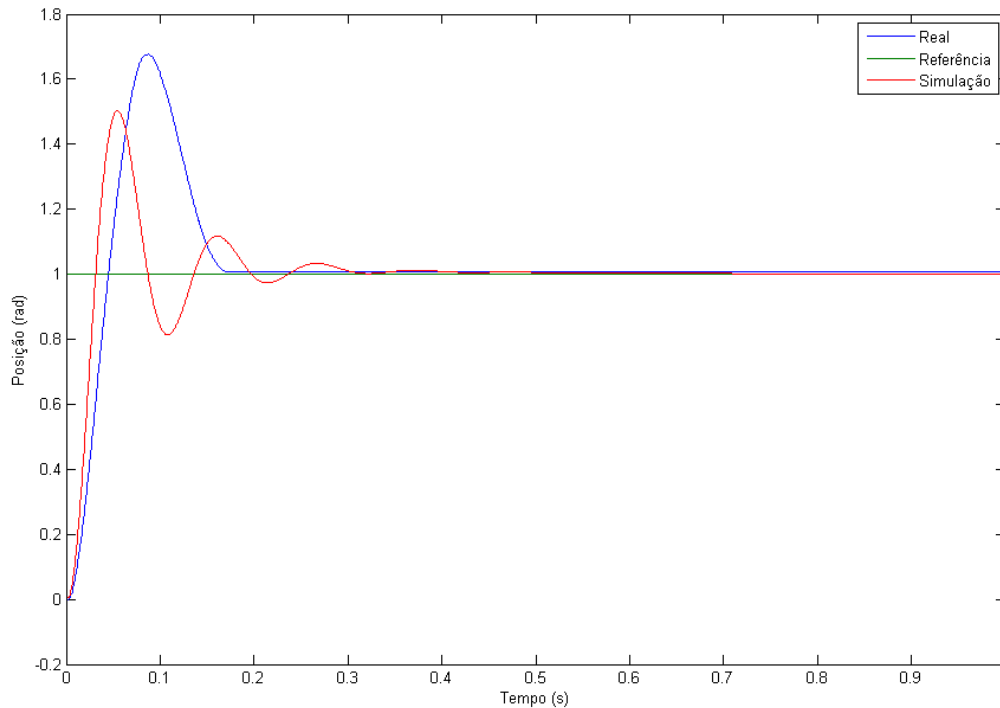


Figura 3.11 - Implementação de controlador PI

Portanto, é possível realizar uma comparação entre a resposta ao degrau de posição do sistema real, incluindo suas não-linearidades, e a resposta ao degrau de posição de um sistema ideal.



**Figura 3.12 - Comparação entre a resposta ao degrau de posição do sistema real e do sistema ideal**

A Figura 3.12 mostra esta comparação. Nota-se que ambas as respostas exibem um erro de regime permanente nulo, porém a dinâmica das respostas apresenta características diferentes. Percebe-se que este sistema apresenta características típicas de sistemas não-lineares.

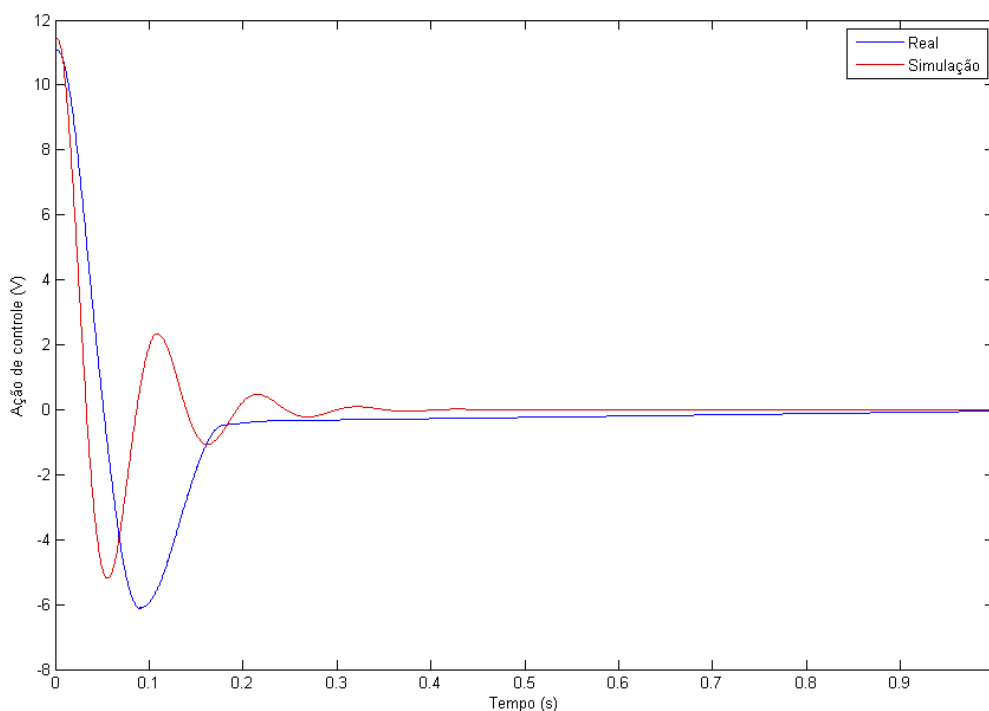
Primeiramente, a resposta atrasada em relação ao sistema teórico pode ser explicada pelo acionamento do motor realizado por meio do L298, conforme efeitos observados na seção 2.5.

Em seguida, é possível notar uma resposta com alto índice de sobressinal, porém com pouco ou nenhum comportamento oscilatório. Esta característica foi induzida por meio dos efeitos da força de atrito de Coulomb, os quais atuam predominantemente quando o motor está operando em baixas velocidades.

Por fim, o erro nulo em regime permanente é proporcionado pela ação integral do controlador para o caso do sistema físico. Sem esta ação de controle, existiria um erro de regime não-nulo ocasionado pelos distúrbios de torque dos fenômenos não-lineares.

A Figura 3.13 indica a ação de controle desenvolvida para ambos os casos. A ação de controle se inicia próxima a 12 V, decaindo consideravelmente à medida que o sistema se aproxima de seu *setpoint*. Devido ao erro de regime permanente nulo, ambas as curvas se aproximam indefinidamente de zero. Novamente, porém, a ação de controle apresenta

diferenças significativas em sua dinâmica devido aos efeitos não-lineares que não foram considerados.

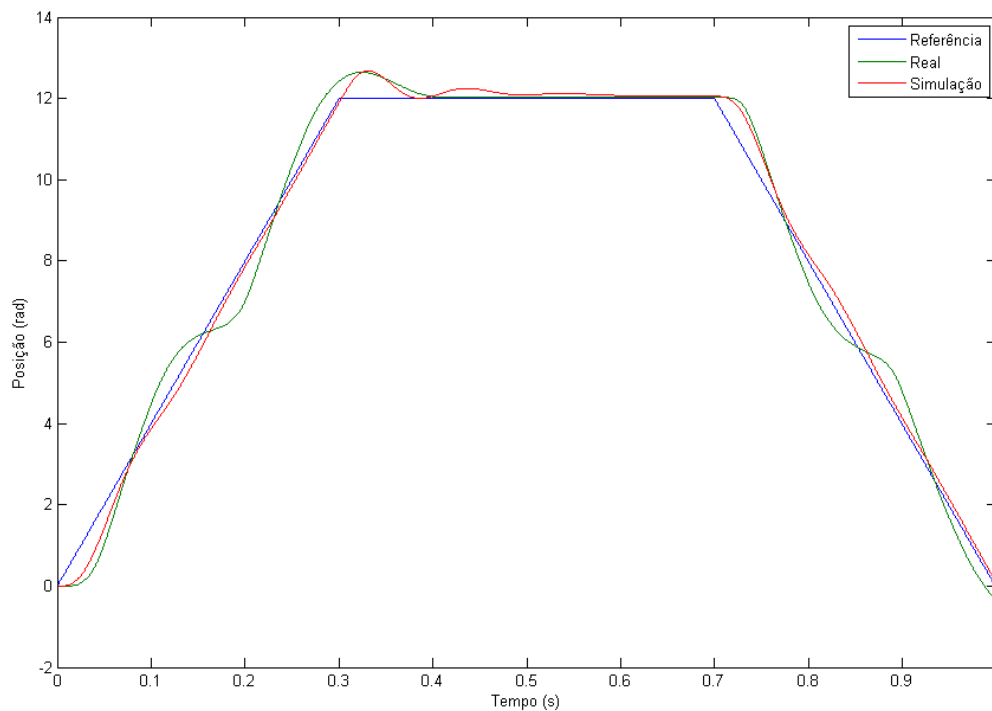


**Figura 3.13 - Comparação entre a ação de controle do sistema real e do sistema ideal**

Um segundo ensaio será realizado para se analisar o comportamento do sistema para uma referência do tipo rampa. A rampa foi implementada em três etapas conforme as características indicadas na Figura 3.14. O diagrama do Simulink usado para este ensaio é o mesmo do utilizado no ensaio anterior, mudando-se somente o bloco *step* por um bloco *repeating sequence*.

A rampa cresce linearmente entre os instantes de 0 a 0,3s, permanece em repouso dos instantes de 0,3 a 0,7s, e decresce linearmente dos instantes de 0,7 a 1s.

A resposta do sistema ideal pôde, por fim, ser comparada à implementação do controle PI no sistema físico. Entre os instantes de 0 a 0,3s, nota-se que a resposta real é significativamente mais oscilatória que a resposta do sistema ideal. Para o patamar entre 0,3 e 0,7s, ambas as respostas atingem um erro de regime permanente nulo, e durante os instantes finais entre 0,7 e 1s, o comportamento oscilatório do sistema real se repete.



**Figura 3.14 - Resposta à rampa**

### 3.6. Discussão

As diferenças observadas entre as respostas esperadas e as obtidas com o sistema real são explicadas pelos fenômenos abordados na seção 2.2.2. Para ações de controle em que não é necessária uma precisão muito grande, estes fenômenos podem ser desprezados. Porém, o controle de posição de motores de corrente contínua envolve regiões de operação extremamente não-lineares, e que a não consideração destes efeitos pode acarretar em diferenças significativas entre projeto e realidade.

As comparações entre simulações e realizações diferem por falta ou deficiência na descrição dos efeitos não lineares verificados no sistema real. Também foram observados comportamentos não lineares no acionamento por meio do L298, o qual não é capaz de fornecer um degrau de tensão devido a um atraso inerente ao seu acionamento.

## 4. Conclusão

O uso de computadores atuando em tempo real no controle de sistemas provê uma grande versatilidade ao projetista. A suíte MATLAB/Simulink/*Real-time Workshop* se mostrou eficiente e a implementação de técnicas de controle se faz de forma natural com esta ferramenta totalmente integrada ao MATLAB.

Foi demonstrado que a correta modelagem de sistemas dinâmicos influencia diretamente na precisão da ação de controle que se deseja alcançar. Para situações em que se deseja controlar a velocidade de motores CC, o modelo linear garante bons resultados devido à operação em sua região linear. Por outro lado, o controle de posição de motores de corrente contínua atua em regiões de operação não-lineares, causadas principalmente pelas forças de atrito de Coulomb. Neste caso, a negligência destes efeitos pode afetar o desempenho do sistema de controle.

Outro fato importante é que o *hardware* de acionamento de motores de corrente contínua pode inserir outra gama de efeitos não-lineares. Desta forma, a escolha correta de seus componentes é vital para o bom funcionamento do sistema.

Trabalhos futuros podem ser realizados na identificação e modelagem dos efeitos não-lineares presentes em motores de corrente contínua. A partir da identificação destes parâmetros, podem ser investigadas técnicas de controle não-linear, adaptativo e robusto para alcançar desempenhos mais satisfatórios.

Por fim, o controle em tempo real se mostrou muito eficiente, permitindo ao projetista implementar protótipos de controladores rapidamente, e a projetá-los iterativamente. As técnicas aqui apresentadas podem ser utilizadas no ensino de práticas de controle para cursos de Engenharia, tornando possível realizar projetos flexíveis sem o uso de programação.



## Referências Bibliográficas

- AGUIAR, M. L. **Material de ensino da disciplina SEL0359 - Controle Digital**. Universidade de São Paulo. São Carlos. 2011.
- AGUIAR, M. L.; OLIVEIRA, V. A.; VARGAS, J. B. **Sistemas de Controle - Aulas de Laboratório**. 1. ed. São Carlos: Universidade de São Paulo, v. 1, 2005. 308 p.
- ANON. Computing Control - A Commercial Reality. **Control Engineering**, v. 9, n. 5, p. 40, 1959.
- BENNETT, S. **Real-Time Computer Control: An Introduction**. 2nd. ed. New York: Prentice Hall, 1994. 417 p.
- DORF, R. C.; BISHOP, R. H. **Sistemas de controle modernos**. 8. ed. Rio de Janeiro: LTC, v. 1, 2001.
- FAIRCHILD SEMICONDUCTOR CORPORATION. **DM74LS74A Dual Positive-Edge-Triggered D Flip-Flops with Preset, Clear and Complementary Outputs - Datasheet**. [S.l.]. 2000.
- FAIRCHILD SEMICONDUCTOR CORPORATION. **DM74LS86 Quad 2-Input Exclusive-OR Gate - Datasheet**. [S.l.]. 2000.
- FAIRCHILD SEMICONDUCTOR CORPORATION. **TIL111 phototransistor optoisolator - Datasheet**. [S.l.]. 2000.
- FRANKLIN, G. F.; POWELL, J. D.; EMAMI-NAEINI, A. **Feedback Control of Dynamic Systems**. 4. ed. Upper Saddle River: Prentice-Hall, 2002.
- GUERRA, R. S.; GONÇALVES JÚNIOR, H. J.; LAGES, W. F. **A digital PID controller using RTAI**, Porto Alegre, 2002.
- GUIMARÃES, D. V. **Kit didático para o aprendizado de controladores em tempo real utilizando MATLAB e toolbox RealTime Workshop**. Universidade de São Paulo. São Carlos. 2011.
- KARA, T.; EKER, I. Energy Conversion and Management. **Nonlinear modeling and identification of a DC motor for bidirectional operation with real time experiments**, Gaziantep, 2003.
- NATIONAL INSTRUMENTS CORPORATION. **AT-MIO/AI E Series User Manual**. Austin. 1996.
- NEWTON, A. C. M. **Experimentos de controle digital em tempo real com auxílio do MATLAB e SIMULINK**. Universidade de São Paulo. São Carlos. 2009.
- OGATA, K. **Engenharia de controle moderno**. 1. ed. Rio de Janeiro: Prentice Hall do Brasil, v. 1, 1985.

QUARANTA, G.; MANTEGAZZA, P. **Using MATLAB-SIMULINK RTW to build real time control applications in user space with RTAI-LXRT**, Milão, 2001.

STMICROELECTRONICS. **L298 dual full bridge driver - Datasheet**. [S.l.]. 2000.

THE MATHWORKS, INC. **Real-Time Workshop: For Use with Simulink**. Natick. 1999.

VILLAÇA, M. V. M. **Apostila de sistemas de controle**. Centro Federal de Educação Tecnológica de Santa Catarina. Florianópolis. 2007.

WILLIAMS, T. J. Two decades of change. **Control Engineering**, v. 24, n. 9, 1977.



## Apêndice A – Diagrama esquemático do projeto

