

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS DE CURITIBA
CURSO DE ENGENHARIA INDUSTRIAL ELÉTRICA**

ANA LÚCIA VIEIRA IAREMCZUK

JULIO CESAR MOURA FAÉ

**OBTENÇÃO DO MODELO MATEMÁTICO DE UM MOTOR BLDC
UTILIZANDO MÉTODOS DE IDENTIFICAÇÃO COM DISPOSITIVO
DSC**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2014

ANA LÚCIA VIEIRA IAREMCZUK

JULIO CESAR MOURA FAÉ

**OBTENÇÃO DO MODELO MATEMÁTICO DE UM MOTOR BLDC
UTILIZANDO MÉTODOS DE IDENTIFICAÇÃO COM DISPOSITIVO
DSC**

Proposta de Trabalho de Conclusão de Curso de Graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 2, do curso de Engenharia Industrial Elétrica da Universidade Tecnológica Federal do Paraná (UTFPR), Campus Curitiba, como requisito parcial para obtenção do título de Engenheiro Eletricista.

Orientador: Prof. Dr. Amauri Amorim Assef

Coorientador: Prof. Me. Guilherme Luiz Moritz

CURITIBA

2014

RESUMO

IAREMCZUK, Ana L.; FAÉ, Julio C. Obtenção do modelo matemático de um motor BLDC utilizando métodos de identificação com dispositivo DSC. 2014. 72 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Industrial Elétrica. Universidade Tecnológica Federal do Paraná, Curitiba, 2014.

Em muitas situações práticas, deseja-se interagir com os sistemas ao redor, seja para o conhecimento, para a análise ou para o controle. Para que isso seja possível, é necessário obter um modelo que represente as características destes sistemas através de técnicas de modelagem. No caso do motor BLDC, uma resposta desejada poderia ser alcançada através do projeto de um controlador. Porém, para isso, é necessário possuir um modelo matemático que represente a dinâmica do sistema. A obtenção do modelo poderia ser realizada a partir de um estudo detalhado, equacionamento físico e obtenção dos parâmetros referentes à sua estrutura, processo denominado modelagem pela física, o que poderia ser inviável pelo tempo disponível ou equipamentos de medição necessários. Alternativamente, existem as técnicas de identificação, em que são necessários apenas dados de entrada e saída do sistema. Este trabalho avalia métodos de identificação determinísticos e estocásticos através da obtenção de um modelo para um motor BLDC, apresentando os resultados e as conclusões alcançadas.

Palavras-chave: BLDC, DSC, Modelagem, Identificação, PRBS, Resposta ao degrau, Determinístico, Estocástico.

ABSTRACT

IAREMCZUK, Ana L.; FAÉ, Julio C. Obtenção do modelo matemático de um motor BLDC utilizando métodos de identificação com dispositivo DSC. 2014. 72 f. Trabalho de Conclusão de Curso (Graduação) – Curso de Engenharia Industrial Elétrica. Universidade Tecnológica Federal do Paraná, Curitiba, 2014.

In many practical situations, it may be wanted to interact with the systems around for the knowledge, analysis or control. To make it possible, it is needed to obtain a model that represents the characteristics of these systems by modeling techniques. In the case of the BLDC motor, a desired response could be achieved by the design of a controller. But, for this, it is necessary to have a mathematical model that represents the dynamics of the system. The mathematical model can be built from a detailed study of the system, finding the physical equations and obtaining parameters related to its structure, which can be infeasible because of the time available or measuring equipment needed. Alternatively, there are some identification techniques in which only input and output of the system are required. This study evaluates deterministic and stochastic identification methods by obtaining models for a BLDC motor, presenting the results and conclusions reached with the experiments and analysis.

Palavras-chave: BLDC, DSC, Modeling, Identification, PRBS, Step response, Deterministic, Stochastic.

LISTA DE FIGURAS

Figura 1 - Vista de corte de um motor CC com ímãs permanentes.....	12
Figura 2 - Vista de corte do motor CC sem escovas	12
Figura 3 - Plataforma utilizada.....	14
Figura 4 - Estrutura de acionamento	22
Figura 5 – Conjunto inversor	23
Figura 6 – Sinais provenientes do sensor Hall e dos sinais de acionamento da ponte	24
Figura 7 – Driver utilizado no acionamento da ponte inversora	25
Figura 8 – Representação da disposição dos sensores Hall no rotor	26
Figura 9 – Sequência de fase em função dos sensores.....	27
Figura 10 – Submódulo do PWM	29
Figura 11– Circuito equivalente de um motor BLDC em Y	34
Figura 12 – Etapas de um processo de identificação.....	36
Figura 13 – Gráfico para determinação de parâmetros do modelo de resposta ao degrau a dois parâmetros	40
Figura 14 - Gráfico para determinação de parâmetros do modelo de resposta ao degrau a três parâmetros	42
Figura 15 – Sistema de acionamento e leitura de velocidade do motor	48
Figura 16 - Análise Estática do Sistema (Rotação X Tensão de entrada).....	49
Figura 17 - Resposta ao degrau de 17,32 V.....	51
Figura 18 - Curva real de resposta ao degrau (vermelho) e resposta da função de transferência utilizando constante de tempo A-C (azul)	53
Figura 19 - Curva real de resposta ao degrau (vermelho) e resposta da função de transferência utilizando constante de tempo A-B (azul)	54
Figura 20 - Identificação com timer em modo de disparo.....	55
Figura 21 - Resposta ao degrau com variação de 46% a 78% de razão cíclica e ajuste de offset	56
Figura 22 - Parâmetros do controlador PID a partir do método de resposta ao degrau de Ziegler-Nichols	56
Figura 23 - Resposta do sistema em malha fechada com valor de referência de 3500 rpm	57
Figura 24 - Autocorrelação da sequência binária	59
Figura 25 - Periodograma dos sinais de saída (y1) e entrada (u1)	59
Figura 26 - Sinal de saída (y1) e sinal de entrada (u1)	60
Figura 27 - Análise dos resíduos.....	62
Figura 28 - Polos e zeros do sistema	62
Figura 29 - Resposta do sistema e do modelo a uma entrada PRBS	63
Figura 30 - Resposta a um degrau de 12,4 V.....	63
Figura 31 - Autocorrelação do ruído.....	64
Figura 32 - Sinal original	65

Figura 33 - Sinal com ruído aditivo.....	65
Figura 34 - Resposta ao degrau com ruído aditivo.....	66
Figura 35 - Resposta real e do modelo a um degrau de 12,4 V.....	67

LISTA DE SIGLAS

ADC	<i>Analog to digital converter</i> – Conversor analógico digital
ARMA	<i>Autoregressive moving average</i> – Auto regressivo de média móvel
ARMAX	<i>Autoregressive moving average with exogenous</i> – Auto regressivo de média móvel exógeno
ARX	<i>Autoregressive exogenous</i> - Auto regressivo exógeno
BIT	<i>Binary digit</i> – Dígito Binário
BLDC	<i>Brushless direct current</i> – Sem escovas de corrente contínua
CC	Corrente contínua
CA	Corrente alternada
DSC	<i>Digital signal controller</i> – Controlador de sinal digital
DSP	<i>Digital signal processor</i> – Processador de sinal digital
FMM	Força magnetomotriz
I ² C	<i>Inter Integrated Circuit</i> - Circuito Inter Integrado
IEEE	<i>Institute of Electrical and Electronics Engineers</i> - Instituto de Engenheiros Elétricistas e Eletrônicos
MLP	Modulação em largura de pulso
MOSFET	<i>Metal oxide semiconductor field effect transistor</i> - Transistor de efeito de campo metal - óxido – semicondutor
PGA	<i>Programmable gain amplifier</i> – Amplificador de ganho programável
PID	(Controlador) Proporcional-Integral-Derivativo
PRBS	<i>Pseudo random binary sequence</i> – Sequência binária pseudo randômica
PWM	<i>Pulse width modulation</i> – Modulação por largura de pulso
SCI	<i>Serial communications interface</i> – Interface de comunicação serial
SPI	<i>Serial peripheral interface</i> - Interface periférica serial
VSI	<i>Voltage source inverter</i> - Inversor fonte de tensão

LISTA DE SÍMBOLOS

A_0	Parâmetro obtido graficamente para modelagem a dois parâmetros
T_{ar}	<i>Average Residence Time</i>
$v(k)$	Variável aleatória que representa o ruído
$A(q)$	Polinômio que multiplica a saída em modelos de erro na equação
$B(q)$	Numerador da função de transferência do processo
$C(q)$	Numerador da função de transferência do ruído
$D(q)$	Denominador da função de transferência do ruído
$F(q)$	Denominador da função de transferência do processo
$G(q)$	Função de transferência do ruído
$G(s)$	Função de transferência do sistema
$H(q)$	Função de transferência do processo
K	Ganho estático do sistema
L	Tempo morto
T	Constante de tempo
$s(t)$	Resposta ao degrau
$u(k)$	Entrada do sistema em tempo discreto
$y(k)$	Saída do sistema em tempo discreto
α	Parâmetro obtido graficamente para modelagem a dois parâmetros

SUMÁRIO

1. INTRODUÇÃO	11
1.1. TEMA	11
1.1.1. Delimitação do Tema	14
1.2. PROBLEMAS E PREMISSAS	15
1.3. OBJETIVOS	15
1.3.1. Objetivo geral	15
1.3.2. Objetivos específicos	15
1.4. JUSTIFICATIVA	16
1.5. PROCEDIMENTOS METODOLÓGICOS	17
1.6. ESTRUTURA DO TRABALHO	18
2. CARACTERÍSTICAS DO BLDC	20
2.1. ROTOR	20
2.2. ESTATOR	20
2.3. ACIONAMENTO ELÉTRICO	21
2.3.1. Driver	25
2.3.2. Sensor Hall	25
3. CONTROLADOR DA PLATAFORMA	28
3.1. MÓDULO DE PWM	28
3.2. SPI	30
3.3. SCI	30
3.4. PONTO FLUTUANTE	30
4. MODELAGEM APLICADA A UM MOTOR BLDC	32
4.1. TIPOS DE MODELOS	32
4.1.1. Modelos estáticos e dinâmicos	32
4.1.2. Modelos discretos e contínuos	33
4.1.3. Modelos determinísticos e estocásticos	33
4.2. IDENTIFICAÇÃO	34
4.2.1. Método determinístico – resposta ao degrau	38
4.2.1.1. Modelos a dois parâmetros	39
4.2.1.2. Modelos a três parâmetros	41
4.2.1.3. Etapas de identificação do BLDC utilizando métodos determinísticos	43

4.2.2. Método estocástico	43
4.2.2.1. Modelos ARX	44
4.2.2.2. Modelos ARMAX	45
4.2.2.3. Modelos ARMA	45
4.2.2.4. Modelos de erro na saída	46
4.2.2.5. Modelo Box-Jenkins	46
4.2.2.6. Determinação do grau dos polinômios	46
4.2.2.7. Etapas de identificação do BLDC utilizando métodos estocásticos	47
5. RESULTADOS	48
5.1. PREPARAÇÃO DO SISTEMA	48
5.2. PROCESSO DE IDENTIFICAÇÃO	49
5.2.1. Método Determinístico	49
5.2.1.1. Modelos a dois parâmetros	51
5.2.1.2. Modelos a três parâmetros	52
5.2.2. Método Estocástico	57
5.2.2.1. Modelagem no MATLAB	60
6. CONSIDERAÇÕES FINAIS	68
REFERÊNCIAS	70
APÊNDICE A – TUTORIAL	72

1. INTRODUÇÃO

1.1. TEMA

Máquinas elétricas dinâmicas são amplamente utilizadas nos dias atuais, nas mais diversas aplicações. A escolha da máquina para cada uma delas deve ser realizada com base nas características da carga, dimensões físicas e de acordo com sua eficiência. Estes fatores são definidos pelos aspectos construtivos de cada motor.

Dentre os vários tipos existentes, os motores de corrente contínua convencionais são vastamente utilizados. Possuem boa resposta para conjugados variados, boa eficiência e o controle sobre eles é simples. Porém, possuem inconvenientes devidos à necessidade de um conjunto composto de comutador e escovas, que está sujeito a desgastes e acaba resultando em uma elevada taxa de manutenção (MONTEIRO, 2006).

Também muito utilizado, o motor à indução com rotor curto-circuitado (gaiola de esquilo) possui características construtivas simples, seu custo é muito inferior em comparação com os motores CC de mesma potência, são mais robustos e há facilidade na manutenção. Todavia, dependendo das características da carga utilizada, há a necessidade de utilizar um inversor de frequência, que possui custo elevado (PIRES, 2006).

Outro motor com baixa complexidade construtiva é o motor síncrono de ímãs permanentes sem escovas. Este possui alto grau de eficiência, que o enquadra em aplicações que demandem alto desempenho do acionamento e em sistemas de posicionamento (YADAMALE, 2003).

Motores síncronos de ímãs permanentes sem escovas são muito semelhantes aos motores CC convencionais. No entanto, seu acionamento é feito de forma diferente da comutação entre coletor e escovas presente nestes. Sem utilizar esse mecanismo, os motores sem escovas possuem um acionamento derivado da comutação eletrônica em seu estator, o que elimina os problemas de desgastes nas escovas e elimina as perdas provenientes destas.

Dentro deste grupo de motores, encontra-se o BLDC, que é o objeto de estudo do presente trabalho. Nas Figura 1 e Figura 2, pode ser visualizada a distinção entre as duas arquiteturas citadas anteriormente.

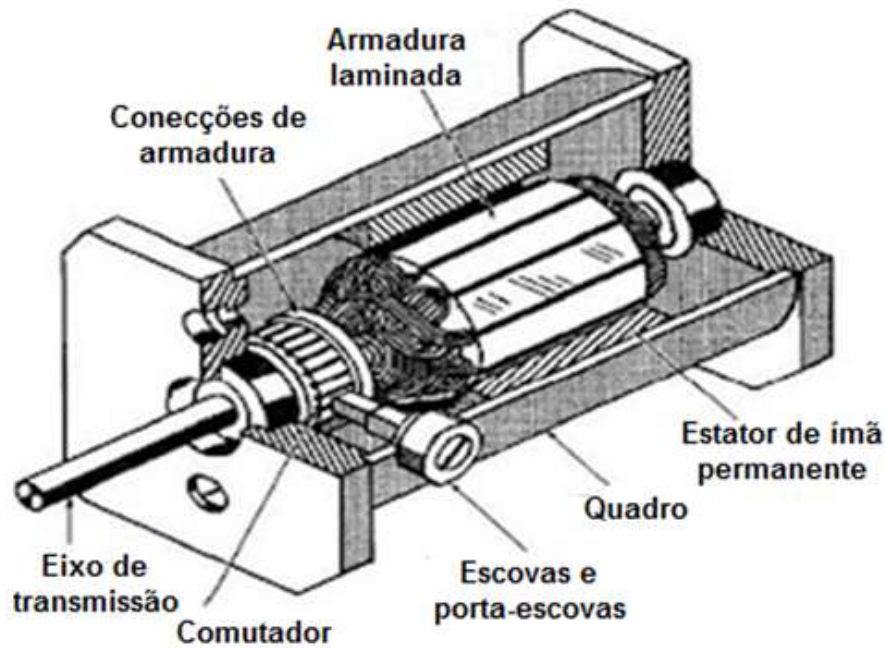


Figura 1 - Vista de corte de um motor CC com ímãs permanentes
 Fonte: Adaptado de: SCLATER & CHIRONIS (2001)

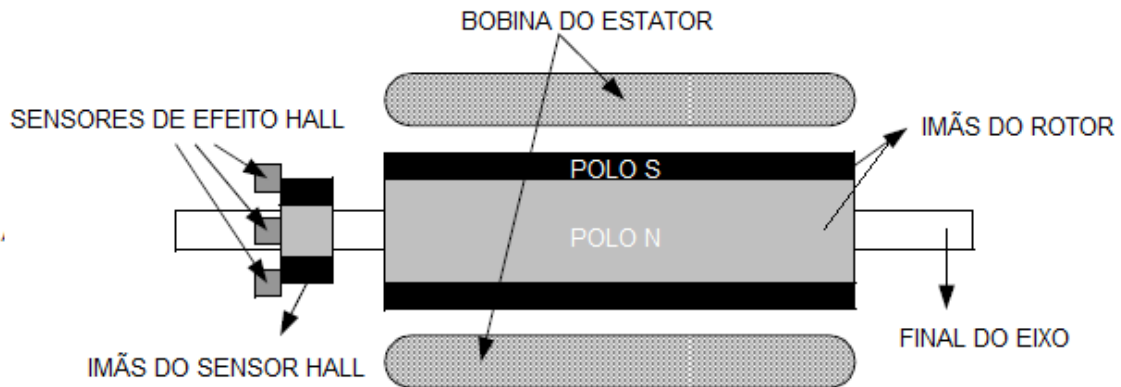


Figura 2 - Vista de corte do motor CC sem escovas
 Fonte: Adaptado de: YADAMALE (2003)

A utilização dos motores elétricos pode demandar a obtenção de características específicas de operação. A máquina pode estar operando sobre cargas sensíveis a oscilações de torque ou velocidade, ou ainda estar atuando em um posicionamento que requer elevado grau de precisão. Além disso, pode-se desejar aprimorar a eficiência da utilização do motor, fazendo-o trabalhar em uma região de operação em que são obtidos os melhores rendimentos.

Os fatores supracitados podem ser obtidos através da utilização de uma malha de controle adequada.

Existem algoritmos de controle que independem do conhecimento aprofundado do sistema, sendo elaborados diretamente a partir da observação da curva de resposta do sistema a um determinado sinal de excitação. Em contrapartida, há uma variedade de controladores que podem ser projetados a partir de um modelo criado para este sistema.

Modelos matemáticos descrevem a dinâmica dos mais variados sistemas através de relações matemáticas que servem para entender e explicar fenômenos observados, projetar sistemas de monitorização e controle, predição, estimação de estados, simulação e treinamento (AGUIRRE, 2007, p.52). A área do conhecimento que estuda o levantamento de tais modelos de sistemas reais é denominada modelagem matemática.

A forma de construção de um modelo pode ser classificada em categorias que levam em conta o conhecimento prévio sobre o sistema em estudo e as leis físicas que regem seu comportamento ou informações auxiliares que suportem a criação do modelo.

Porém, muitas vezes não se dispõe de informação suficiente, como é o caso do motor BLDC utilizado neste trabalho, ou de tempo hábil para realizar o equacionamento dos fenômenos envolvidos. Desta forma, utiliza-se alternativamente um método experimental da área de modelagem: a identificação.

Na identificação, um modelo matemático para um sistema real é desenvolvido a partir de uma relação entre os dados obtidos na saída para uma entrada aplicada. Sabe-se que estes valores estão corrompidos por ruídos de diversas origens. Há métodos de identificação que dão tratamento especial ao ruído, enquanto outros não. Tratam-se dos Métodos Estocásticos e dos Métodos Determinísticos, respectivamente, que serão abordados em mais detalhes ao longo do desenvolvimento.

1.1.1. Delimitação do Tema

No presente trabalho, serão descritos todos os procedimentos utilizados para obter um modelo que melhor caracterize a dinâmica de um motor BLDC, com o uso dos métodos descritos. Para isso, será utilizado um sistema de desenvolvimento com o Controlador de Sinal Digital (DSC) da empresa Freescale para o controle de um motor BLDC de pequeno porte. O sistema utilizado está representado através da Figura 3.

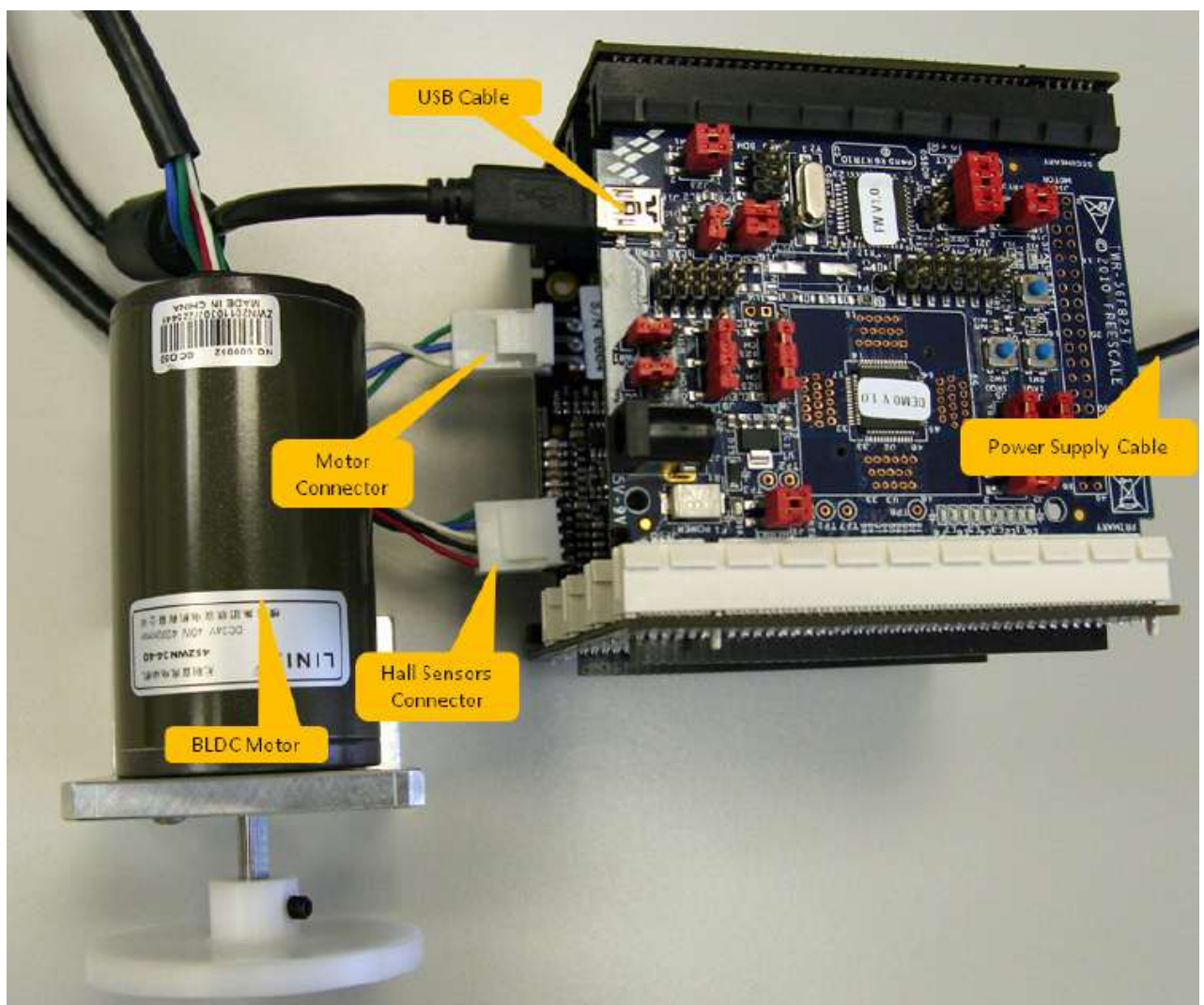


Figura 3 - Plataforma utilizada

Fonte: Adaptado de: FREESCALE (2011, P.4)

1.2. PROBLEMAS E PREMISSAS

Em muitas situações práticas, deseja-se interagir com os sistemas ao redor, seja para o conhecimento, para a análise ou para o controle. Para que isso seja possível, é necessário possuir um modelo que represente as características destes sistemas.

Este modelo matemático do sistema real pode ser determinado de diversas maneiras, processo denominado modelagem. Porém, alguns exigem conhecimentos prévios sobre o sistema, ou demandam um tempo que não está disponível.

Esta é uma problemática muito presente nas indústrias, onde grande parte dos processos não disponibiliza conhecimento suficiente da planta para criar o modelo, o sistema pode estar inacessível ou é excessivamente complexo.

Porém, existe um método alternativo. De forma geral, é possível agrupar as técnicas de modelagem em duas grandes categorias, a saber: modelagem pela física ou identificação. Esta permite levantar modelos de sistemas com detalhamento incompleto, ou até mesmo sem nenhum, a respeito da planta em estudo.

Nesse trabalho, serão definidos métodos para a identificação não intrusiva de um modelo para o motor BLDC, com o intuito de que seja possível estabelecer um controle efetivo para o sistema.

1.3. OBJETIVOS

1.3.1. Objetivo geral

Obter uma representação da dinâmica de um motor BLDC por um modelo matemático, que se aproxime o quanto possível do sistema real, através de variadas técnicas de identificação.

1.3.2. Objetivos específicos

O presente trabalho tem como objetivos específicos:

- Apresentar a estrutura e o acionamento do motor BLDC;
- Descrever a arquitetura do DSC da Freescale;
- Aplicar os métodos (determinístico e estocástico) de identificação de sistemas descritos nas principais literaturas da área e mais amplamente utilizados;
- Selecionar e justificar, com base em pesquisa, quais os sinais de excitação a serem aplicados ao sistema;
- Realizar e descrever ensaios para a obtenção das informações necessárias aos métodos de identificação;
- Explorar ferramentas computacionais que suportem a obtenção e avaliação de modelos de sistemas;
- Validar os modelos obtidos a partir de análises propostas pelos autores de referência;
- Verificar, dentro de cada método, o modelo que melhor representou o sistema real;
- Descrever a comparação entre modelos obtidos a partir de identificações de complexidade distinta;
- Apresentar os resultados e conclusões extraídas do desenvolvimento do trabalho.

1.4. JUSTIFICATIVA

Devido às características de estrutura e funcionamento, o motor BLDC possui aspectos particulares ideais para determinadas aplicações. Normalmente está vinculado a sistemas que exigem elevado grau de dinamismo, alta eficiência, longa vida útil, operação silenciosa e que trabalhem com maiores faixas de velocidade.

Nestes sistemas, devem-se empregar controles que atendam à necessidade da carga, com boa resposta às suas variações, justificando sua aplicação. Desta maneira, o estudo de formas eficientes de controle e acionamento gera interesse no aspecto de que se almeja obter o maior rendimento possível e que haja satisfação das demandas de cada aplicação.

Os aspectos citados anteriormente justificam a empregabilidade deste tipo de motor em diversas aplicações na indústria nas mais diversas áreas como automobilística, aeroespacial, médica, instrumentação e indústrias de automação (YADAMALE, 2003), o que incita a sua utilização como objeto de estudo deste trabalho.

Uma resposta satisfatória do conjunto pode ser obtida empregando um controle bem elaborado, que é a garantia da obtenção dos requisitos estabelecidos.

Alguns controladores podem ser projetados sem considerar o modelo matemático do processo, outros necessitam desta representação. Nestes casos, necessitam-se métodos de identificação eficientes para estabelecer o modelo e assim elaborar o controle desejado.

Há técnicas de identificação que possuem uma teoria envolvida complexa. Porém, com a disposição de ferramentas computacionais, o tempo necessário pode ser bastante reduzido.

Estas técnicas de identificação são empregadas, por exemplo, em alguns inversores de frequência da empresa WEG que possuem controle vetorial. Nestes casos há necessidade de se conhecer os parâmetros do motor (WEG, 2004), no entanto, estas informações de como são obtidos os parâmetros são segredos industriais, onde o estudo dos métodos utilizados se torna um atrativo, acrescentando de forma substancial em termos de conhecimento pelo seu estudo.

1.5. PROCEDIMENTOS METODOLÓGICOS

Para o desenvolvimento do trabalho, foi realizada inicialmente uma revisão bibliográfica com o intuito de contemplar todas as áreas abordadas, através de livros, artigos técnicos, teses de mestrado e doutorado, e elaborada a fundamentação teórica necessária ao trabalho.

Estes estudos abordam os tipos estruturais dos motores dando ênfase no BLDC, e também nos métodos de acionamento deste, sendo analisada sua aplicabilidade de acordo com suas características.

Da mesma forma, técnicas de modelagem e identificação de sistemas, formas de obtenção de parâmetros para os mais variados tipos de modelos, sinais de excitação e sua influência na identificação.

Além disso, o conjunto necessário para o acionamento do motor abrangendo a estrutura de potência formada por ponte inversora, *drivers* de acionamento, sensores de posição, corrente e tensão também serão estudadas e detalhadas.

Outra composição de pesquisa essencial é a estrutura que foi utilizada para o processamento, o DSC, onde sua arquitetura, memória, interrupções e periféricos, foram verificados para a análise da implementação do acionamento e controle.

Foi elaborado um *firmware* utilizando o *software* CodeWarrior, implementando toda a lógica necessária para o acionamento do motor, aplicação de sinais de excitação e aquisição dos dados necessários para cada modelo de identificação selecionado.

Dois sinais de excitação diferentes foram aplicados ao motor, que estão entre os mais empregados no cenário industrial: a perturbação degrau e o PRBS (*Pseudo Random Binary Sequence*) (MACHADO, 2004).

Em posse das curvas e valores de saída obtidos do sistema é possível dar início ao procedimento de identificação. Para alguns modelos, a forma de onda obtida a partir do sinal de entrada é suficiente para obter os parâmetros. Já para outros, há a necessidade de realizar estimação pelo método dos Mínimos Quadrados. Para estes, a ferramenta MATLAB apoiou o processo de modelagem.

Por fim, após a obtenção dos modelos que representam a dinâmica do motor em estudo, foi realizada a verificação de qual modelo é mais adequado via simulação e comparação com o modelo real.

1.6. ESTRUTURA DO TRABALHO

O primeiro capítulo do trabalho de conclusão de curso é dado pela introdução ao tema de desenvolvimento. Nele consta uma breve introdução a alguns tipos de máquinas elétricas, a importância da obtenção de um modelo matemático e uma ideia de como este pode ser obtido através dos processos de modelagem. Nele também estão descritos os objetivos do trabalho e os métodos que serão adotados para o seu desenvolvimento.

O capítulo seguinte é constituído de uma fundamentação teórica sobre as características dos motores de ímãs permanentes no rotor, a metodologia de acionamento e também sua estrutura.

No terceiro capítulo, são descritos todos os módulos necessários para a realização da prática proposta, detalhando o dispositivo empregado no controle e processamento e salientando os seus recursos que serão utilizados.

No quarto capítulo são detalhados os tipos de modelagem de sistemas, os tipos de modelos matemáticos que podem representá-los para diferentes tipos de identificação e as etapas que compõe um processo de modelagem em detalhes.

O quinto capítulo apresenta os resultados obtidos ao longo do desenvolvimento do trabalho.

O último capítulo apresenta as considerações finais a respeito do trabalho em questão e as conclusões obtidas a partir da análise dos resultados.

2. CARACTERÍSTICAS DO BLDC

2.1. ROTOR

Existem algumas formas de se configurar a disposição dos ímãs permanentes em um rotor. As mais comuns são rotor de ímã permanente na superfície (*surface magnets*), rotor com ímã permanente interno transversal, rotor com ímã permanente interno longitudinal e rotor Lundell (BEZERRA, 2004).

O tipo de rotor utilizado no presente trabalho é o de ímã permanente na superfície do rotor com dois pares de polos. Esse tipo de rotor é seguramente o mais utilizado em aplicações práticas, devido à sua simplicidade construtiva em relação aos demais. O uso desta configuração de rotor viabiliza o emprego de várias possibilidades de enrolamento em seu estator, de tal maneira que possa apresentar uma forma de onda de tensão induzida senoidal ou então formas de onda não senoidais variadas (trapezoidais, retangulares, etc.) (MONTEIRO, 2006).

Essa máquina possui baixo momento de inércia no rotor devido, principalmente, à sua estrutura oca. Outra vantagem reside no fato do rotor apresentar uma única fonte de calor e próxima à superfície, devido há ausência de enrolamento rotórico, o que facilita muito a refrigeração e conseqüentemente reduz o tamanho do motor em relação às máquinas CC convencionais, ou mesmo de indução (MONTEIRO, 2006).

2.2. ESTATOR

Os motores síncronos de ímãs permanentes são classificados de acordo com o perfil da forma de onda da força eletromotriz. Existem duas principais classes, as máquinas senoidais ou motores de corrente alternada sem escovas, e as máquinas não senoidais que também são chamadas de máquinas retangulares ou trapezoidais, que são os motores de corrente contínua sem escovas.

A diferença entre estas duas máquinas está na disposição das bobinas em seu estator, onde os enrolamentos das fases do motor de ímã permanente de onda senoidal se assemelham aos das máquinas de indução, que apresentam

encurtamento de passo. Para este tipo de enrolamento o torque é menos oscilatório devido à característica de seu campo girante, no entanto o custo de manufatura é mais elevado comparado com o de acionamento não senoidal em função de sua complexidade.

Enrolamentos de passo encurtado são geralmente usados em máquinas polifásicas CA, pois reduzem significativamente, a amplitude de determinadas harmônicas da distribuição da FMM (Força Magneto-Motriz), e, portanto da tensão induzida no enrolamento (SEM, 1997, p.573).

Os motores BLDC apresentam um fluxo de entreferro não senoidal, sendo que em alguns casos é aproximadamente retangular ou trapezoidal. Os enrolamentos desse tipo de motor são dispostos de maneira mais simples que os motores de corrente alternada sem escovas, não apresentando grande complexidade em sua construção.

Outra diferença relevante entre o acionamento senoidal e o não senoidal é que este último produz um torque maior para um mesmo valor de corrente, considerando máquinas com características equivalentes de potência e velocidade nominais (MONTEIRO, 2006), sendo que, além disso, pode ainda trabalhar com uma amplitude de velocidade maior.

As diversas formas para a disposição dos enrolamentos estáticos de máquinas síncronas de ímãs permanentes, onde esta configuração irá influenciar no desempenho no sentido de minimizar as ondulações no conjugado rotórico.

2.3. ACIONAMENTO ELÉTRICO

De forma geral, o acionamento elétrico do motor é composto por cinco partes fundamentais representadas na Figura 4. Consiste em uma fonte de energia do sistema, composta por uma fonte de tensão ou de corrente ligada a uma ponte inversora, responsável pelo chaveamento do motor sem escovas com ímã permanente. Neste, é posicionado um sistema de detecção da posição do rotor, que adquire a informação necessária para o sincronismo do chaveamento, viabilizando o processo de controle realizado pelo DSC.

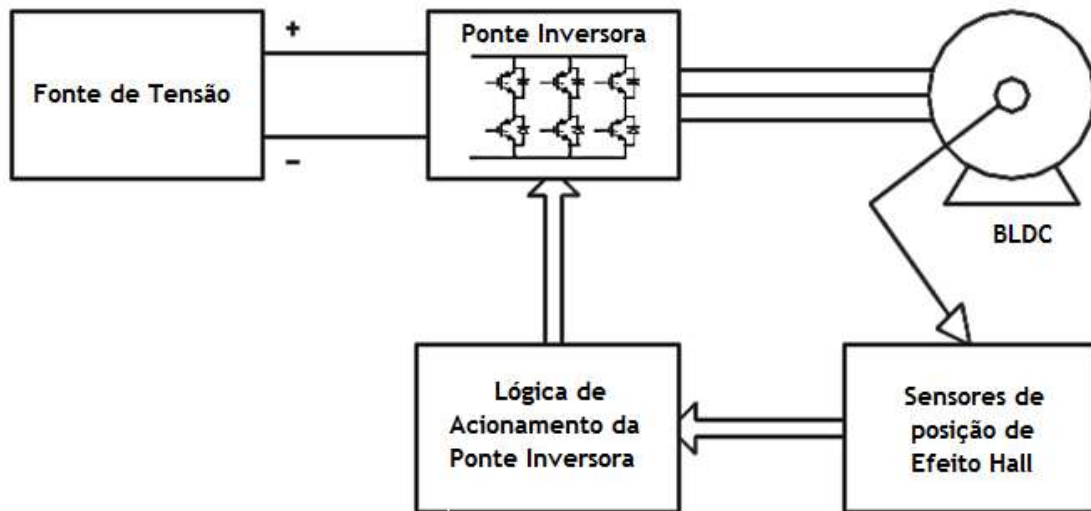


Figura 4 - Estrutura de acionamento
 Fonte: Adaptado de: MONTEIRO, (2006, P.41)

O conjunto inversor utilizado neste trabalho, como na Figura 5, é do tipo Inversor de tensão (*Voltage Source Inverter - VSI*), e sua topologia é a mais utilizada para acionamento de máquinas elétricas, sendo constituída por três braços de acionamento. Cada ramo possui dois MOSFETs (*Metal Oxide Semiconductor Field Effect Transistor*) IRF540, ambos com diodos anti-paralelos, que são necessários em sistemas com características indutivas, que permitem a circulação de corrente durante a abertura das chaves, desempenhando papel de roda-livre, e um *shunt* para a medição de corrente no valor $0,05\Omega$ para a necessidade do controle de corrente. Para a proteção do sistema, é feito o monitoramento da corrente total através de outro *shunt* de $0,05\Omega$ posicionado em série com os três braços, para os casos de sobrecarga ou curto-circuito.

Através desta configuração é realizado o acionamento do BLDC de forma que a frequência da componente harmônica fundamental de tensão ou corrente na saída da ponte inversora seja sempre igual à frequência elétrica do rotor da máquina, de tal maneira que a ponte inversora produza um campo no estator sempre em sincronismo com o campo do rotor. Para a realização deste sincronismo do campo girante com o rotor é utilizado um sensor de posição de efeito Hall.

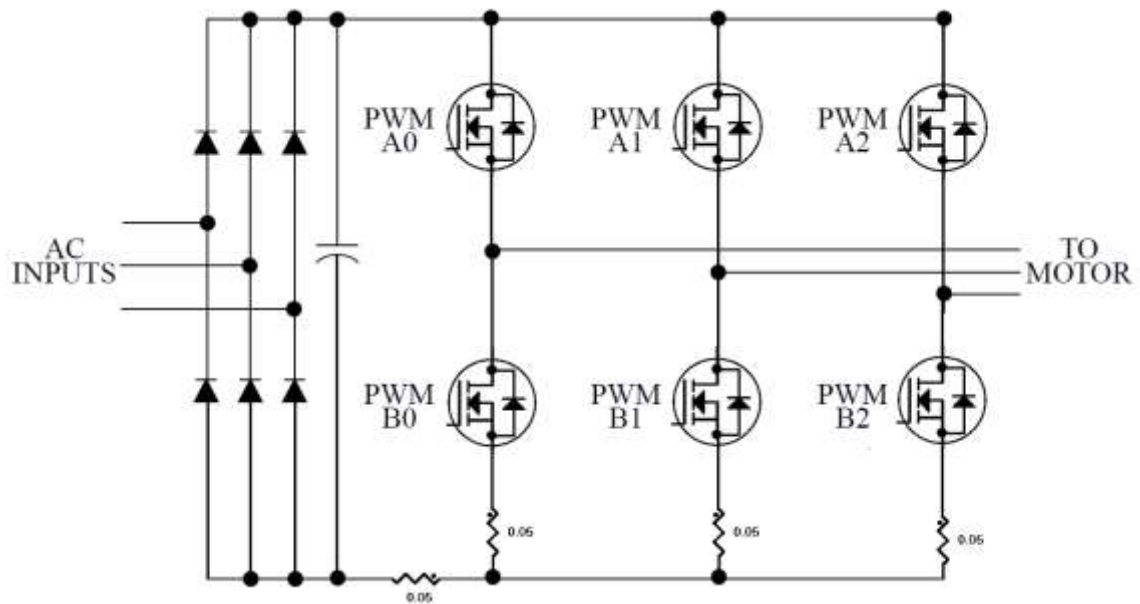


Figura 5 – Conjunto inversor
 Fonte: Adaptado de: FREESCALE (2010, P.269)

São posicionados três sensores Hall instalados no estator distribuídos a cada 120° . Estes enviam sinais lógicos ao circuito de controle que, através da informação dos três sinais, possui a informação da posição rotórica a cada 60° elétricos e determina qual chave da ponte inversora deve ser acionada a cada instante.

Somente duas chaves da ponte inversora estão ligadas num mesmo instante no modo de operação a seis pulsos, que será o empregado neste trabalho, sendo uma chave da semi-ponte negativa e outra da semi-ponte positiva. Através da Figura 6, são representados os sinais provenientes do sensor Hall e a comutação das chaves em função deste.

Contudo, pode-se obter melhor qualidade nas tensões e correntes sintetizadas pelo inversor com o emprego de técnicas de modulação por largura de pulso.

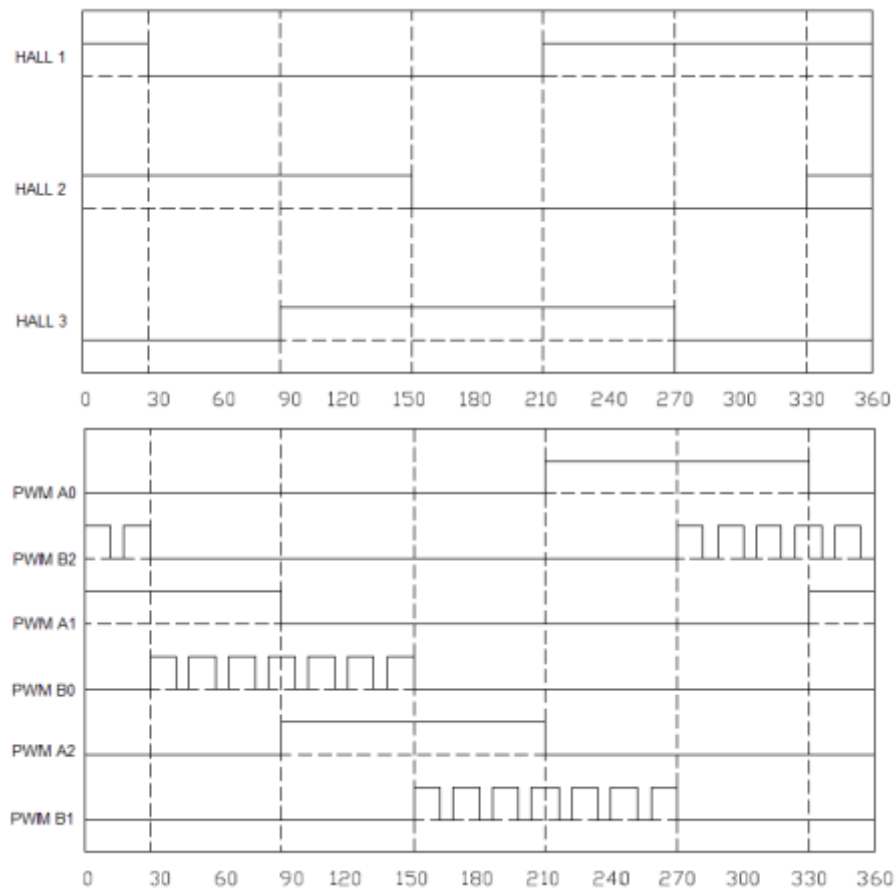


Figura 6 – Sinais provenientes do sensor Hall e dos sinais de acionamento da ponte
Fonte: Adaptado de: MONTEIRO, (2006, P.57)

O controle que será empregado é realizado através da modulação em largura de pulso na chave semicondutora superior ou na inferior. Aplicando-se o sinal PWM (*Pulse width modulation*) somente nas chaves da semi-ponte negativa, obtém-se uma diminuição das ondulações na corrente de estator sendo mais indicado para diversas aplicações.

É inserido um tempo morto (*dead-time*) nos PWMs de mesmo braço, como uma proteção das chaves semi-condutoras que possuem atraso de comutação intrínsecas, para não ocorrer uma situação curto-circuito em algum dos braços.

É possível fazer o monitoramento da posição rotórica *sensorless* utilizando várias estratégias. Sem a necessidade dos sensores, conseqüentemente o custo do projeto é reduzido. No entanto este tipo de acionamento possui várias desvantagens como o aumento na complexidade do controle, inviabilizando seu uso em algumas aplicações.

2.3.1. Driver

Para o acionamento da ponte inversora que é constituída por transistores de efeito de campo, é necessário um *driver* para o acionamento dos MOSFETs. Para a condução do MOSFET é necessária uma referência de tensão entre o *gate* e o *source*. Como os MOSFETs superiores possuem referências diferentes entre o *gate* e o dreno de cada fase, este *driver* é necessário para a identificar cada uma das referências e assim, possibilitar o chaveamento.

Este *driver* possui uma interface SPI (*Serial Peripheral Interface*), desta forma é possível estabelecer uma comunicação com o DSC para diagnósticos detalhados, sincronismo de fase e proteção programável. Este é responsável por aplicar os sinais de PWM do DSC no *gate* dos seis MOSFETs destinados a ponte inversora. Através da Figura 7 é possível verificar a configuração empregada.

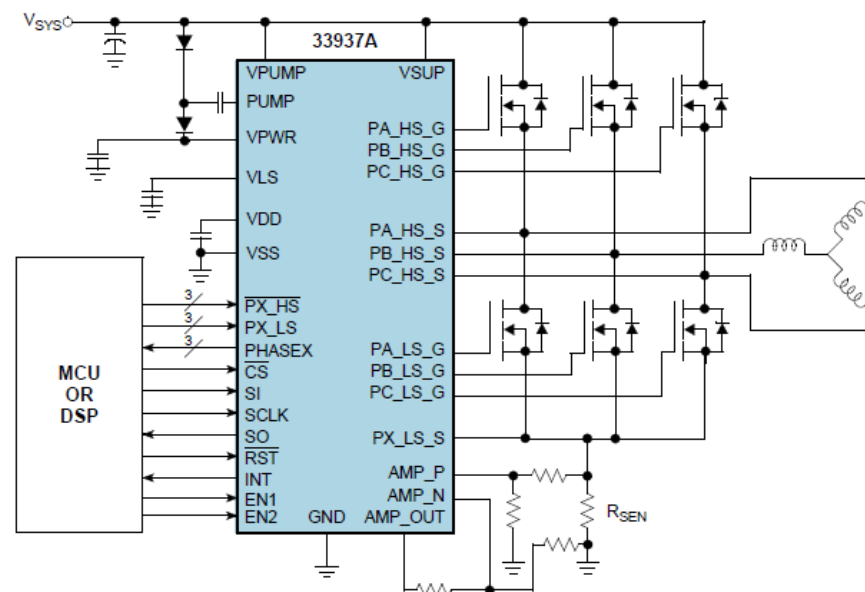


Figura 7 – Driver utilizado no acionamento da ponte inversora
 Fonte: Adaptado de: FREESCALE (2012, P.1)

2.3.2. Sensor Hall

Os métodos mais comumente aplicados na detecção da posição do rotor em um motor BLDC é o que utiliza o sensor de efeito Hall e a força eletromotriz inversa,

conhecido também como *sensorless*. A força eletromotriz inversa é utilizada em aplicações de baixo custo e não utiliza sensores para detectar a posição do rotor. Sua desvantagem é não abranger todas as faixas de velocidade (somente de média a alta) e não permite controle de posição. Para aplicações industriais, utilizam-se os sensores de efeito Hall, o que garante um monitoramento preciso (permitindo o uso de qualquer faixa de velocidade), proporcionando uma excelente solução para controle destes dispositivos (NOLL, 2010).

Este sensor é baseado no efeito Hall, como o nome sugere, pois, no momento em que um campo magnético atua perpendicularmente à sua superfície, gera uma tensão nos elementos Hall internos do sensor. Essa tensão é amplificada e aplicada a um *Schmitt Trigger* que, por sua vez, excita um transistor NPN, assim este sinal lógico é enviado ao controlador.

O motor BLDC utilizado no presente trabalho contém três sensores de efeito Hall com um espaçamento de 120° entre eles, e o campo magnético recebido por eles é derivado de dois ímãs permanentes posicionados no eixo do rotor, um em cada extremidade do eixo, de forma que produza um campo perpendicular sobre os sensores. A Figura 8 descreve a disposição dos sensores acoplados ao eixo do rotor.

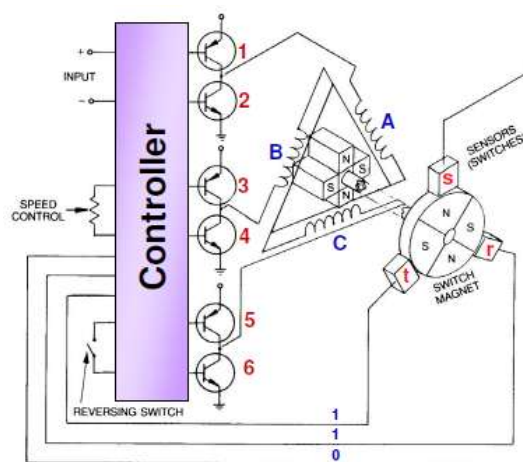


Figura 8 – Representação da disposição dos sensores Hall no rotor
 Fonte: Adaptado de: FREESCALE (2006, P. 17)

O segredo para promover a comutação de um motor BLDC é detectar a posição do rotor e em seguida energizar as fases que irão produzir a maior quantidade de torque nos ímãs do rotor, assim, de uma forma sequencial é feita a

comutação das chaves de acordo com a leitura dos sensores como é identificado através da Figura 9 determinando a sequência de fase.

Sensor Hall A	Sensor Hall B	Sensor Hall C	Fase A	Fase B	Fase C
1	1	0	$-V_{DCB}$	NC	$+V_{DCB}$
0	1	0	NC	$-V_{DCB}$	$+V_{DCB}$
0	1	1	$+V_{DCB}$	$-V_{DCB}$	NC
0	0	1	$+V_{DCB}$	NC	$-V_{DCB}$
1	0	1	NC	$+V_{DCB}$	$-V_{DCB}$
1	0	0	$-V_{DCB}$	$+V_{DCB}$	NC

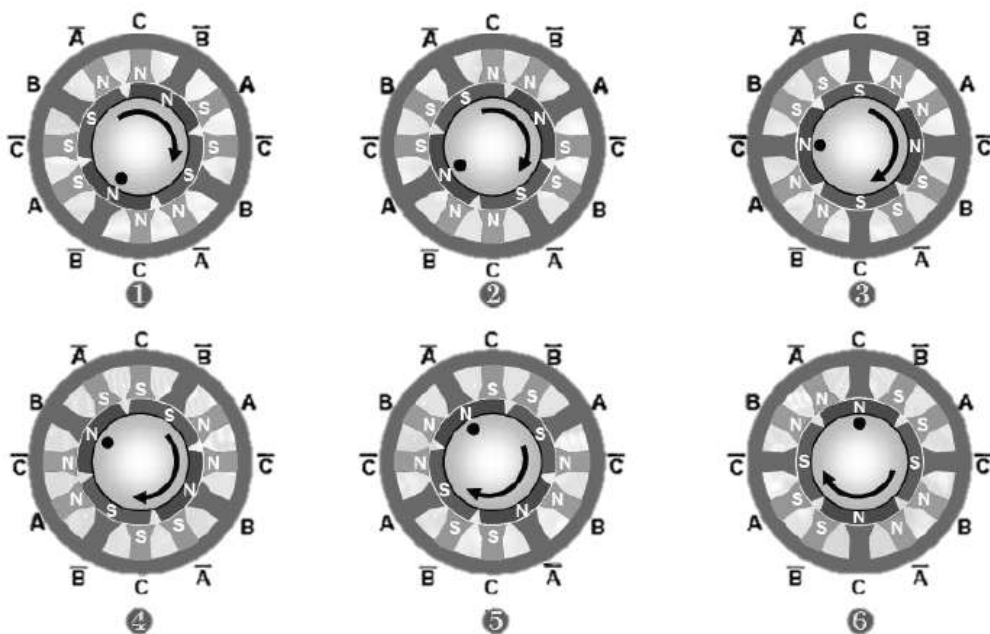


Figura 9 – Sequência de fase em função dos sensores
 Fonte: Adaptado de: FREESCALE (2006, P. 26)

3. CONTROLADOR DA PLATAFORMA

A implementação do controle e acionamento do motor foram realizadas utilizando um controlador que possui determinadas características a fim de satisfazer as necessidades das práticas a serem elaboradas. Neste capítulo será descrito o processador utilizado e seus periféricos.

O controlador escolhido foi um DSC, que corresponde a um misto entre microcontrolador e processador de sinais digitais – DSP, ou seja, combina em apenas um *chip* as vantagens de processamento de um DSP com todas as funcionalidades que um microcontrolador possui (FREESCALE, 2010).

A sua escolha foi realizada levando-se em conta algumas características que podem ser observadas neste dispositivo, como as citadas abaixo, que o tornam interessantes a esta aplicação:

- Operação em frequência de 60 MHz;
- Possui quatro saídas PWM com um submódulo de dois PWMs cada;
- Dois ADCs com oito canais de 12 *bits*, com conversão em 600 ns, e possuem amplificação programável (PGA);
- Conversor digital – analógico;
- Porta de comunicação serial (SCI);
- Módulo de interface periférica serial (SPI);
- Barramento I²C (*Inter Integrated Circuit*);
- Quatro *timers* de 16 *bits*.

3.1. MÓDULO DE PWM

Uma estratégia de controle do motor BLDC é através do controle do inversor baseada na modulação por largura de pulso, ou PWM. Para este tipo de acionamento é necessário que o controlador possua este periférico.

Neste caso, o DSC possui quatro saídas PWM, onde cada módulo possui um submódulo, indicado pela Figura 10, onde são configuradas todas as suas características como o tempo morto, complementaridade dos PWMs e *fault protection*, que são utilizados no acionamento do motor. Estes periféricos estão

implementados em *hardware*, o que torna possível a obtenção de um melhor desempenho do sistema.

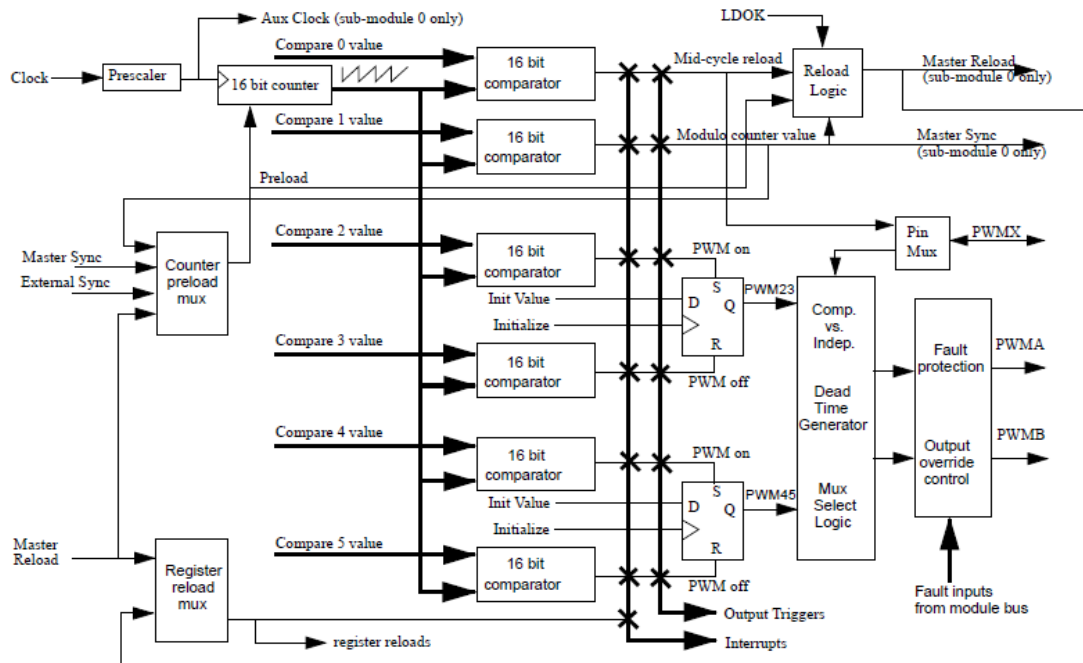


Figura 10 – Submódulo do PWM
 Fonte: Adaptado de: FREESCALE (2010, P.191)

Devido às características intrínsecas das chaves, é necessário inserir um tempo morto que é responsável por atrasar o sinal de *gate* dos MOSFETs, de modo a não deixar o MOSFET superior conduzir simultaneamente com o inferior, evitando possíveis curtos-circuitos de braço devido a atrasos da comutação do chaveamento. Este tempo morto é estabelecido de acordo com o componente utilizado para o chaveamento, obtido com as especificações fornecidas pelo fabricante. O DSC utilizado possui uma configuração de tempo morto que é configurado em *hardware*, pois esta manipulação por *software* pode gerar atrasos prejudicando a eficiência do sistema.

Os valores de referência para a configuração dos PWMs são obtidos a partir do controle estabelecido, e também de acordo com a frequência a ser utilizada no chaveamento.

Uma das características interessantes encontradas neste módulo é a atuação de desligamento dos PWMs via *hardware* pelo mecanismo *fault protection*, garantindo um acionamento rápido e assim a qualquer anomalia encontrada, a saída

do PWM é interrompida. A utilização do *fault protection* é interessante no caso da necessidade de alguma atuação de proteção, para os casos de curto-circuito ou potência excessiva, e quanto maior a velocidade de resposta desta proteção maior a garantia de que o sistema não sofra danos.

3.2. SPI

Para a comunicação com o *driver* de potência (MC33937), foi utilizado um módulo de comunicação SPI. As funcionalidades *dead-time* e *fault protection* vistos anteriormente poderiam ser implementadas através deste mecanismo de comunicação com o *driver*, porém, para esta aplicação, optou-se por configurá-las no próprio DSC.

Com isso, o SPI será empregado apenas no diagnóstico de erros fornecidos pelo MC33937.

3.3. SCI

Para a comunicação serial com um microcomputador será utilizado o módulo SCI. Através deste módulo é possível a utilização de *softwares* de depuração, como a ferramenta de testes Freemaster disponibilizada pela Freescale. Neste caso, a aplicação se torna muito importante para fins de depuração do código, pois nela é possível monitorar todas as variáveis do *firmware* com o sistema em funcionamento, tornando a plataforma utilizada mais didática e simples.

3.4. PONTO FLUTUANTE

O DSC utilizado é capaz de trabalhar apenas com números inteiros, sendo seus registradores e acumuladores de 16 *bits*, dificultando a manipulação de números reais que são utilizados frequentemente no processo de controle. DSCs ou microcontroladores que possuem uma arquitetura com ponto flutuante facilitariam a manipulação desta situação e também trariam agilidade ao processo. No entanto estes circuitos processadores de ponto flutuante exigem operações matemáticas intensas, aumentando a sua complexidade de construção, que o encarecem (BEZERRA, 2004).

Na exigência de se trabalhar com números reais neste tipo de arquitetura, é necessário serem feitas manipulações para a adequação, destinando alguns *bits* à parte fracionária. Esta manipulação acarreta em uma aproximação do valor, e assim há necessidade da aplicação adequada para que se atinja a precisão necessária.

Esta normalização é adotada por diversos fabricantes, sendo que a IEEE-745 especifica regras para a execução de operações aritméticas de números binários com ponto flutuante.

4. MODELAGEM APLICADA A UM MOTOR BLDC

Modelagem matemática é a prática de definir um modelo que represente as características desejadas de um sistema real. A representação utilizando modelos é essencial às simulações e ao controle de processos.

Neste trabalho, busca-se obter uma representação satisfatória do motor BLDC anteriormente descrito através de um modelo linear matemático.

Modelos lineares podem ser exibidos como funções de transferência, espaços de estados ou modelos auto-regressivos. Dentro de cada uma destas formas há uma variedade de formatos. Uma função de transferência que pode estar na forma contínua ou discreta, ou um espaço de estados que pode assumir a forma canônica controlável ou observável são exemplos de variações.

4.1. TIPOS DE MODELOS

Modelos matemáticos são classificados de acordo com alguma de suas características. A seguir será realizada uma revisão conceitual das mais relevantes às análises a serem realizadas no presente trabalho.

4.1.1. Modelos estáticos e dinâmicos

Para processos estáveis, é possível determinar um modelo estático. Este modelo apenas expressa a relação entre a entrada e a saída no estado estacionário, não fornecendo informações a respeito da dinâmica, porém é útil na determinação do conjunto de valores para o sinal de entrada que façam o processo responder com valores dentro de uma desejada faixa (ÅSTRÖM & HÄGGLUND, 2006, p.13). Além disso, são interessantes para análises em que a dinâmica é rápida ou lenta demais em relação ao período em verificação.

O modelo estático consiste em uma relação entre os valores de entrada e de saída no estado estacionário do sistema, e geralmente pode ser descrito por equações algébricas.

Para o caso do motor BLDC, o modelo estático pode ser interessante para verificar a faixa de operação do motor, e com isso poder analisar que tipo de elemento será utilizado para a obtenção dos valores de saída.

Modelos dinâmicos servem para ilustrar o comportamento do sistema durante o transitório e retratam a dependência temporal. Segundo Aguirre (2007), todo sistema real é dinâmico. Logo, quando esta dinâmica é relevante, este é o tipo de modelo a ser utilizado. Sua descrição se dá por equações diferenciais ou a diferenças e pode incluir equações algébricas.

4.1.2. Modelos discretos e contínuos

Nos modelos contínuos a representação do sistema se dá pela evolução contínua do sistema no tempo, e em modelos discretos por amostras. São descritos por equações diferenciais e equações de diferenças, respectivamente.

Para projetar um algoritmo de controle para ser utilizado no DSC, um modelo discreto é necessário. Ele pode ser obtido diretamente na forma discreta, ou pode-se, alternativamente, realizar a discretização de um modelo contínuo.

4.1.3. Modelos determinísticos e estocásticos

Esta classificação está atrelada à presença de variáveis aleatórias no modelo. Modelos determinísticos são aqueles que para dado conjunto de valores na entrada, há apenas um conjunto de valores resultante na saída, sem apresentar incertezas. Já nos modelos estocásticos, as incertezas existentes nas situações reais são representadas por meio das variáveis aleatórias, ou seja, uma saída em um dado instante não pode ser prevista pelo seu comportamento no passado.

Na indústria, não-linearidades presentes nos processos são confundidas com distúrbios não medidos. Desta forma, é importante que os modelos contemplem na sua obtenção estes componentes estocásticos, bem como a distribuição estatística do ruído de medida (MACHADO, 2004).

Para verificar a influência da consideração do erro na modelagem do sistema em questão, os dois tipos de modelos foram avaliados.

4.2. IDENTIFICAÇÃO

Os tipos de modelagem podem ser divididos em termos do conhecimento disponível a respeito do sistema em análise. Nesta classificação estão as modelagens caixa branca, caixa preta e caixa cinza (XXVIII ENCONTRO NACIONAL..., 2008).

A primeira, também chamada de modelagem pela física, pela natureza do processo, fenomenológica ou conceitual, exige, além de uma grande familiaridade com o sistema, o conhecimento a respeito dos fenômenos que regem seu comportamento (AGUIRRE, 2007, p.51). Nela, o processo é descrito em termos de equações, geralmente envolvendo variáveis advindas da estrutura.

No modelo caixa preta, apenas dados colhidos do sistema são suficientes para obter um modelo, ou seja, não é necessário conhecer previamente o sistema, o que dá uma alta aplicabilidade a este método. Também pode ser denominado como identificação ou modelagem empírica.

O modelo caixa cinza, como o próprio nome sugere, é intermediário aos dois anteriormente descritos. Ele utiliza as mesmas estruturas aplicadas à modelagem caixa preta, porém, os parâmetros poderão ser determinados com base no conhecimento prévio, informação não-experimental, ou auxiliar do sistema, como por exemplo um modelo estático.

No caso do motor, é possível realizar a modelagem pela física considerando um modelo equivalente para o motor, apresentado na Figura 11, e então o representando em forma de equações diferenciais.

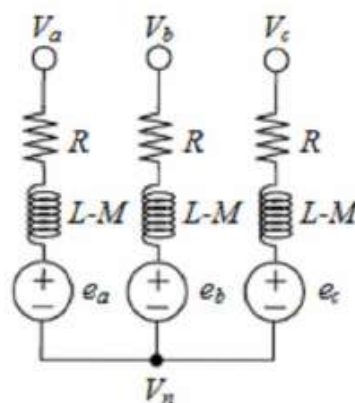


Figura 11– Circuito equivalente de um motor BLDC em Y
Fonte: Trindade (2009, P.9)

Porém, considerando o modelo anterior e as perdas mecânicas no motor, são necessários, além do levantamento das equações que regem os fenômenos envolvidos, os valores de resistência da armadura por fase, indutância por fase, indutância mútua, momento de inércia do rotor e coeficiente de atrito viscoso, que não foram disponibilizados pelo fabricante em sua totalidade. Neste caso, seria necessária a utilização de instrumentos e ensaios para a determinação dos mesmos, que demandariam tempo e equipamentos que talvez não estivessem disponíveis.

Em muitas outras situações práticas além dos fatores tempo e conhecimento necessários à prática de modelagem pela física, pode ocorrer ainda que o sistema a ser analisado esteja inacessível, ou que seja extremamente complexo. Isso acaba demandando algum método alternativo. Pensando-se em um motor, este pode já estar instalado como parte de um sistema maior, impedindo que se realizem ensaios invasivos. Para casos como esses, as soluções são denominadas técnicas de identificação (AFONSO, 2013).

No procedimento de identificação, a obtenção do modelo se dá a partir de dados adquiridos de saída do sistema para determinados sinais de entrada. Uma relação de causa e efeito entre estes valores deve fornecer uma representação aproximada do sistema (AGUIRRE, 2007, p.81).

Através da Figura 12 é representado o diagrama de blocos das principais etapas de uma identificação.

Como já foi comentado, este tipo de modelagem requer o levantamento de dados de entrada e saída do sistema real, que corresponde à primeira etapa do procedimento.

Em uma etapa futura, a de validação, será necessária a utilização de um novo conjunto de dados para testar o modelo. Desta forma, o interessante é que nesta fase sejam obtidos pelo menos dois conjuntos de dados observados do sistema ou, na impossibilidade, o conjunto único de dados deve ser dividido entre as duas necessidades.

É importante ressaltar que todos os sistemas são praticamente não-lineares e, no processo de modelagem, são aproximados para sistemas lineares. Assim, o modelo obtido representará a dinâmica do sistema em torno de um ponto de operação correspondente àquele considerado na aquisição dos dados (AGUIRRE, 2007, p.484). Levando-se isso em consideração, as medições devem ser realizadas

sob condições muito próximas, para evitar uma comparação incorreta no momento da validação.

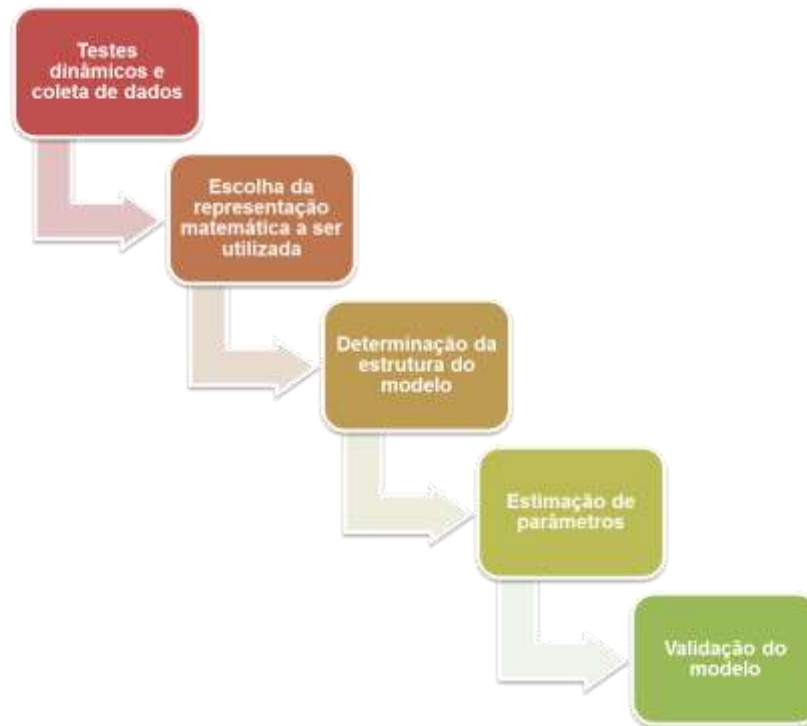


Figura 12 – Etapas de um processo de identificação
 Fonte: Autoria Própria

Exceto o caso em que se esteja trabalhando com dados anteriormente observados de um processo em seu modo normal de operação, para obter os dados necessários à modelagem é preciso decidir qual o sinal de excitação a ser empregado.

Os sinais mais empregados e conhecidos na indústria são o sinal degrau e o PRBS (*Pseudo Random Binary Sequence*). A perturbação degrau promove a excitação, principalmente, das características estacionárias do sistema, isto é, o intervalo de baixas frequências (MACHADO, 2004). Este tipo de sinal é largamente empregado nas indústrias pela sua fácil aplicação na execução da identificação. No entanto para determinadas plantas, com este tipo de sinal não é possível obter todos os parâmetros necessários.

O PRBS possui a grande vantagem de excitar, com muita intensidade, uma grande faixa de frequência. Ele assume dois valores, um nível alto e um baixo, permanecendo em períodos distintos aleatoriamente. Para isso, deve ser estipulado

o período máximo e mínimo que o sinal deve ficar num determinado valor e também sua amplitude. A característica de excitação dos polos torna este tipo de sinal uma perturbação muito útil na identificação, por enriquecer as informações do sinal de resposta da planta.

A escolha de um sinal de excitação deve, então, ser uma relação de compromisso entre simplicidade e nível de exigência de informação.

A segunda etapa do processo de identificação diz respeito à escolha da representação a ser utilizada dentre as diversas formas existentes. Para métodos determinísticos, utilizam-se, por exemplo, funções de transferência e espaços de estado. Para os estocásticos, ARX e ARMAX são exemplos de representações. Os modelos selecionados para a presente aplicação estão divididos entre as duas metodologias e serão detalhados nos tópicos seguintes.

Determinação da estrutura do modelo se refere, basicamente, à escolha do número de polos e zeros que representarão o sistema, bem como o atraso puro de tempo (AGUIRRE, 2007, p.81).

A etapa de estimação dos parâmetros consiste em obter, a partir de algum algoritmo, os parâmetros desconhecidos do modelo escolhido. Uma forma muito utilizada é a estimação por mínimos quadrados, que estima os parâmetros de forma que a soma dos quadrados dos resíduos seja mínima. O conceito desta aproximação é que se um determinado número de medidas é realizado para uma mesma quantidade física e se estas medidas estão sujeitas apenas a erros aleatórios, então a teoria dos mínimos quadrados estabelece que o valor mais provável da quantidade medida seja aquele que faz a soma dos quadrados dos erros um mínimo.

Finalmente é necessário realizar a validação do modelo resultante para observar se este representa as características desejadas da dinâmica do sistema. Uma forma de verificar se o modelo obtido representa de forma suficientemente adequada o sistema real para determinada aplicação, é a simulação do modelo obtido sem realizar nenhum ajuste adicional, fazendo uso de valores de entrada diferentes daqueles utilizados no procedimento de modelagem. Esse cuidado deve ser tomado para verificar se a solução de identificação obtida é geral ou não, em outras palavras, a generalização do modelo. Após isso, comparam-se os valores de saída prática e simulada (AGUIRRE, 2007, p.79).

A qualidade de um modelo depende da finalidade com que ele foi obtido. No caso de um modelo obtido com o objetivo de realizar um controle a partir do mesmo, este será avaliado no momento em que o controlador desenvolvido a partir dele for aplicado sobre a planta. Apresentando um resultado adequado, julga-se o modelo como válido (FERNANDES, 2006).

4.2.1. Método determinístico – resposta ao degrau

O método determinístico de modelagem mais comum é o método de resposta ao degrau (AGUIRRE, 2007, p.689). Ele é determinístico porque o ruído presente no processo não é tratado, apesar da sua existência não ser ignorada. Este método é um tipo de *Bump Test* (ÅSTRÖM & HÄGGLUND, 2006, p.47), que verifica o impacto de uma mudança abrupta na variável manipulada.

No método de resposta ao degrau, com o sistema em repouso, altera-se rapidamente a variável de controle para um valor que deve ser suficientemente alto, de modo a obter uma boa relação sinal-ruído, que é a única ação tomada para minimizar seus efeitos. A resposta do sistema a esta alteração brusca deve ser observada através da leitura da variável de processo.

Repetindo este processo para diferentes valores de entrada e condições variadas, é possível determinar para qual faixa de valores o sistema se comporta como linear e permite verificar se o sistema sofre alterações em diferentes pontos de operação (ÅSTRÖM & HÄGGLUND, 1995, p.12). Como os sistemas em questão são considerados lineares, a saída é proporcional ao sinal aplicado na entrada. Logo, é interessante que as medições da variável de controle sejam normalizadas, isto é, divididas pelo valor da variável de controle.

Da forma de onda gerada pela leitura da variável de processo é possível obter informações sobre o tipo de processo.

A resposta ao degrau é uma forma conveniente de caracterizar a dinâmica de um processo pela sua simples interpretação física (ÅSTRÖM & HÄGGLUND, 1995, p.13).

As propriedades estáticas de um processo podem ser descritas por um único parâmetro dado pela razão entre o valor do processo no estado estacionário e o valor do degrau aplicado. Ele é chamado de ganho do processo. Observando a

forma de onda gerada pela leitura da variável de processo, o ganho corresponde à inclinação da curva.

Para obter as propriedades dinâmicas, são necessários mais parâmetros. Há formas bem estabelecidas ao longo dos anos para fazer a modelagem do processo baseando-se em dois, três ou quatro parâmetros. Os modelos a serem utilizados estão descritos a seguir.

4.2.1.1. Modelos a dois parâmetros

Os modelos paramétricos mais simples da dinâmica de processos têm dois parâmetros (ÅSTRÖM & HÄGGLUND, 1995, p.13).

Um exemplo é o modelo *residence time approximation*, dado pela função de transferência abaixo:

$$G(s) = \frac{K}{1 + sT_{ar}} \quad (4.1)$$

O primeiro parâmetro para a determinação do modelo é o ganho estático, que fornece a característica do processo no seu estado estacionário.

O segundo parâmetro é o *average residence time*, que caracteriza o comportamento temporal. Ele corresponde a uma aproximação do tempo de estabelecimento da resposta. Assim como o ganho, pode ser obtido pela curva de resposta ao degrau como:

$$T_{ar} = \frac{A_0}{K} \quad (4.2)$$

Onde A_0 é dado por:

$$A_0 = \int_0^{\infty} (s(\infty) - s(s(t))) dt \quad (4.3)$$

E corresponde à área hachurada da Figura 13.

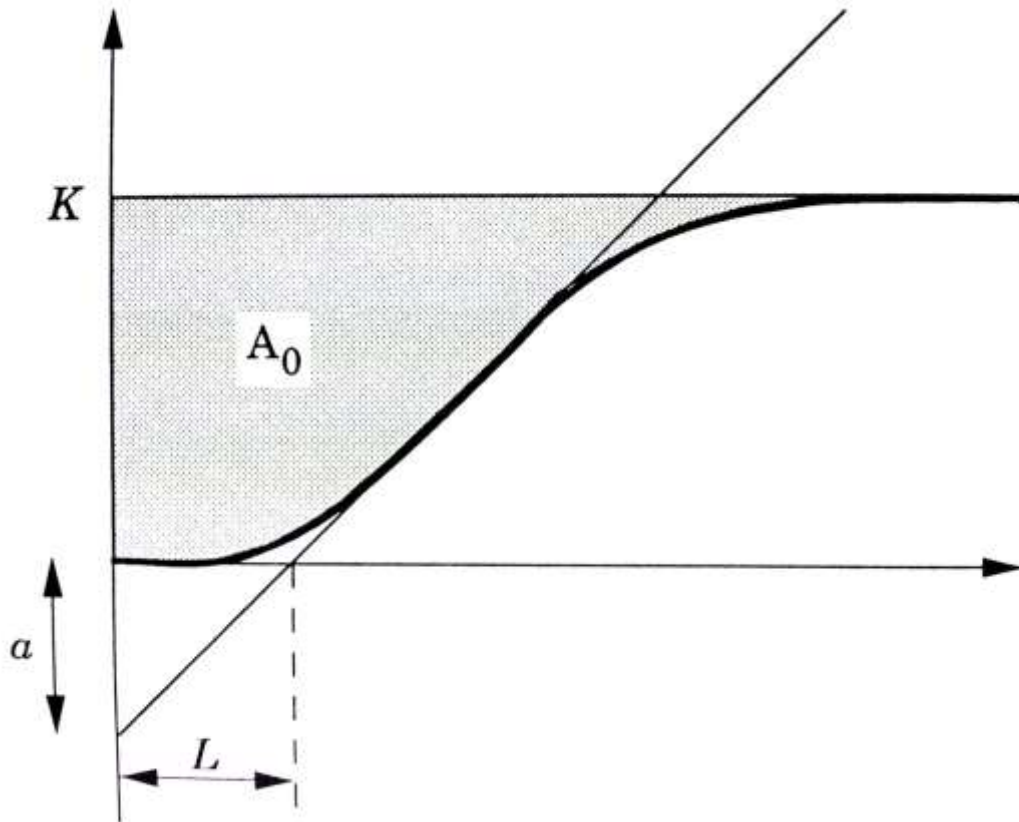


Figura 13 – Gráfico para determinação de parâmetros do modelo de resposta ao degrau a dois parâmetros
 Fonte: Åström & Hägglund (1995, P.14)

Outro modelo consiste em um integrador com tempo morto:

$$G(s) = \frac{\alpha e^{-sL}}{sL} \quad (4.4)$$

Conforme a função de transferência acima, os parâmetros necessários para defini-la são α e L . Ambos são determinados graficamente de acordo com a Figura 13. A reta traçada sobre a curva de resposta serve de apoio para encontrá-los, e é traçada tangentemente ao ponto de maior inclinação.

Este modelo é a base para o processo de sintonia de PID (Controlador Proporcional – Integral – Derivativo) pelo método de Ziegler-Nichols em malha aberta (ÅSTRÖM & HÄGGLUND, 1995, p.14).

Para decidir entre os modelos, é necessário avaliar em que estados da operação uma resposta mais próxima do real é requerida, pois cada um se comporta de uma maneira diferente ao longo do tempo.

O primeiro fornece uma melhor aproximação para grandes valores de tempo, enquanto o segundo possui um bom comportamento para valores de tempo mais baixos, correspondentes à etapa transitória da resposta.

A análise também pode ser feita para a frequência. Em altas frequências, a melhor opção é o modelo *the residence time approximation*, já para frequências intermediárias, o modelo a ser escolhido é o integrador com tempo morto (ÅSTRÖM & HÄGGLUND, 1995, p.15).

4.2.1.2. Modelos a três parâmetros

Um modelo também pode ser caracterizado por três parâmetros, o que fornecerá um resultado mais aproximado do sistema real. O modelo abaixo é o mais utilizado em artigos sobre sintonia de PID (ÅSTRÖM & HÄGGLUND, 1995, p.16).

$$G(s) = \frac{K}{1 + sT} e^{-sL} \quad (4.5)$$

Como se pode observar, os parâmetros necessários para caracterizar totalmente este modelo são K , T e L . O ganho estático K continua sendo a razão entre o valor de saída na estabilidade e o valor do degrau aplicado. L , como no modelo anterior, é o tempo morto, e pode ser obtido graficamente da mesma maneira. T é a constante de tempo que pode ser obtida de diferentes formas, porém, isso pode resultar em modelos diferentes. No primeiro, ela é dada pela diferença entre o tempo onde a reta tangente atinge o valor final e o ponto em que ela cruza com o eixo real. No segundo, corresponde ao tempo desde o cruzamento da tangente com o eixo real e o momento em que a resposta atinge 63% do valor final.

As aproximações pelo segundo método costumam ser melhores (ÅSTRÖM & HÄGGLUND, 1995, p.16).

As informações necessárias para a obtenção dos valores podem ser observadas na Figura 14.

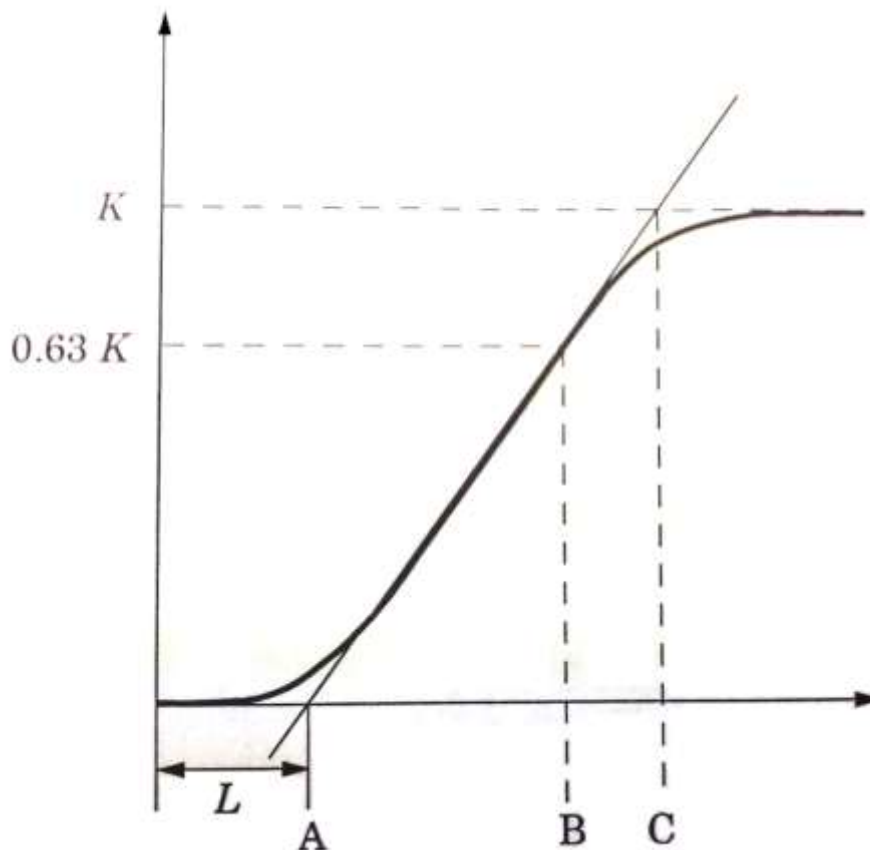


Figura 14 - Gráfico para determinação de parâmetros do modelo de resposta ao degrau a três parâmetros

Fonte: Åström & Hägglund (1995, P.17)

Apesar de muito utilizado em artigos de sintonia de PID, este modelo não funciona bem na prática (ÅSTRÖM & HÄGGLUND, 1995, p.17).

Alternativamente, os mesmos valores de K e L podem ser aplicados à função de transferência abaixo:

$$G(s) = \frac{K}{(1 + sT)^2} e^{-sL} \quad (4.6)$$

A dificuldade deste modelo está no parâmetro T , que, conhecendo-se um valor da resposta ao degrau para um determinado tempo pode ser obtido através da solução numérica da equação:

$$s(t) = K\left(1 - \left(1 + \frac{t-L}{T}\right)e^{-\frac{t-L}{T}}\right) \quad (4.7)$$

4.2.1.3. Etapas de identificação do BLDC utilizando métodos determinísticos

Como anteriormente descrito, o método determinístico selecionado para a identificação foi resposta ao degrau, por ser simples e amplamente utilizado nas indústrias.

A perturbação degrau é produzida por meio da ponte inversora contida na placa de potência através do chaveamento dos MOSFETs, que recebem os sinais PWM produzidos pelo DSC. O DSC executa um *firmware* desenvolvido especificamente para este sistema, responsável pela comutação das chaves e leitura dos sinais de interesse.

O *encoder* acoplado ao eixo do motor retorna a posição do rotor. Os dados de posição são adquiridos, interpretados e manipulados pelo dispositivo DSC, que informa a velocidade em que o motor está operando.

A validação é realizada a partir da simulação dos modelos obtidos em comparação aos valores práticos para um mesmo valor de referência.

Como a teoria de Ziegler-Nichols é mais aplicada no projeto de controladores PID, como um segundo método de validação desenvolve-se um controlador com base nos parâmetros obtidos e verifica-se seu desempenho.

4.2.2. Método estocástico

Para que o erro não seja desprezado no processo de modelagem, e, desta forma, sua influência sobre o modelo identificado seja reduzida, foram utilizados os chamados “modelos de erro na equação” e “modelos de erro na saída”, em que o erro aparece diretamente no equacionamento e na saída do modelo, respectivamente. Isso faz dos métodos aplicados estocásticos.

Estes modelos são lineares, do tipo discreto e, de forma geral, podem ser descritos da seguinte forma:

$$A(q)y(k) = \frac{B(q)}{F(q)}u(k) + \frac{C(q)}{D(q)}v(k) \quad (4.8)$$

Ou

$$y(k) = H(q)u(k) + G(q)v(k) \quad (4.9)$$

e os polinômios em função de q são definidos como:

$$A(q) = 1 - a_1q^{-1} - \dots - a_{n_y}q^{-n_y} \quad (4.10)$$

$$B(q) = b_1q^{-1} + \dots + b_{n_u}q^{-n_u} \quad (4.11)$$

$$C(q) = 1 + c_1q^{-1} + \dots + c_{n_v}q^{-n_v} \quad (4.12)$$

$$D(q) = 1 + d_1q^{-1} + \dots + d_{n_d}q^{-n_d} \quad (4.13)$$

$$F(q) = 1 + f_1q^{-1} + \dots + f_{n_f}q^{-n_f} \quad (4.14)$$

As funções $H(q)$ e $G(q)$ são ditas como funções de transferência do processo e do ruído, respectivamente (AGUIRRE, 2007, P.125).

4.2.2.1. Modelos ARX

O modelo ARX ou modelo auto-regressivo com entradas externas é um dos modelos de erro na equação de tempo discreto. A saída $y(k)$ é composta por uma função de transferência aplicada à entrada $u(k)$ somada a uma incerteza $v(k)$, um processo branco aleatório, que pode ser representado por:

$$A(q)y(k) = B(q)u(k) + v(k) \quad (4.15)$$

Os polinômios $A(q)$ e $B(q)$ da equação são arbitrários e estão em função de q^{-1} , que representa o atraso, de forma que $u(k)q^{-1} = u(k-1)$.

O ruído representado na equação por $v(k)/A(q)$ não é branco, e corresponde a $v(k)$ após um filtro auto-regressivo. Como os polos de uma função de transferência são dados pela equação característica extraída do denominador, o ruído possui os mesmos polos do processo.

4.2.2.2. Modelos ARMAX

Outro modelo de tempo discreto é o Modelo ARMAX ou modelo auto-regressivo de média móvel e entradas exógenas. Ele difere do modelo ARX no aspecto de que o processo branco aleatório neste caso passa por um filtro tipo ARMA (*Autoregressive moving average*), do tipo $G(q) = C(q)/A(q)$.

A equação que descreve este modelo é dada por:

$$A(q)y(k) = B(q)u(k) + C(q)v(k) \quad (4.16)$$

4.2.2.3. Modelos ARMA

O modelo ARMA é um caso particular do ARMAX em que $u(k) = 0$, ou seja, a equação toma a forma:

$$y(k) = \frac{C(q)}{A(q)}v(k) \quad (4.17)$$

Como não possui entradas externas, o único fator a ser considerado como entrada é o processo aleatório branco. Portanto, a saída corresponde a uma resposta filtrada do ruído.

4.2.2.4. Modelos de erro na saída

Nos modelos de erro na saída, o ruído é branco e aparece diretamente da saída do sistema (AGUIRRE, 2007, p.131).

Pode ser visto como um modelo ARX sem ruído somado a um ruído branco. A visualização pode ser facilitada expressando o modelo da seguinte forma:

Modelo ARX

$$F(q)w(k) = B(q)u(k) \quad (4.18)$$

$$y(k) = w(k) + v(k) \quad (4.19)$$

Substituindo $w(k)$ na equação acima

$$y(k) = \frac{B(q)}{F(q)}u(k) + v(k) \quad (4.20)$$

4.2.2.5. Modelo Box-Jenkins

O modelo Box-Jenkins também é um modelo de erro na saída.

$$y(k) = \frac{B(q)}{F(q)}u(k) + \frac{C(q)}{D(q)}v(k) \quad (4.21)$$

As funções de transferência do sistema e do ruído não possuem parâmetros comuns (AGUIRRE, 2007, p.131).

4.2.2.6. Determinação do grau dos polinômios

Caso a ordem escolhida para um polinômio seja superior à necessária, a função de transferência irá conter valores de polos e zeros muito próximos, que

tendem a se cancelar. Desta forma, pode-se facilmente monitorar o lugar das raízes do sistema utilizando a ferramenta MATLAB para realizar esta verificação.

4.2.2.7. Etapas de identificação do BLDC utilizando métodos estocásticos

No método estocástico de identificação de sistemas, há uma maior flexibilidade no que diz respeito ao sinal de perturbação a ser aplicado. Desta forma, foi possível buscar um sinal que excitasse de forma mais completa o sistema.

O mesmo procedimento aplicado à identificação com resposta ao degrau foi adotado para produzir o sinal de entrada com o acionamento adequado, sendo que a única alteração estará no *firmware* desenvolvido.

Da mesma forma, o DSC foi utilizado para adquirir as informações de posição geradas pelo *encoder*.

Todos os modelos estocásticos apresentados foram levantados de forma experimental. Para obter os parâmetros destes, os dados adquiridos de saída para conhecidos valores de entrada foram carregados no programa MATLAB e então a *toolbox* de identificação foi utilizada.

Os modelos encontrados foram validados na própria ferramenta seguindo a metodologia proposta na literatura estudada e por meio da realização de simulações.

5. RESULTADOS

5.1. PREPARAÇÃO DO SISTEMA

O sistema montado para a realização da modelagem do motor BLDC é formado por um *encoder* acoplado ao seu eixo através de uma estrutura de poliéster, ambos fixados através de um suporte de acrílico representados na Figura 15. Por intermédio deste conjunto é possível obter todos os dados necessários para a identificação.



Figura 15 – Sistema de acionamento e leitura de velocidade do motor
Fonte: Autoria Própria

Neste trabalho foram obtidas representações matemáticas lineares do sistema em análise. Desta forma, como primeira etapa do processo de identificação foi crucial realizar a verificação da sua linearidade através da análise estática. Esta consiste na obtenção de diversos valores de entrada e saída do sistema dentro de uma faixa em que se deseja operar. Optou-se por trabalhar variando o sinal de entrada proveniente da fonte de tensão entre 46,67% e 78,67% de razão cíclica

obedecendo a sequência de acionamento do motor BLDC. Analisando a curva levantada, observou-se que o sistema pode ser considerado linear ao longo de todo o intervalo selecionado. Na Figura 16 está representada esta curva.

Este comportamento é justificado pelas características construtivas do motor, que fazem com que a tensão seja proporcional à velocidade.

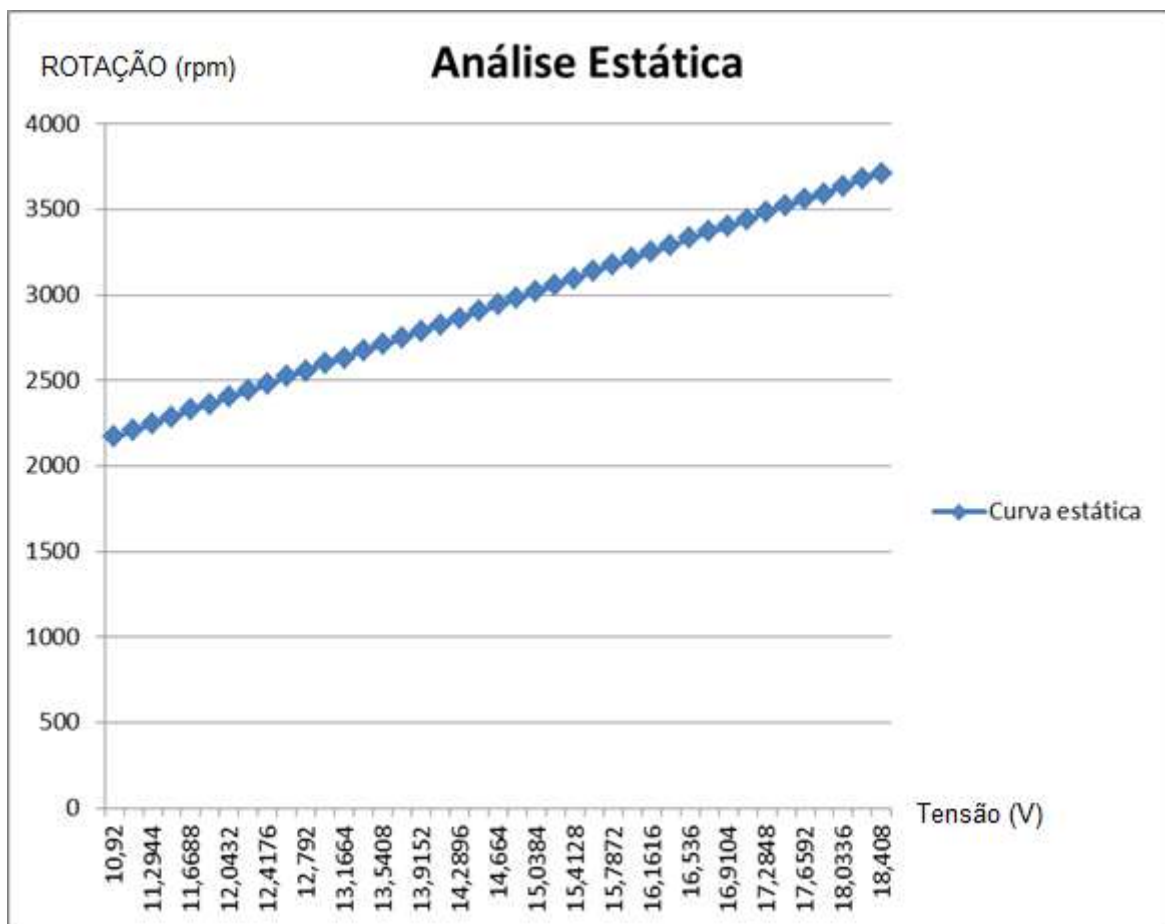


Figura 16 - Análise Estática do Sistema (Rotação X Tensão de entrada)
 Fonte: Autoria Própria

5.2. PROCESSO DE IDENTIFICAÇÃO

5.2.1. Método Determinístico

Obedecendo à sequência de etapas do processo de identificação descrito na Figura 12, inicialmente foram realizados os ensaios necessários para coletar os

dados de entrada e saída do sistema a serem utilizados na obtenção dos modelos e na sua validação. Para isso, foi definido o sinal de excitação.

Como no método determinístico utiliza-se um sinal de excitação do tipo degrau, a única característica a ser determinada é sua amplitude. Esta pode ser de qualquer valor selecionado arbitrariamente entre aqueles dentro da faixa linear de operação do sistema, representados na análise estática.

O valor de amplitude de tensão escolhido para levantar os dados a serem utilizados na modelagem foi 17,32 V, correspondente a um percentual de razão de trabalho de 74,67%.

Já para a validação do modelo, utilizou-se uma tensão de 12,43 V na entrada, equivalente a 53,6% de razão de trabalho.

A detecção de velocidade, a princípio, foi feita através do *encoder* com a leitura utilizando um *timer* temporizador configurado em modo de quadratura e sua aquisição foi feita de forma periódica, no entanto para obter-se uma resolução satisfatória o tempo de amostragem deve ser relativamente baixo (500 μ s), e em determinados casos esta taxa de atualização se torna lenta. Os valores foram adquiridos por meio do *software* Freemaster, sendo necessário disponibilizar um espaço da memória do DSC onde são armazenadas as amostras periodicamente, e em um determinado tempo a leitura é realizada via serial. A amostragem foi feita em 62,5 μ s sendo satisfatória para esta aplicação devido à dinâmica do sistema.

Este método de aquisição da velocidade é periódico se tornando interessante nos casos em que é necessária a obtenção de assíntotas como descrito em 4.2.1.1.

Para melhorar a visualização dos pontos obtidos, os valores repetidos na amostragem foram eliminados, mantendo-se a escala de tempo inalterada. Estes dados selecionados foram plotados no *software* MATLAB, como pode ser observado na Figura 17.

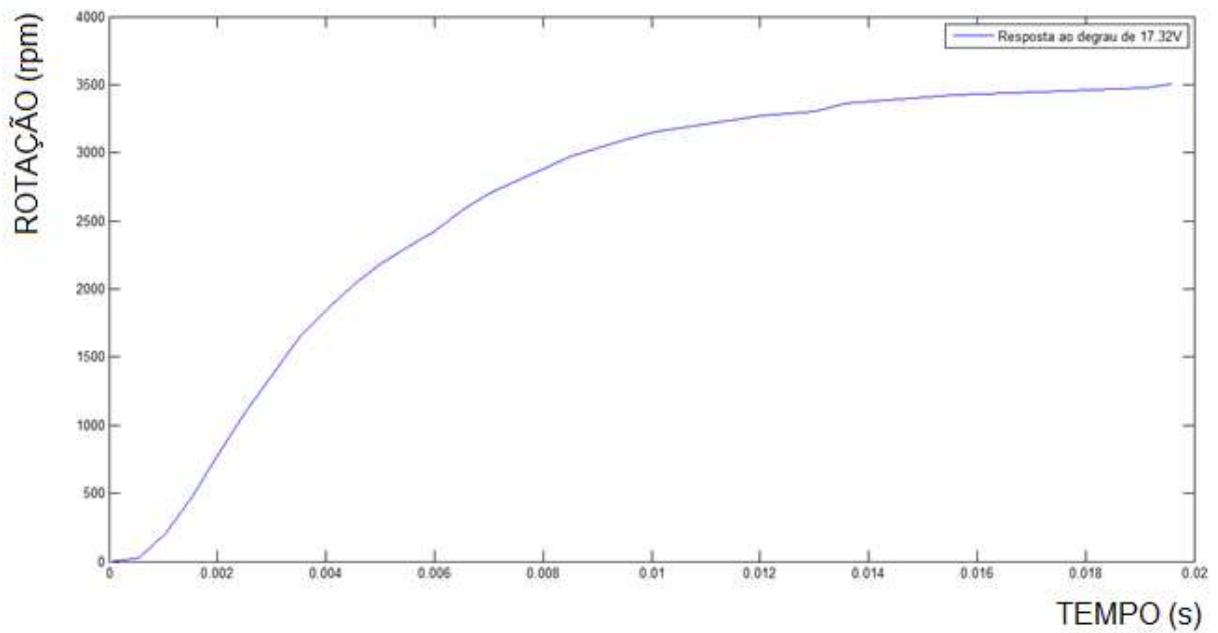


Figura 17 - Resposta ao degrau de 17,32 V
Fonte: Autoria Própria

Com a curva de resposta, foi possível iniciar a determinação dos parâmetros para os modelos de função de transferência de acordo com os métodos de Ziegler-Nichols descritos em 4.2.1.

5.2.1.1. Modelos a dois parâmetros

Os modelos a dois parâmetros forneceram respostas pobres no que diz respeito à representação dinâmica do sistema, o que já era previsto pela teoria exposta por Aström e Hägglund (1995).

A partir dos dados coletados foi feita a interpolação polinomial e assim, adquirido o polinômio da curva de resposta para a obtenção da área sobre ela, como é proposto no item 4.2.1.1.

Utilizando o modelo *Residence Time Approximation* verificou-se boa resposta para o regime permanente em relação ao sistema, no entanto o modelo correspondente ao integrador de tempo morto não representou satisfatoriamente ao transitório do sistema, assim sua apresentação não se tornou relevante.

5.2.1.2. Modelos a três parâmetros

Entre as funções de transferência baseadas em três parâmetros, o modelo que mais se aproximou da curva real foi o modelo que utiliza o tempo em que a resposta atinge 63% do seu valor de estabilização, que, para um sistema de primeira ordem, esse valor corresponde à constante de tempo do sistema.

Isso pode ser explicado pela análise da transformada inversa de Laplace da resposta ao degrau.

Sendo a função de transferência de um sistema de primeira ordem dado por

$$H(s) = \frac{K}{1 + s \tau} \quad (5.1)$$

E a transformada de Laplace do sinal degrau unitário $1/s$, então, a resposta do sistema no domínio do tempo corresponde a

$$y(t) = \mathcal{L}^{-1} \left\{ \frac{H(s)}{s} \right\} = K(1 - e^{-t/\tau}) \quad (5.2)$$

Substituindo-se t por τ na equação acima, observa-se que a saída corresponde a 63.2% do valor de estabilização K .

Uma função de transferência de primeira ordem pôde ser utilizada na representação deste sistema, mesmo sabendo-se que este possui pelo menos dois polos, um mecânico e um elétrico, pois o polo mecânico é mais lento e faz com que a sua resposta seja dominante no sistema. Com este tipo de identificação não é possível levantar todos os polos do sistema, pois o método aproxima o sistema para um modelo de primeira ordem, e, além disso, o sinal degrau utilizado excita apenas uma pequena faixa de frequências.

Por esta mesma razão, não foi possível chegar a uma representação do sistema utilizando o modelo de segunda ordem a três parâmetros citado neste trabalho representado pela equação 4.6, pois possuía dois polos iguais, o que não corresponde a este sistema.

Para a validação dos modelos, foi aplicado a cada modelo obtido um degrau de 12,43 V, a mesma amplitude utilizada na obtenção dos dados de validação. A

resposta foi então comparada aos dados do sistema real em resposta a esta mesma excitação.

Na Figura 18 está exibida a resposta da função de transferência obtida utilizando a variação de tempo entre A e C como constante de tempo representada pela equação 4.5. Ela ilustra a afirmação de que adotar este valor, obviamente, superestimaria este parâmetro, tornando a resposta do sistema mais lenta.

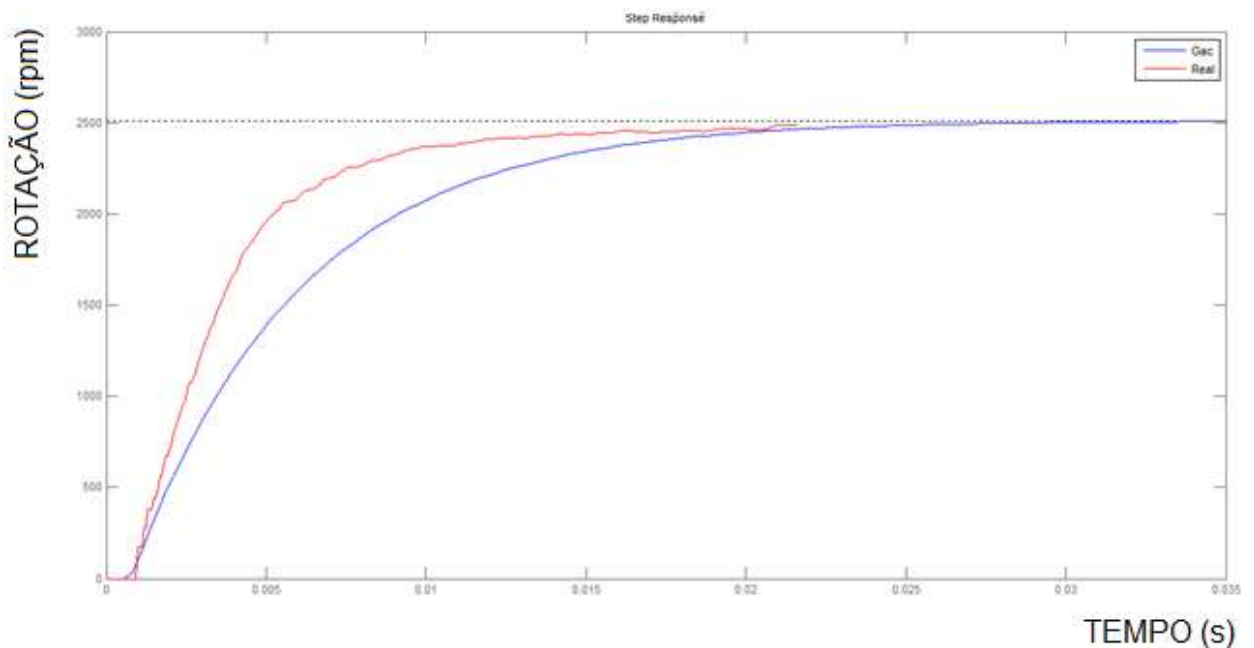


Figura 18 - Curva real de resposta ao degrau (vermelho) e resposta da função de transferência utilizando constante de tempo A-C (azul)
Fonte: Autoria Própria

A Figura 19 exibe a resposta ao degrau do modelo obtido utilizando variação de tempo entre A e B, ou seja, aproximadamente uma constante de tempo. Facilmente percebe-se que o modelo representou melhor a dinâmica do sistema.

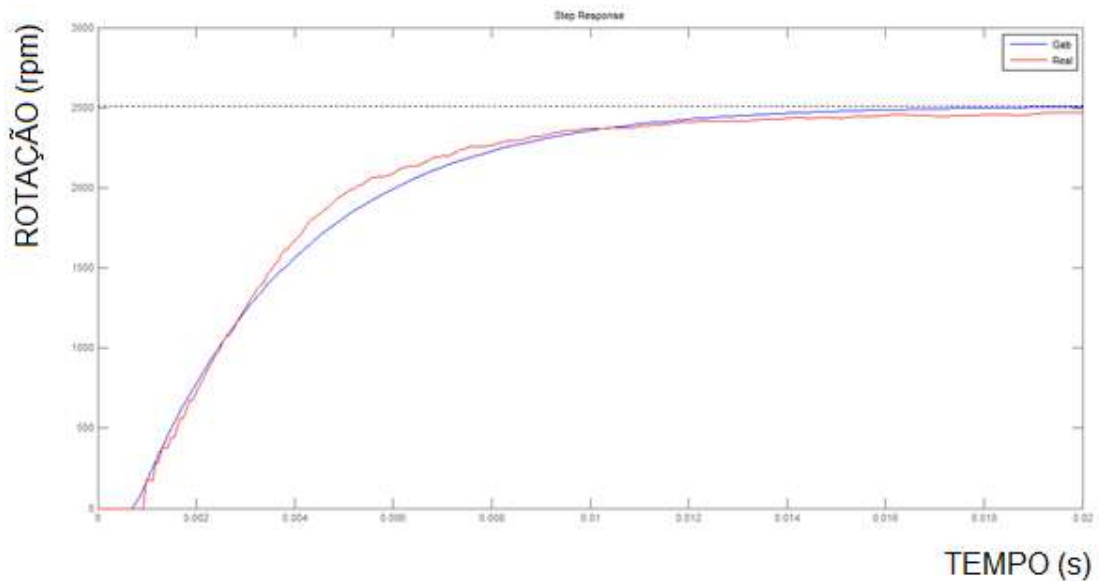


Figura 19 - Curva real de resposta ao degrau (vermelho) e resposta da função de transferência utilizando constante de tempo A-B (azul)
Fonte: Autoria Própria

Outra observação pode ser feita, a existência de um erro em regime estacionário presente nos modelos. Na Figura 19 isso pode ser observado mais facilmente. Este tipo de imprecisão presente nesses modelos pode ser consequência de algumas questões inerentes ao processo de identificação, como o ruído proveniente do próprio sistema e de fontes externas ou a falta de resolução na medição do sinal de saída do sistema.

Na tentativa de melhorar os modelos aumentando a resolução da leitura da saída, a forma de aquisição de velocidade foi alterada para a que utiliza um *timer* operando em modo “disparo”, contando o número bordas de pulsos de *clock* entre duas bordas consecutivas geradas pelo *encoder*.

Como pode ser visto na Figura 20, houve uma piora. Esse resultado é explicado pela necessidade de se traçar uma reta tangente ao ponto de inflexão da curva ditado pelo método de Ziegler-Nichols, de onde são extraídos os parâmetros da identificação. Nesta configuração de *timer* existe uma variação na taxa de atualização da velocidade, tornando-se interessante sua aplicação em velocidades mais altas, como será o caso da identificação por métodos estocásticos que será visto posteriormente. Em velocidades baixas a resolução se torna pior por levar mais tempo entre as bordas do sinal produzido pelo *encoder*. Dessa forma, não

houve resolução suficiente para obter o ponto de inflexão de maneira satisfatória e a tangente foi traçada de forma inadequada.

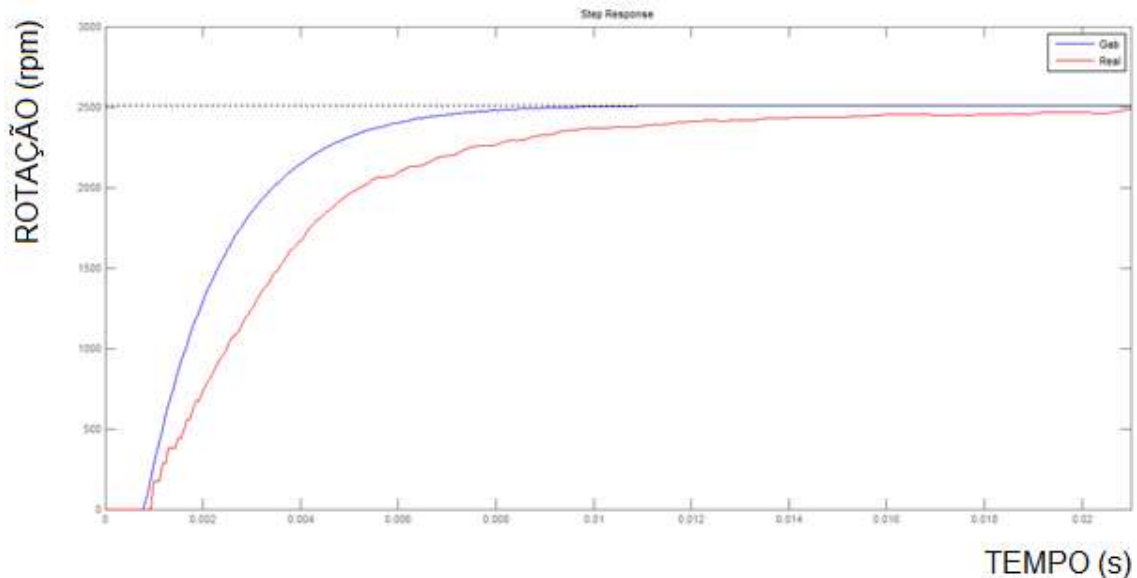


Figura 20 - Identificação com *timer* em modo de disparo
Fonte: Autoria Própria

Outro tipo de teste foi então proposto para encontrar, finalmente, a melhor curva dentro do método. Neste caso, o degrau não foi aplicado ao sistema em repouso, mas em seu estado estacionário respondendo a um degrau de outra amplitude. Desta forma, com o motor já funcionando a uma velocidade que fornece uma boa resolução, elimina-se o problema na obtenção do ponto de inflexão e da reta tangente. Está resposta está representada através da Figura 21.

Outra forma de se validar um modelo matemático é verificar se ele atende às necessidades, por exemplo, de se realizar um controle sobre o sistema. Com isso, foi desenvolvido um controlador PI, com base na teoria de Ziegler-Nichols, para averiguar a empregabilidade do modelo. Os valores propostos do controlador estão identificados através da Figura 22 e a função de transferência do controlador está representada pela equação 5.3.

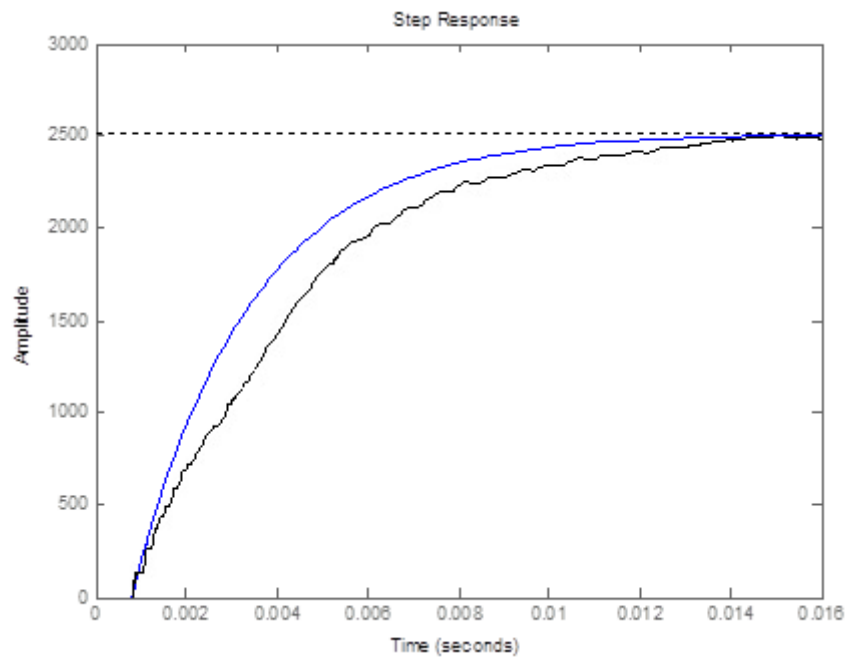


Figura 21 - Resposta ao degrau com variação de 46% a 78% de razão cíclica e ajuste de *offset*
Fonte: Aatoria Própria

Controlador	K	T_i	T_d	T_p
P	$1/\alpha$			$4L$
PI	$0,9/\alpha$	$3L$		$5,7L$
PID	$1,2/\alpha$	$2L$	$L/2$	$3.4L$

Figura 22 - Parâmetros do controlador PID a partir do método de resposta ao degrau de Ziegler-Nichols
Fonte: Aatoria Própria

$$G(s) = K \left(1 + \frac{1}{sT_i} \right) \quad (5.3)$$

O sistema sem controle aplicado se comporta de maneira instável quando em malha fechada. Utilizando o controlador projetado, ele passou a ser estável, corrigiu pequenos distúrbios e o erro estacionário. Ou seja, o modelo é válido. Como pode ser observada na Figura 23, a resposta do modelo obtido foi muito semelhante à resposta do sistema, e o seu controle obteve desempenho desejado.

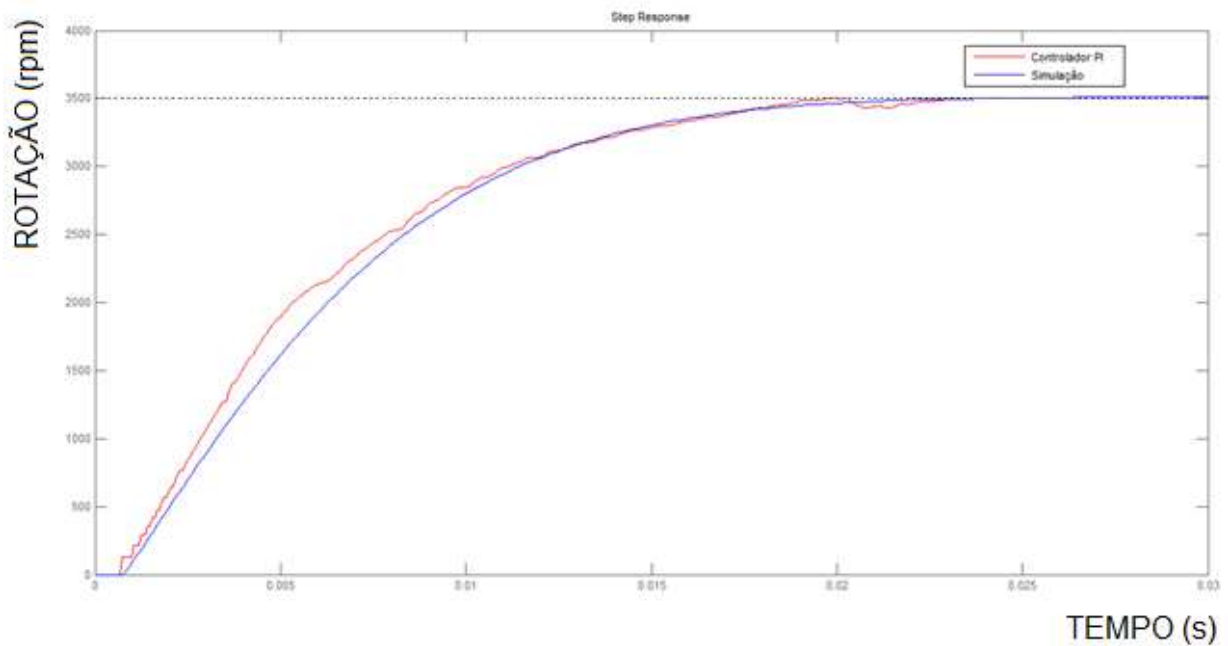


Figura 23 - Resposta do sistema em malha fechada com valor de referência de 3500 rpm
Fonte: Autoria Própria

5.2.2. Método Estocástico

Para o método estocástico, foi necessária a realização de alguns testes para a definição e produção do PRBS de entrada. A escolha das suas características é fundamental para a obtenção de um sinal adequado.

A primeira é o tempo de bit T_b . Sua escolha baseia-se no valor da constante de tempo do sistema. O mesmo degrau empregado na identificação determinística foi utilizado. A aproximação da constante de tempo para esse sistema difere da constante de tempo dos modelos de Ziegler-Nichols por ser obtida diretamente dos dados, e não de uma reta tangente ao ponto de inflexão, ou da área sobre a curva. Nestas condições, foi obtida uma constante de tempo igual a 3,28 ms.

Conhecendo-se o valor de τ , o T_b pode ser escolhido entre os limites abaixo, que é um resultado heurístico que normalmente fornece bons resultados (AGUIRRE, 2007, p.202).

$$\frac{\tau}{10} \leq T_b \leq \frac{\tau}{3} \quad (5.4)$$

É necessário considerar que para um valor selecionado muito alto, o sistema entenderá a excitação como um degrau, sinal pobre para o procedimento de identificação, e para um valor muito baixo não haverá tempo suficiente entre as transições para responder adequadamente. Em outras palavras, deve ser longo o suficiente para produzir uma boa resposta, e curto o suficiente para não permitir que o sistema entre em regime estacionário.

Foi adotado o valor de $T_b = \frac{\tau}{10}$ pois, segundo Aguirre (2007), para sistemas lineares é comum utilizar os valores mais baixos.

Outro parâmetro a ser escolhido é a amplitude do sinal a ser aplicado. A amplitude geralmente é limitada pela máxima excursão do sinal de entrada para uma mesma faixa linear (AGUIRRE, 2007, p.201). O sinal PRBS é normalmente de baixa amplitude, porém, deve ser levada em consideração a intensidade do ruído no sistema, pois se for muito baixa, o ruído pode se sobrepor ao sinal de interesse.

Nesta aplicação, foi adotada uma amplitude de 3,7 V para o PRBS, somada a um degrau de amplitude 14,4 V.

Para gerar o sinal determinado, podem ser realizadas operações de OU-exclusivo sobre alguns bits de uma dada sequência de m bits, esta função é também chamada de LFSR (AGUIRRE, 2007, p.199). Neste caso, o período o PRBS será de $T = N T_b$, em que $N = 2^m - 1$.

Alternativamente, pode ser gerada diretamente uma sequência de N bits no software MATLAB através da função *randn()*, que gera uma sequência aleatória de bits. Este foi o método adotado para este trabalho por permitir um maior controle sobre o período do sinal, o que influencia para um T maior. O número de *bits* na sequência gerada foi $N = 430$. Para um LFSR de 8 *bits*, o valor seria inferior, e para um de 9, o *buffer* seria saturado, justificando sua escolha.

Para medir o grau de correlação de uma variável em dado instante com ela mesma em um instante anterior é utilizada a função de autocorrelação. Em outras palavras: ela pode ser utilizada para determinar a aleatoriedade da mesma.

Sinais aleatórios ou pseudo-aleatórios possuem uma função de autocorrelação que se aproxima de um impulso unitário localizado na posição de deslocamento zero (AGUIRRE, 2007, p.193). O PRBS, como deve ser um sinal pseudo-aleatório, deve possuir uma autocorrelação deste tipo. A autocorrelação da sequência utilizada na identificação está apresentada na Figura 24.

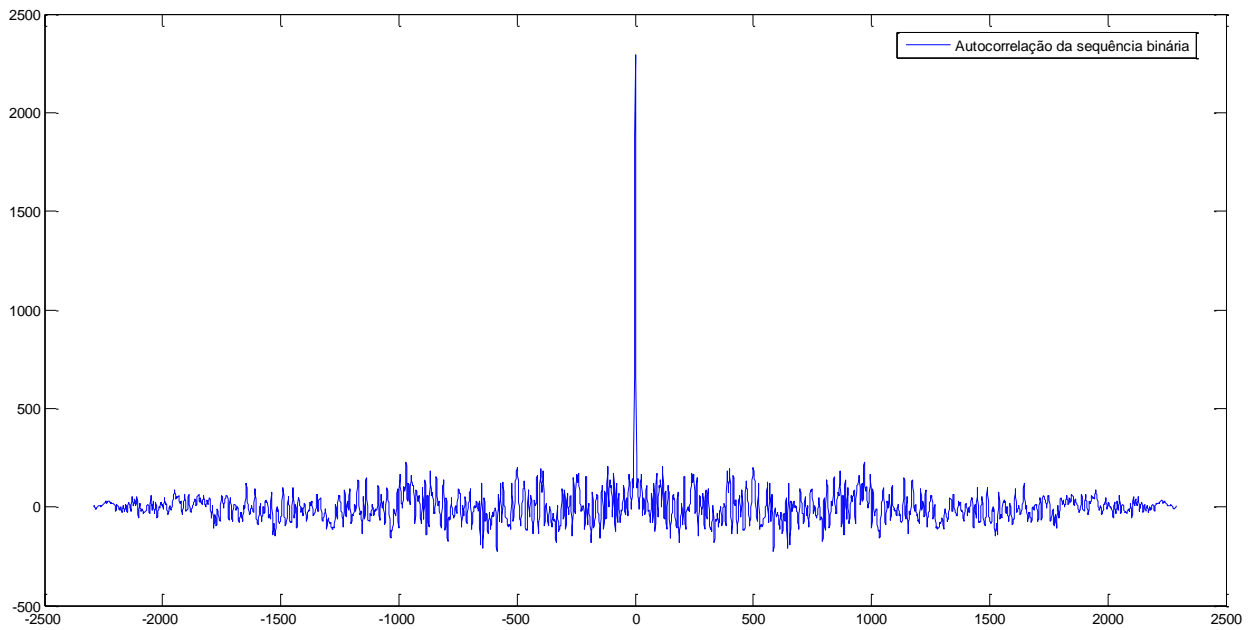


Figura 24 - Autocorrelação da sequência binária
Fonte: Autoria Própria

A qualidade do sinal produzido também foi avaliada a partir do periodograma na Figura 25. O espectro do sinal de entrada é aproximadamente plano, característica que aproxima-o de um ruído branco, que possui um espectro plano ao longo de toda a faixa de frequências.

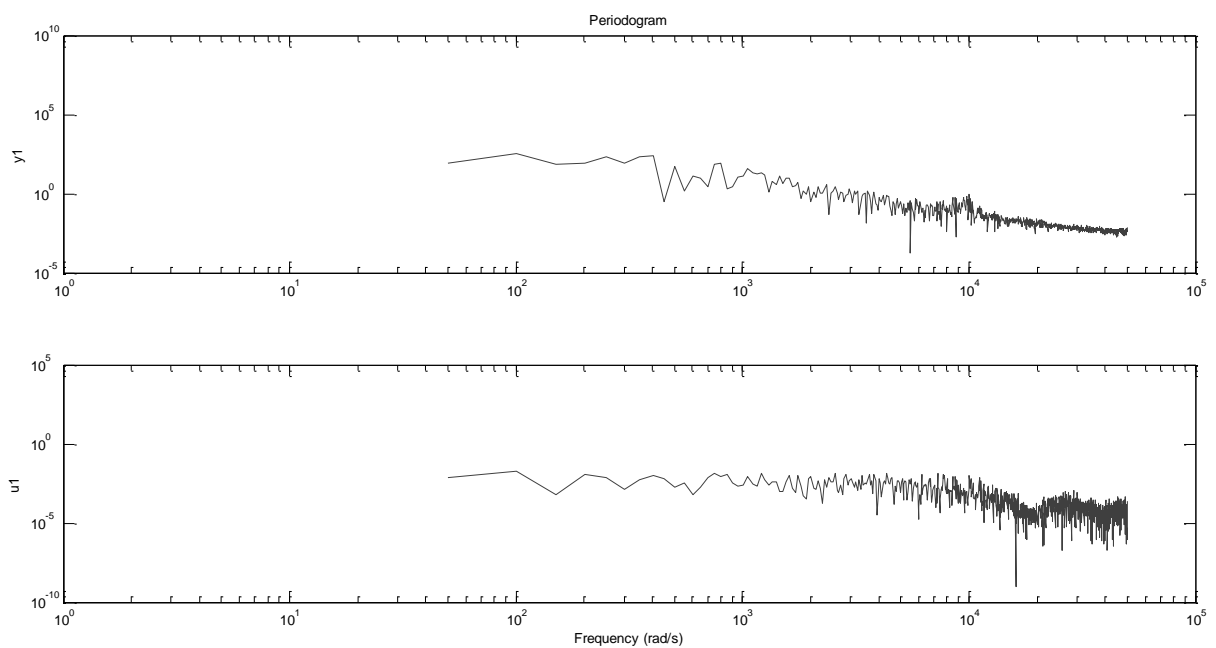


Figura 25 - Periodograma dos sinais de saída (y_1) e entrada (u_1)
Fonte: Autoria Própria

A resposta do sinal produzido a ser utilizada na identificação estocástica através da ferramenta do MATLAB está exibida na Figura 26, bem como a faixa do sinal que servirá para a modelagem e a faixa que será utilizada para validação.

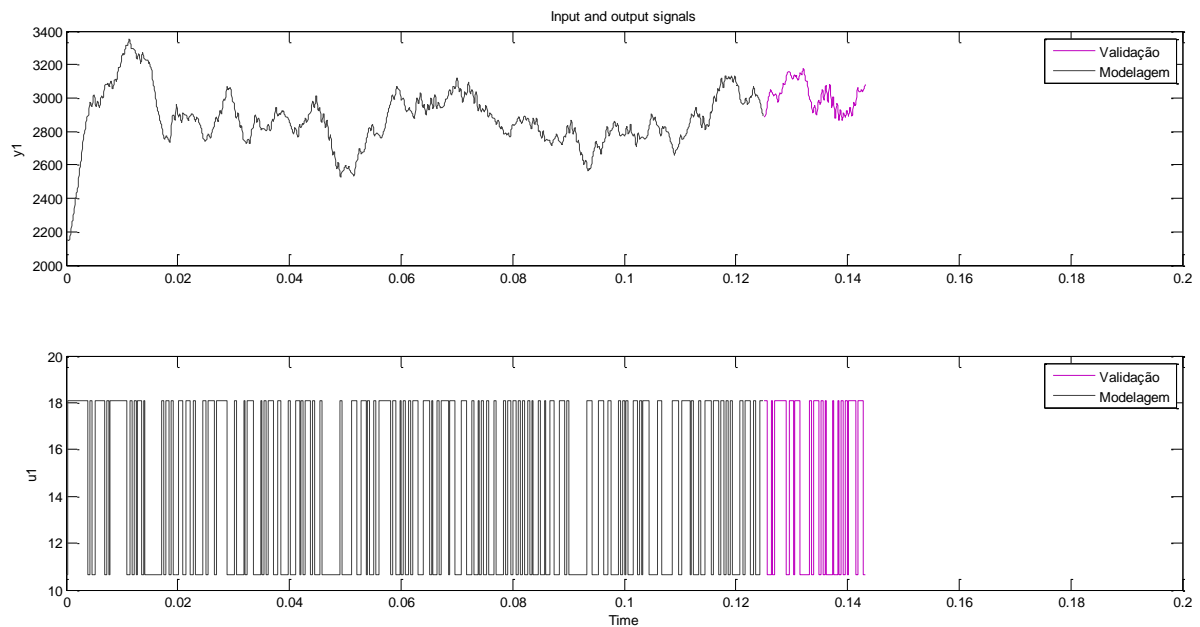


Figura 26 - Sinal de saída (y_1) e sinal de entrada (u_1)
Fonte: Autoria Própria

5.2.2.1. Modelagem no MATLAB

Com os dados levantados, passa-se à segunda etapa do processo de identificação de sistemas, que consiste na escolha das representações matemáticas a serem utilizadas. Foi realizados testes em cima dos modelos citados em 4.2.2, o único modelo que não será abordado será o do tipo ARMA, pois não se adequa ao sistema em estudo por tratar a saída apenas como uma função do ruído.

Para cada modelo é necessário determinar a sua estrutura. No caso dos tipos abordados, isso equivale a dizer que é preciso selecionar a ordem dos polinômios. Existem algumas formas de realizar uma estimativa destes valores, porém, o método que surtiu resultados mais satisfatórios foi o de tentativa e erro dentro do ambiente de identificação do MATLAB.

O método de tentativa e erro consiste em arbitrar as ordens dos polinômios, obter o modelo pela ferramenta, e fazer a sua verificação.

A primeira verificação a ser realizada é a da análise dos resíduos. Para cada modelo é feita uma autocorrelação da parte do sinal modelada como a variável aleatória $v(k)$. Se esta apresentar uma autocorrelação fora dos limites de confiança, isso significa que o ruído não é aleatório, portanto o modelo é falho.

Outra verificação é a análise da correlação cruzada dos resíduos da saída $v(k)$ e a entrada. Ela expressa se há relação temporal entre o ruído e a entrada, desta forma pode-se verificar se a ordem adotada no modelo está correta. Pois uma correlação da entrada com o resíduo do sistema indica que o modelo não capturou a dinâmica do sistema de maneira completa.

Para avaliar se a ordem escolhida foi exagerada são observados os polos e os zeros. Se houver cancelamento entre eles significa que os valores foram superestimados.

Há outros resultados que podem ser analisados como a resposta ao degrau e a resposta ao sinal de entrada de validação.

O melhor modelo dentre os obtidos foi o ARMAX, com os polinômios mostrados nas equações 5.5, 5.6 e 5.7.

$$A(q) = 1 - 3.479 q^{-1} + 4.825 q^{-2} - 3.172 q^{-3} + 0.827 q^{-4} \quad (5.5)$$

$$B(q) = 0.2515 q^{-6} - 0.2472 q^{-7} + 0.08838 q^{-8} \quad (5.6)$$

$$C(q) = 1 - 1.909 q^{-1} + 1.343 q^{-2} - 0.3682 q^{-3} \quad (5.7)$$

A autocorrelação do resíduo e a sua correlação cruzada com a entrada se mostraram dentro dos limites estabelecidos, como pode ser observado na Figura 27.

Nesta análise foi utilizado um nível de confiança de 99% ou 2,5758 sigmas.

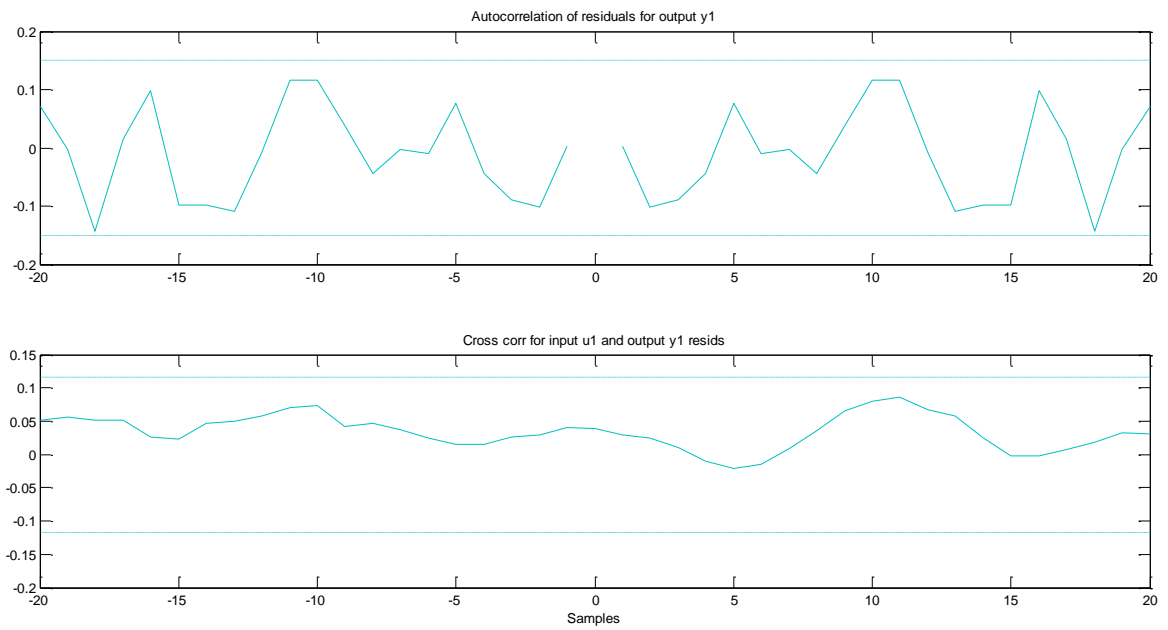


Figura 27 - Análise dos resíduos
Fonte: Autoria Própria

Nenhum polo e zero do modelo ficaram muito próximos ou se cancelaram como pode se observar na Figura 28.

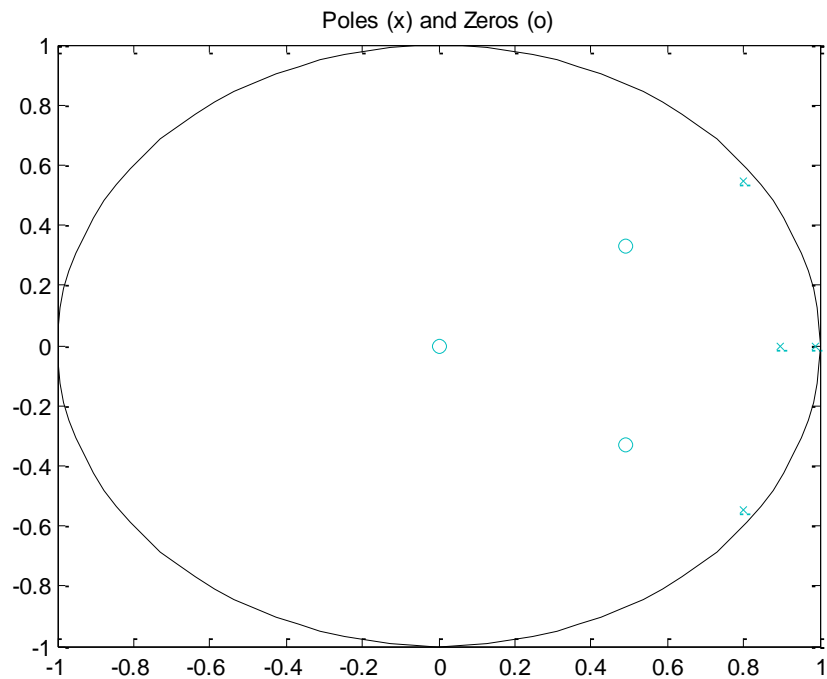


Figura 28 - Polos e zeros do sistema
Fonte: Autoria Própria

A Figura 29 apresenta a resposta real do sistema a uma faixa do sinal de entrada com PRBS e a saída do modelo para a mesma excitação. É possível visualizar que o modelo obtido conseguiu capturar bem a dinâmica do sistema.

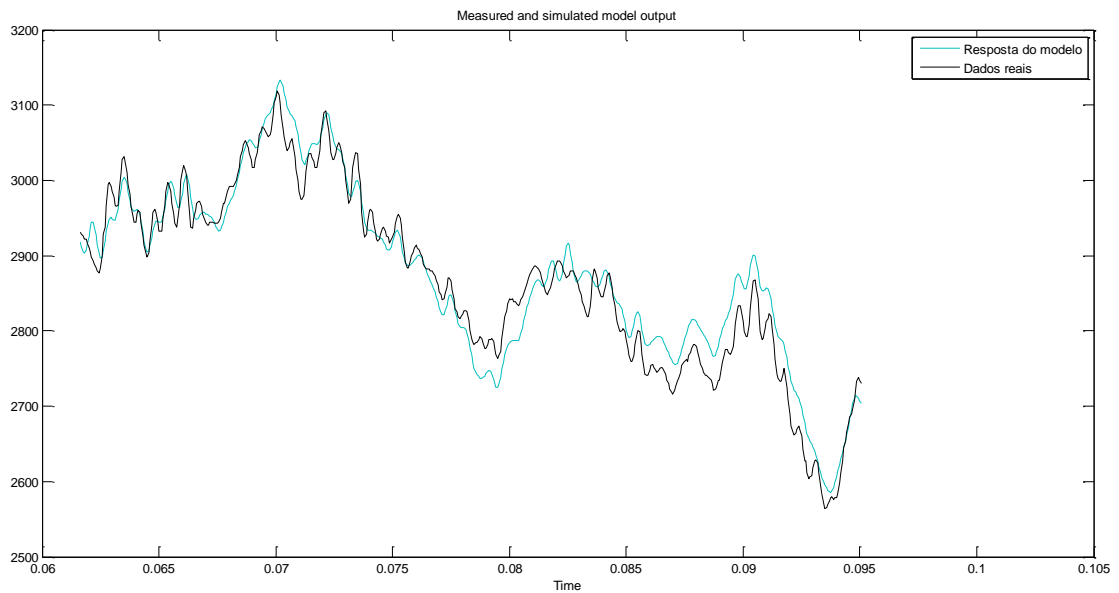


Figura 29 - Resposta do sistema e do modelo a uma entrada PRBS
Fonte: Autoria Própria

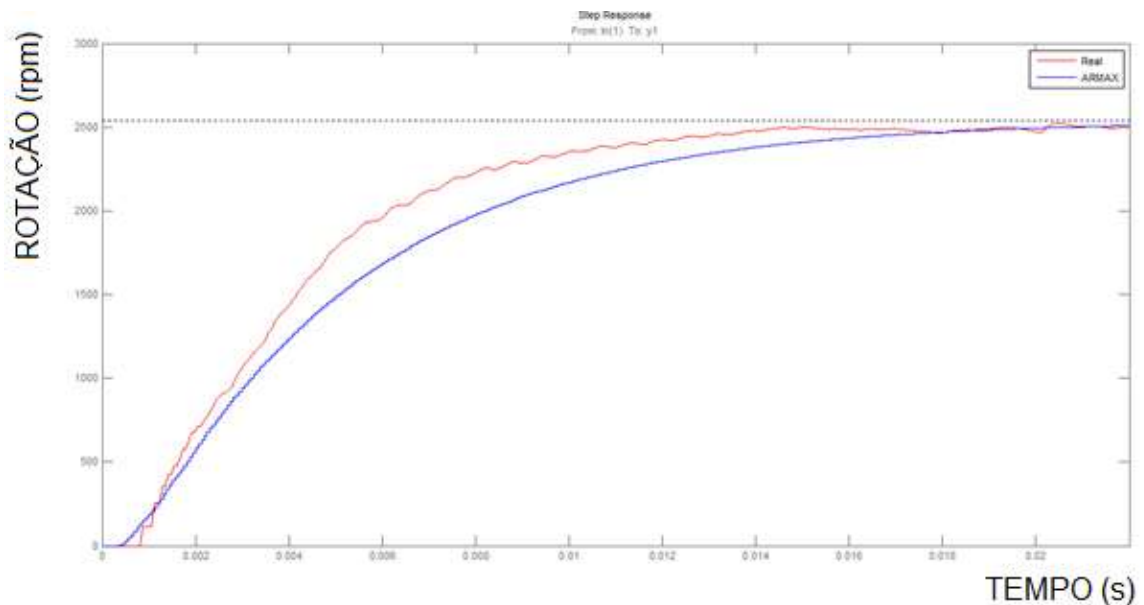


Figura 30 - Resposta a um degrau de 12,4 V
Fonte: Autoria Própria

A Figura 30 mostra a resposta do modelo e do sistema real a um degrau de 12,4 V. É importante lembrar que a obtenção deste modelo levou em consideração a

presença do ruído no sistema, o que faz com que sua resposta apresente diferenças quando comparada aos dados reais que possuem uma parcela aleatória.

A fim de comprovar as vantagens da utilização do método estocástico optou-se pela realização de um novo teste. O teste consiste em somar um ruído gaussiano aleatório à saída do sistema antes de realizar a identificação, situação que pode ocorrer na prática.

O ruído foi produzido no MATLAB utilizando novamente a função *randn()*. Para verificar sua aleatoriedade foi aplicada a função de autocorrelação sobre ele conforme a Figura 31.

O sinal original está apresentado na Figura 32.

O sinal somado ao ruído pode ser observado na Figura 33. O sinal ficou visivelmente distorcido. Este é o sinal que será utilizado para a identificação estocástica.

Um ruído desta ordem tornaria inviável a identificação determinística realizada neste trabalho, pois a resposta ao degrau ficaria muito corrompida, atrapalhando na obtenção dos parâmetros necessários, como exibido na Figura 34.

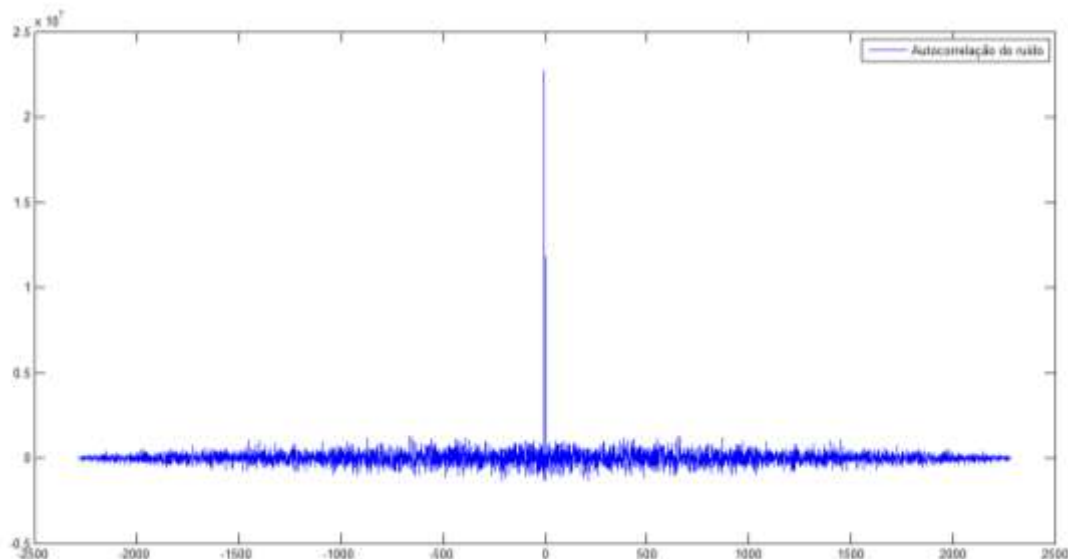


Figura 31 - Autocorrelação do ruído
Fonte: Autoria Própria

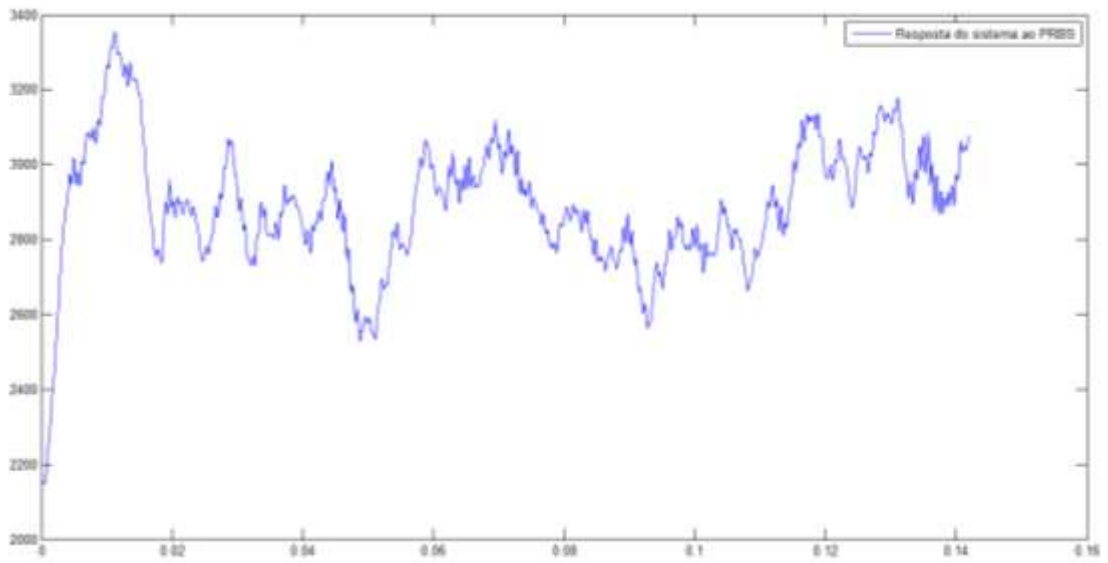


Figura 32 - Sinal original
Fonte: Autoria Própria

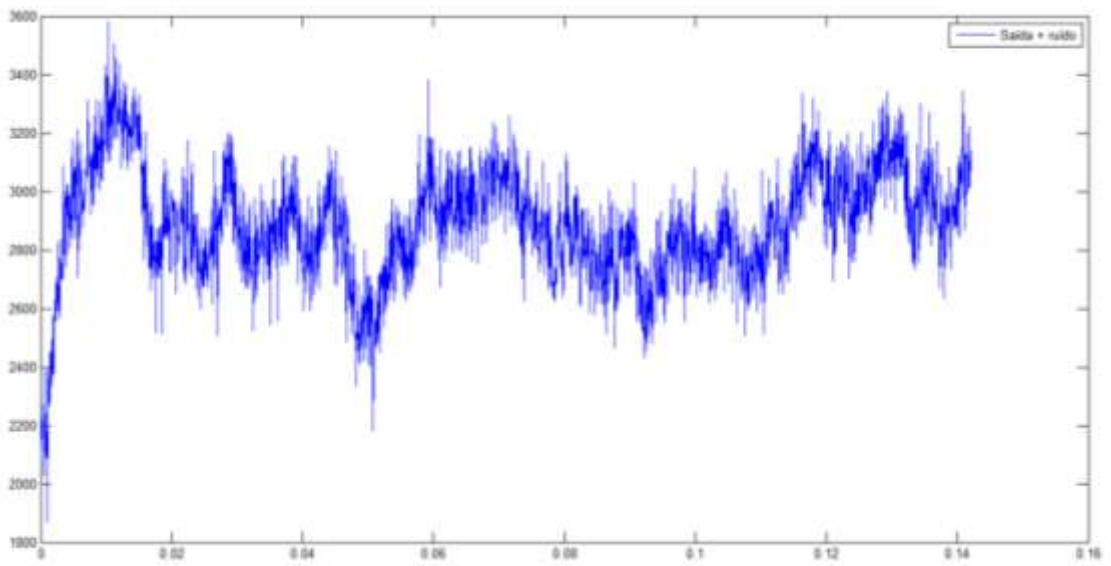


Figura 33 - Sinal com ruído aditivo
Fonte: Autoria Própria

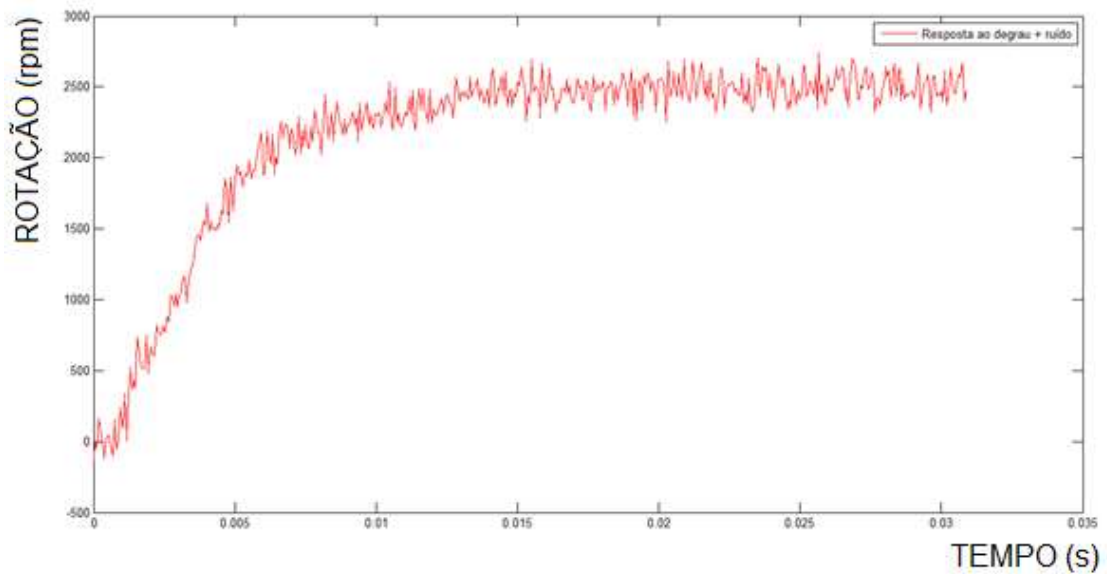


Figura 34 - Resposta ao degrau com ruído aditivo
Fonte: Autoria Própria

Utilizando os dados preparados, iniciou-se novamente o processo de identificação.

O modelo que melhor representou o modelo, satisfazendo todos os critérios de identificação, foi novamente o ARMAX com os polinômios das equações 5.8, 5.9 e 5.10. Sua resposta ao degrau comparada com um valor real está apresentada na Figura 35.

$$A(q) = 1 - 1.874 q^{-1} + 0.8701 q^{-2} - 0.01932 q^{-3} + 0.0715 q^{-4} - 0.04683 q^{-5} \quad (5.8)$$

$$B(q) = 0.5289 q^{-5} - 0.1906 q^{-6} \quad (5.9)$$

$$C(q) = 1 - 1.83 q^{-1} + 0.8388 q^{-2} \quad (5.10)$$

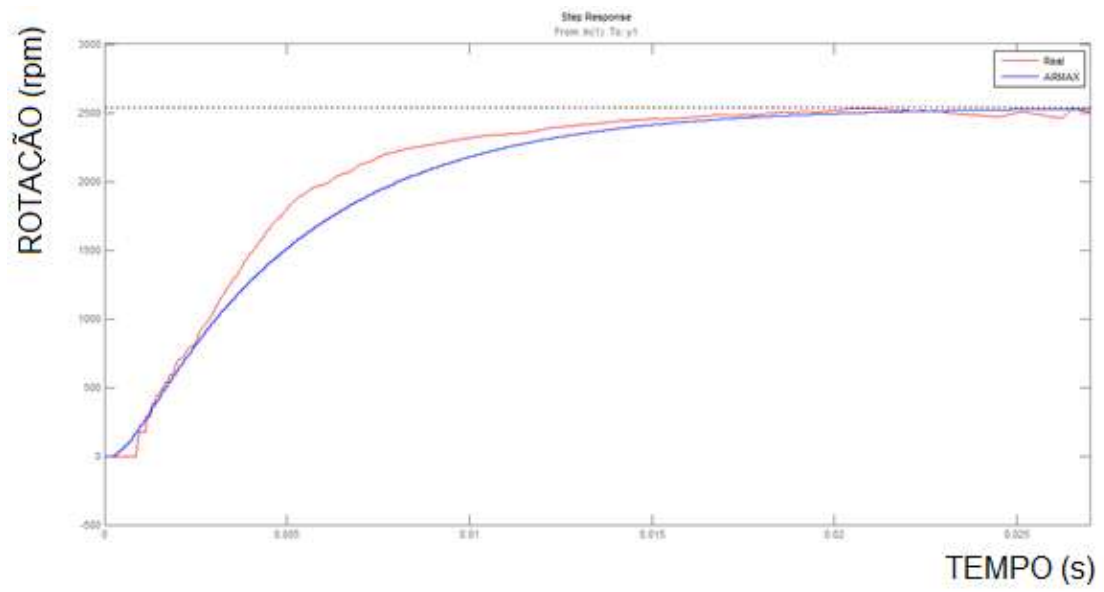


Figura 35 - Resposta real e do modelo a um degrau de 12,4 V
Fonte: Autoria Própria

6. CONSIDERAÇÕES FINAIS

De acordo com a proposta do trabalho verificou-se a relevância da utilização dos motores síncronos de ímãs permanentes no rotor, em relação às suas características e desempenho. Junto a isto, com o processamento de dados e controle é possível obter um bom desempenho utilizando um controlador digital de sinais, de que foram citadas todas as suas características, ressaltando seus pontos que contribuem para o desenvolvimento.

A obtenção de modelos matemáticos que reproduzam a dinâmica de um determinado sistema é uma tarefa de importância capital em Engenharia. Uma abordagem possível para a construção desses modelos é a identificação de sistemas, que permite realizar o processo de modelagem a partir de dados medidos de entrada e saída, sem necessitar da obtenção de parâmetros do sistema, ou do equacionamento dos fenômenos envolvidos.

No âmbito da identificação, é possível dividir a modelagem em métodos que considerem ou não a presença de ruído no modelo. Dentre os métodos utilizados, pode-se citar algumas vantagens e desvantagens, porém ambos se mostraram eficientes para o sistema em questão.

Toda a elaboração do presente trabalho tem o intuito de obter os parâmetros necessários para o levantamento de um modelo de forma rápida e simples em comparação aos métodos de modelagem pela física, e que, ainda assim forneçam uma representação adequada, que viabilize o desenvolvimento de um algoritmo de controle efetivo.

O método determinístico utilizado apresentou-se rápido e simples, desde o sinal de excitação a ser aplicado até a obtenção dos parâmetros. Porém, é um método bastante restritivo por ser muito sensível ao ruído, o que impediria a obtenção gráfica dos parâmetros, e também à ordem do sistema, pois o procedimento utilizado se aplica apenas a sistemas de primeira ordem ou que a isto possam ser aproximados.

No entanto, para o modelo determinístico, quando se trata do controlador para o sistema, este pode ser projetado de maneira imediata a partir dos mesmos parâmetros encontrados no processo de identificação através do método de Ziegler-Nichols.

O método estocástico é mais complexo, no entanto, pode ser utilizado em dados muito corrompidos por ruído. Outra característica interessante é que o sinal de excitação escolhido pode ser visto como uma pequena perturbação no sinal usual de entrada de um sistema. Desta forma, a identificação pode ser realizada durante a própria operação, com poucas influências sobre o modo de operação normal do processo.

Os pontos negativos percebidos na identificação estocástica são a necessidade de um *hardware* com mais recursos que permita um menor tempo de amostragem e uma leitura com maior resolução da velocidade do motor. Além disso, é necessário o conhecimento sobre a teoria envolvida em todas as etapas do processo e sobre a ferramenta a ser utilizada, e, por fim, pode-se citar o modelo obtido, que é mais complexo que o encontrado no método determinístico, fazendo-se necessário um maior conhecimento a respeito de projeto de controladores.

Como resultado deste desenvolvimento, pôde-se concluir que a escolha de um método de identificação não implica em utilizar um procedimento que seja melhor que o outro, mas sim na decisão por um que se aplique melhor às suas condições. Trata-se de uma solução de compromisso entre as possibilidades, ponderando tempo, dificuldades, condições de trabalho, *hardware* disponível, entre outros.

REFERÊNCIAS

- AFONSO, Amintas P., **Modelagem matemática**. Disponível em: <<http://www.matematiques.com.br/download.php?tabela=documentos&id=275>>. Acesso em: 02 de agosto de 2013.
- AGUIRRE, Luis A. **Introdução à identificação de sistemas**. Belo Horizonte: UFMG, 2007.
- ÅSTRÖM, Karl J.; HÄGGLUND, Tore. **Advanced PID Control**. United States of America: Instrument Society of America, 2006.
- ÅSTRÖM, Karl J.; HÄGGLUND, Tore. **PID Controllers: Theory, Design, and Tuning**. 2. ed. United States of America: Instrument Society of America, 1995.
- BELO HORIZONTE, Vinicius C. **Planejamento de Perturbações para a Identificação de Modelos Dinâmicos**. 2004. 110 f. Dissertação (Mestre em Engenharia Química) – Programa de Pós-Graduação em Engenharia Química, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2004.
- BEZERRA, L. B. **Implementação de uma bancada para o controle eletrônico de motores síncronos de ímã permanente utilizando um DSP**. Tese (Doutorado). 2004.
- FERNANDES, Fabrício S. **Identificação por predição de erro e síntese de controladores robustos**. 2006. 108 f. Dissertação (Mestre em Engenharia Elétrica) - Colegiado do Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, 2006.
- FREESCALE. **Controle de motores BLDC com tecnologia da Freescale**. 2006. Disponível em: <http://www.freescale.com/webapp/sps/site/training_information.jsp?code=WBNR_LA_MOTORCTRL#>. Acesso em: 26 de janeiro de 2013.
- FREESCALE. **MC56F825x/MC56F824x Digital Signal Controller**. Folha de especificação de dados. 2010.
- FREESCALE. **Three-Phase BLDC Motor Control Using MC56f8257**. Guia do usuário. 2011.
- FREESCALE. **Three-Phase BLDC Motor Control with Hall Sensors Using 56800/E Digital Signal Controllers**. Nota de aplicação. 2005.
- FREESCALE. **Three-Phase Fild Effect Transistor Pre-driver**. Guia do usuário. 2012.

MACHADO, Vinicius C. **Planejamento de Perturbações para Identificação de Modelos Dinâmicos**. 2004. 110 f. Dissertação (Mestre em Engenharia) - Programa de Pós-Graduação em Engenharia Química, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2004.

MONTEIRO, J. R. B. **Estratégias de Acionamento e Controle em Máquinas CA de Imã Permanente com Fluxo não Senoidal**. Dissertação (Mestrado). 2006.

NOLL, Valdir. **Desenvolvimento de um Servodrives de Arquitetura Aberta para Controle e Acionamento de Servomotores de Corrente Contínua sem Escovas**. Florianópolis, v. 2, n. 1, p.64-70, 2010.

PIRES, W. L. **Motores de indução alimentados por conversores de frequência PWM**. Guia Técnico, 2006.

SEM, P. C. **Principles of electric machines and power electronics**. United States of America: John Wiley & Sons, 1997.

TRINDADE, Rafael H. **Estudo de Máquinas Elétricas Não-Convencionais: Motor Brushless DC**. 2009. 30 f. Trabalho de Conclusão de Curso – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2009.

WEG. **Inversores de frequência**. Guia de Aplicação. 2004.

XXVIII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 2008, Rio de Janeiro. **Desenvolvimento de Metodologia para Parametrização em Tempo Real de Modelos Matemáticos de Processos Decisórios**. Rio de Janeiro: UFBA, 2008. Disponível em: <http://www.abepro.org.br/biblioteca/enegep2008_TN_STO_074_527_12126.pdf>. Acesso em: 07 de agosto de 2013.

YADAMALE, P. **Brushless DC (BLDC) Motor Fundamentals**. Notas de Aplicação. 2003.

APÊNCIE A – TUTORIAL

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS DE CURITIBA
CURSO DE ENGENHARIA INDUSTRIAL ELÉTRICA**

ANA LÚCIA VIEIRA IAREMCZUK

JULIO CESAR MOURA FAÉ

**MATERIAL DE APOIO PARA UTILIZAÇÃO DOS KITS TWR-
56F8257 E TWR-MC-LV3PH NO PROCESSO DE
IDENTIFICAÇÃO DE SISTEMAS**

TUTORIAL

CURITIBA

2014

LISTA DE FIGURAS

Figura 1 - Placa de controle	3
Figura 2 - Placa de Potência	4
Figura 3 - Sistema para leitura de velocidade	5
Figura 4 - Circuito interno do encoder	6
Figura 5 - Ambiente de trabalho do Codewarrior	6
Figura 6 - Passos para importar um projeto	7
Figura 7 - Interface do Processor Expert	8
Figura 8 - Seleção de configurações no Processor Expert	8
Figura 9 - Configuração da porta de interface USB	9
Figura 10 - Escolha do projeto em análise	9
Figura 11 - Escolha das propriedades do Recorder	14
Figura 12 - Propriedades para exibição dos dados	15
Figura 13 - Configurações do trigger	15
Figura 14 - Gráfico exibido utilizando o Recorder	16
Figura 15 - Passo para obtenção dos dados amostrados	16
Figura 16 - Determinação do caminho para armazenamento dos dados	16
Figura 17 - Gráfico da resposta ao degrau com controle PI	17
Figura 18 - Propriedades do Scope	17
Figura 19 - Visualização da variável	18
Figura 20 - Setpoint	18
Figura 21 - Alteração do setpoint	19
Figura 22 - Comando para abrir a ferramenta de identificação	19
Figura 23 - Ambiente para identificação	20
Figura 24 - Importar dados do workspace	20
Figura 25 - Seleção de dados para importação	21
Figura 26 - Seleção dos dados a serem trabalhados	21
Figura 27 - Criação de novo conjunto a partir de faixa de dados	22
Figura 28 - Escolha dos dados para identificação e validação	22
Figura 29 - Abertura da janela pra estimação do modelo polinomial	23
Figura 30 - Janela para seleção do modelo	23
Figura 31 - Estimação do grau dos polinômios para o modelo ARX	24
Figura 32 - Análise dos resíduos	24
Figura 33 - Diagrama de polos e zeros	24
Figura 34 - Seleção de análises desejadas	25

SUMÁRIO

1. APRESENTAÇÃO.....	3
1.1. CONTROLADOR.....	3
1.2. PLACA DE POTÊNCIA.....	4
1.3. SISTEMA PARA AQUISIÇÃO DA VELOCIDADE	5
2. INICIALIZANDO	6
3. CONFIGURAÇÃO	7
3.1. PWM.....	10
3.2. DRIVER (MC33937)	10
3.3. TIMERA.....	11
3.4. TIMERB.....	11
4. PRINCIPAIS FUNÇÕES DO FIRMWARE.....	11
5. REALIZANDO TESTES NO FREEMASTER	13
6. IDENTIFICAÇÃO NO MATLAB	19

1. APRESENTAÇÃO

1.1. CONTROLADOR

A placa de controle (TWR-56F8257) possui alguns elementos básicos para seu funcionamento.

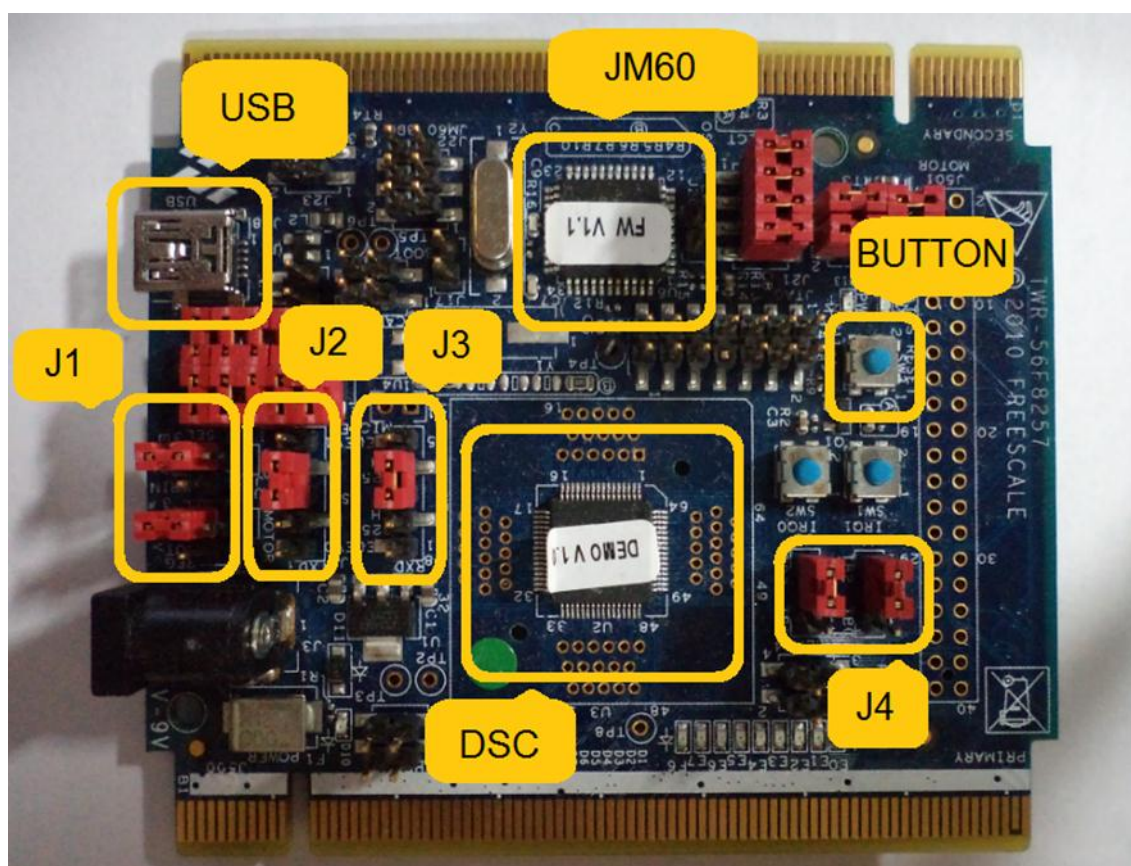


Figura 1 - Placa de controle

Para o estabelecimento da comunicação entre o software Freemaster e o controlador utilizado (DSC), é necessário um microcontrolador que realize a conversão serial (do DSC) para USB (PC). Este dispositivo é o JM60 que já está configurado para a realização desta tarefa. Também há necessidade da instalação de um driver que está disponibilizado pela Freescale.

Outra configuração necessária para o estabelecimento da comunicação é o posicionamento dos jumpers J2 e J3 como estão representados na imagem. O código fornecido foi configurado de acordo com a seleção de pinagem determinada por eles.

Os jumpers representados por J1 selecionam procedência da fonte de alimentação, que em nosso caso, é derivada de uma fonte externa de 24 V plugada na placa de potência, onde há reguladores lineares de tensão que fornecem a alimentação necessária para a estrutura de processamento. A alimentação é distribuída aos componentes da para a estrutura de processamento por meio da placa elevadora.

O push-button representado está ligado ao pino de reset do DSC, se tornando útil nos momentos em que há necessidade de retornar aos parâmetros iniciais utilizados no firmware.

Outra seleção realizada foi através dos jumpers J4. Estes selecionam as pinagens utilizadas para os push-buttons restantes.

Por fim, está destacado o mais importante componente: o DSC. Este componente é o responsável por todo o processamento dos dados e também pelas ações de que se desejam realizar.

1.2. PLACA DE POTÊNCIA

Para o acionamento do motor é necessária uma estrutura que seja capaz de fornecer tensão e corrente mínimas para o seu acionamento. A placa utilizada para funcionamento do motor foi a TWR-MC-LV3PH também produzida pela Freescale. Na Figura 2 estão representadas as partes básicas para o acionamento do motor.

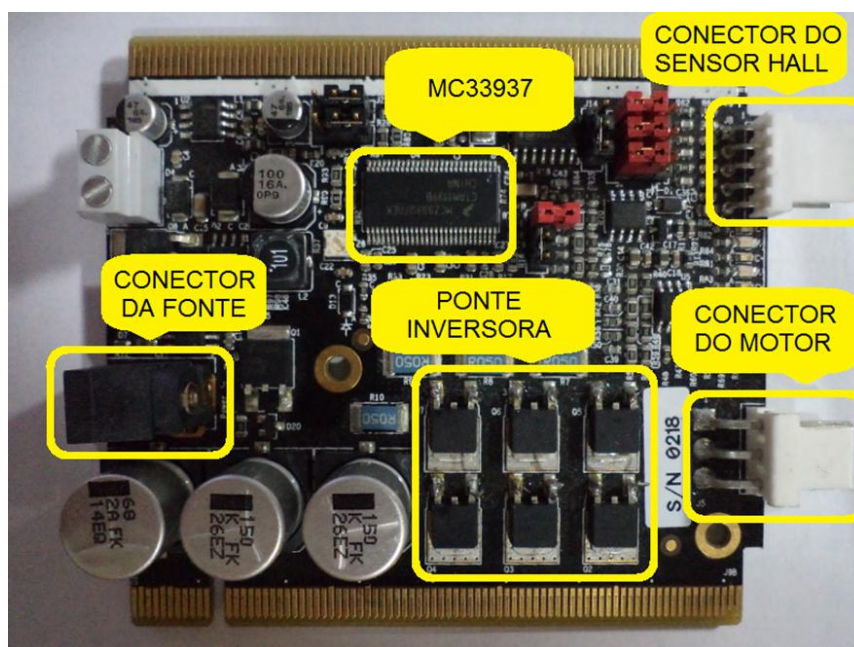


Figura 2 - Placa de Potência

A ponte inversora trifásica que está representada na imagem é constituída por seis MOSFETs IRF540. Com o chaveamento destes componentes é realizado o

acionamento do motor, fornecendo a tensão e corrente provenientes de uma fonte de tensão de 24 V e 5 A.

A placa possui também um conector para os sensores de efeito Hall, que monitoram a posição do rotor, e viabilizam a comutação das chaves para o acionamento do BLDC.

Para o acionamento da ponte inversora, é necessário um driver (MC33937), pois para a condução do MOSFET é necessária uma referência de tensão entre o gate e o source. Como os MOSFETs superiores possuem referências diferentes entre o gate e o dreno de cada fase, este driver é necessário para estabelecer cada uma das referências e assim, possibilitar o chaveamento.

1.3. SISTEMA PARA AQUISIÇÃO DA VELOCIDADE

Para a aquisição da velocidade foi utilizado um encoder incremental com resolução de 1000 pulsos por rotação e 4000 rpm de velocidade máxima devido à sua estrutura mecânica. Foi elaborado um suporte de acrílico, servindo para fixação do encoder e do motor e para o acoplamento de ambos através de uma peça de poliéster confeccionada para esta aplicação. Toda esta estrutura é representada na Figura 3.



Figura 3 - Sistema para leitura de velocidade

O circuito interno do encoder está representado pela Figura 4. Pode-se perceber que se fez necessário utilizar resistores de pull-up nas fases A, B e Z ligados a uma tensão de 3,3 V para conseguir medir valores de tensão. Com esta configuração foi possível realizar a captura dos sinais enviados pelo encoder diretamente pelo controlador.

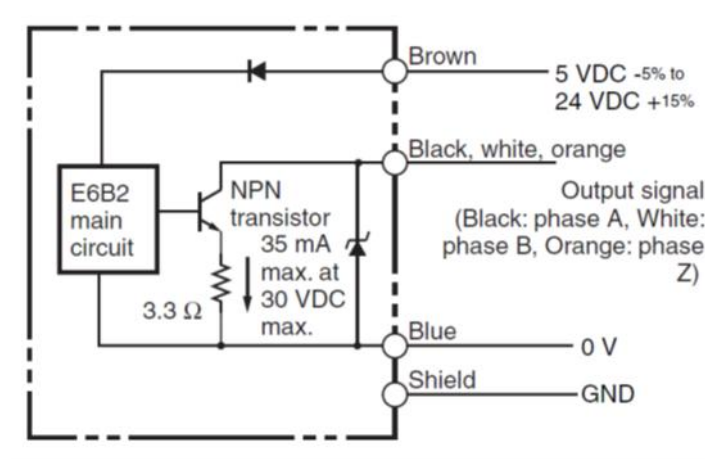


Figura 4 - Circuito interno do encoder

2. INICIALIZANDO

O CodeWarrior é um software de compilação desenvolvido pela Freescale, que engloba todas as suas famílias de componentes.

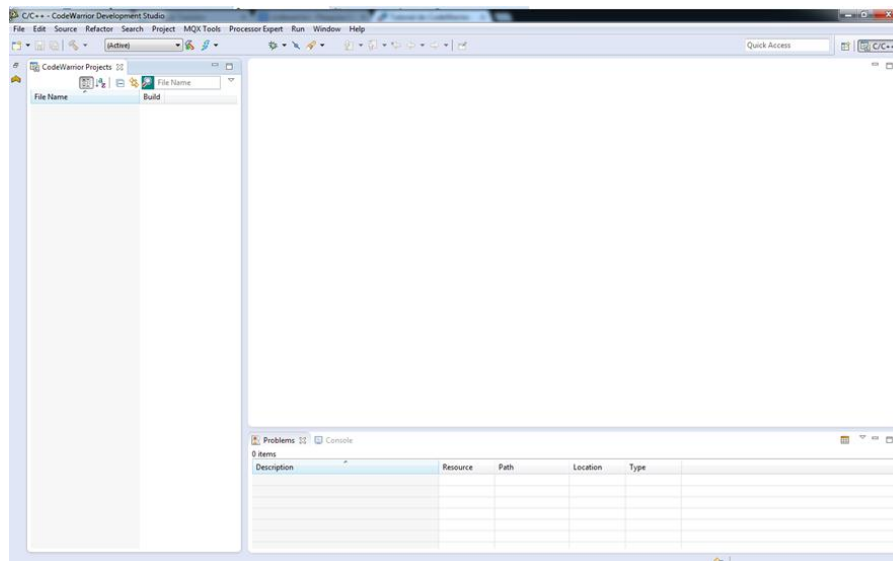


Figura 5 - Ambiente de trabalho do Codewarrior

Para iniciar o desenvolvimento de um trabalho há a necessidade de importar ou criar um novo projeto.

Como o intuito deste material é auxiliar na manipulação do kit desenvolvido, todos os procedimentos iniciais de configuração para o controlador já foram realizados. No código do projeto encontram-se as funções relacionadas ao

procedimento de identificação e controle juntamente a todas as demais funções necessárias à utilização do kit.

Com isso, é preciso apenas a importação do projeto já criado. Os passos deste procedimento estão representados na Figura 6.

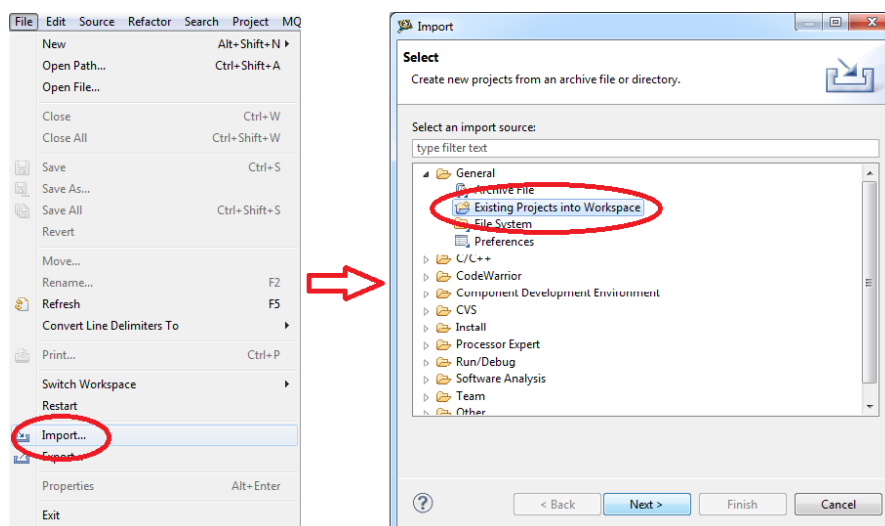


Figura 6 - Passos para importar um projeto

3. CONFIGURAÇÃO

1.1. FREEMASTER

A metodologia empregada na configuração dos registradores foi baseada na utilização de uma ferramenta do CodeWarrior: o Processor Expert. Um projeto foi criado separadamente apenas para gerar bibliotecas com todas as configurações necessárias a uma determinada aplicação através de uma interface gráfica mais amigável. Nele, todas as características determinadas a partir do estudo dos materiais de referência do dispositivo DSC podem ser selecionadas dentre as opções disponíveis.

Os códigos gerados com o auxílio desta ferramenta foram movidos para o contexto do firmware do projeto.

Outra ferramenta de grande utilidade da Freescale é o software Freemaster. Por intermédio dele é possível acompanhar em tempo real o estado das variáveis presentes no firmware através de comunicação serial. Há dois modos disponíveis: Scope e Recorder. Para este trabalho foi utilizado o modo Recorder, por fornecer uma maior variedade de configurações, que resultam em uma melhor resolução. As configurações que habilitam o uso deste recurso também podem ser configuradas via Processor Expert como demonstrado na Figura 7.

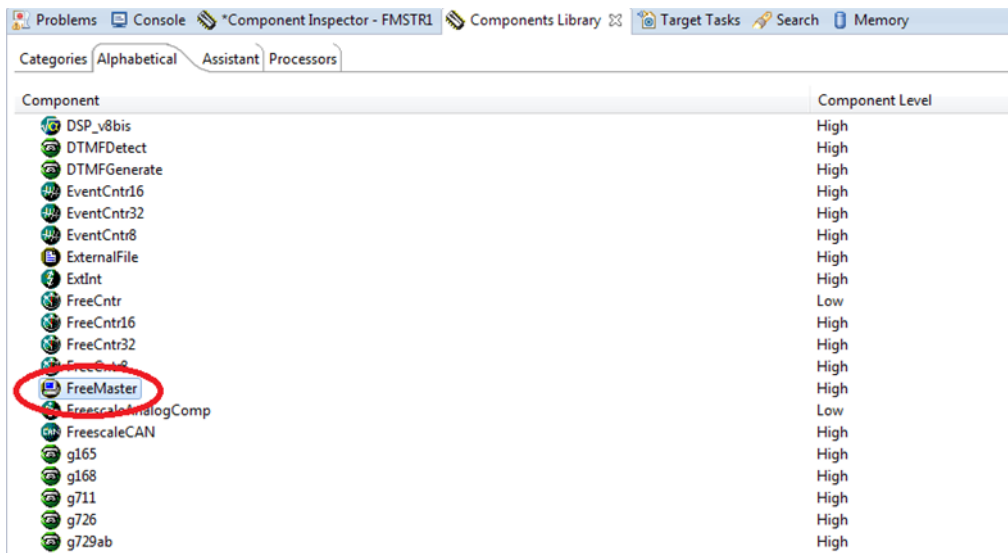


Figura 7 - Interface do Processor Expert

Deve-se configurar o FreeMaster habilitando a função Recorder (Figura 8) que será utilizada nas situações em que houver a necessidade de amostragens, incluindo esta função em interrupções periódicas para que haja uma amostragem das variáveis em um tempo definido. Em nosso caso será gerado uma interrupção no PWM.

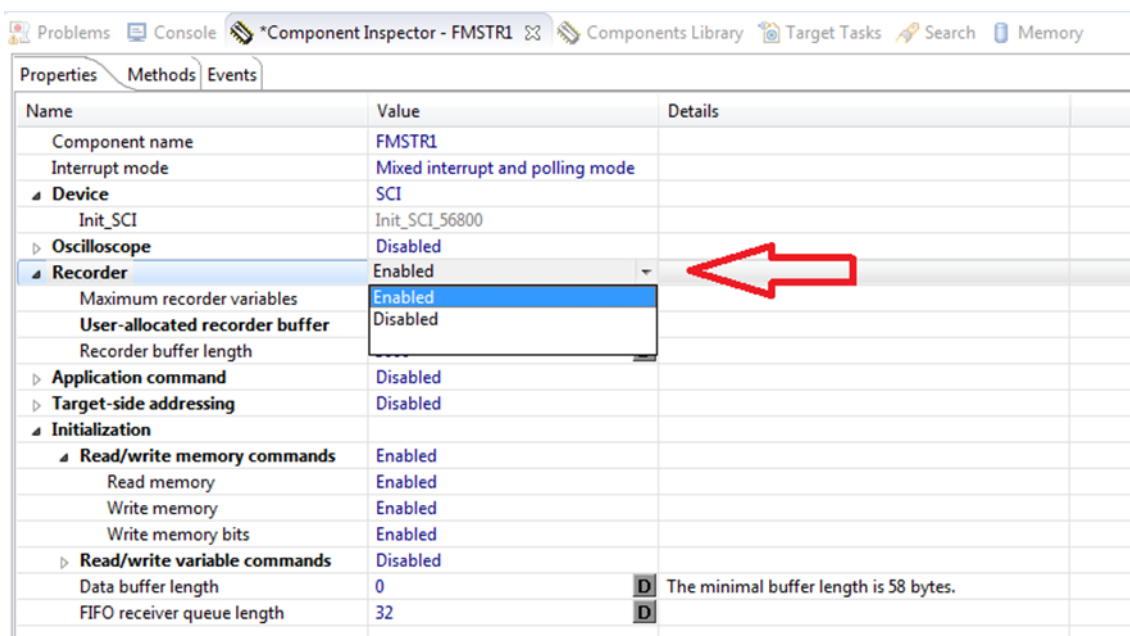


Figura 8 - Seleção de configurações no Processor Expert

Também se deve estipular o tamanho do Buffer (5 kB) que será utilizado para o armazenamento das variáveis, tomando o devido cuidado com o Overflow da

memória. Por fim configurar Baud rate divisor (390) e Fraction baud rate divisor (5) para a taxa de transmissão da comunicação serial ser de 9600 Bd.

O Freemaster deve estar configurado com a velocidade e a porta que será utilizada para ser estabelecida a comunicação como na Figura 9.

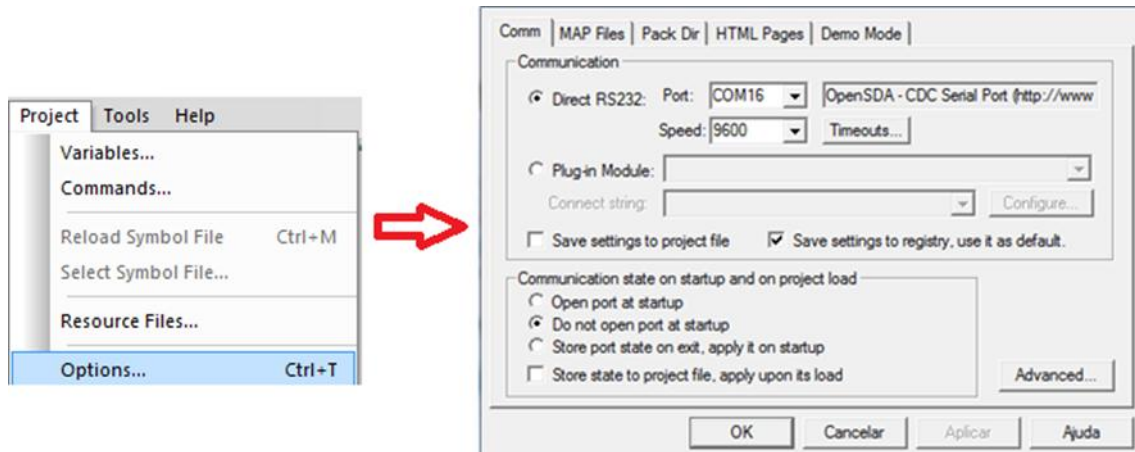


Figura 9 - Configuração da porta de interface USB

Também é necessário especificar qual é o projeto em análise conforme a Figura 10 para que sejam resgatadas as variáveis que deseja-se observar. Estas informações estão contidas em um arquivo gerado pelo CodeWarrior que é do tipo .elf.

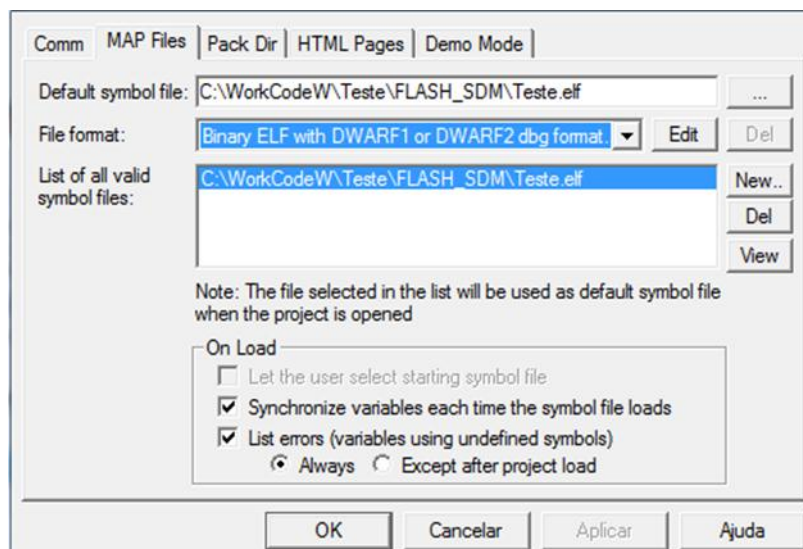


Figura 10 - Escolha do projeto em análise

Principais bibliotecas utilizadas:

ProcessorExpert.c

freemaster_fastrec.c
freemaster_rec.c
freemaster_56F8xxx.c
freemaster_appcmd.c
freemaster_bdm.c
freemaster_can.c
freemaster_protocol.c
freemaster_scope.c
freemaster_serial.c
freemaster_sfio.c
freemaster_tsa_table.c
freemaster_tsa.c
freemaster_protocol.h
freemaster_56F8xxx.h
freemaster_cfg.h
freemaster_cfggen.h
freemaster_defcfg.h
freemaster_private.h
freemaster_rec.h
freemaster_tsa.h
freemaster.h

3.1. PWM

Como o motor é trifásico devem-se habilitar três módulos de PWM, cada módulo com seu submódulo contendo um PWM e seu complementar. É possível realizar várias configurações de acordo com a necessidade. Em nosso caso configuramos o PWM com uma frequência de 16 kHz e tempo morto de 1 μ s.

Principais bibliotecas utilizadas:

PWM_A_config.c
PWM_A_config.h
appconfig.h

3.2. DRIVER (MC33937)

O driver utilizado exige que seja configurado o tipo de dado a ser enviado, no caso, um dado de 8 bits, para iniciar o seu funcionamento. A comunicação utilizada entre o controlador e o driver é a SPI.

Principais bibliotecas utilizadas:

SPI_TWR56F8400.c

MC33927.c
MC33927.h

3.3. TIMERA

Para o acionamento do motor é preciso monitorar a posição do rotor. Isso é feito por meio dos sensores de posição de efeito Hall presentes no motor. Para isso, o TIMERA é configurado de forma a gerar interrupções nas bordas (de subida/descida) dos pinos em que estão sendo feitas as leituras dos sensores. Foram configurados os TIMERA 0, 1 e 2 e a cada interrupção são selecionados quais braços da ponte inversora serão acionados.

Principais bibliotecas utilizadas:

TMRA0.c
TMRA0.h
appconfig.h

3.4. TIMERB

O TIMERB foi utilizado para a realização da leitura de velocidade a partir do encoder. Este foi configurado, de acordo com a referência , para capturar as bordas de subida fornecidas pelo encode, e armazenar a contagem primaria que esta vinculada com o clock do controlador.

Principais bibliotecas utilizadas:

TMRA0.c
TMRA0.h
appconfig.h

4. PRINCIPAIS FUNÇÕES DO FIRMWARE

```
static inline void commutationHS(unsigned int uiDirection)
{
    commutation_HALL++;
    read_C = GPIO_C_RDATA;
    HALL_C = ((read_C&0x0040)>>4)|((read_C&0x0010)>>3)|((read_C&0x0008)>>3);
    PWM_MASK = commutationTableMask[HALL_C];
}
```

```

    PWM_MASK_read      =    PWM_MASK;
    PWM_DTSRCSEL       =    commutationTableSWC[HALL_C];
    PWM_DTSRCSEL_read  =    PWM_DTSRCSEL;
    PWM_SM0_CTRL2     |=    PWM_SM0_CTRL2_FORCE;
}

void TIMER_A_0(void)
{
    commutationHS(0);
    TMRA0_SCTRL&= ~0x800;
}

void PWM_Interrupt(void)
{
    Counter_Capt=TMRB1_CAPT;
    if(Counter_Capt>0) Velo=div_Is(3600000<<1,Counter_Capt);
    else Velo=0;
    FMSTR_Recorder();
    setReg16(PWM_SM0_STS, 0x3FFFU);
}

void TimerB2_int(void){
    if(trigger_recorder>0 && trigger_counter<430){
        PRBS=vetor[trigger_counter];
        trigger_counter++;
        if(!(PRBS&0x1)){
            Set_Alt=1725;
        }
        else Set_Alt=2925;
    }

    PWM_SM0_VAL3      =    Set_Alt;
    PWM_SM1_VAL3      =    Set_Alt;
    PWM_SM2_VAL3      =    Set_Alt;
    PWM_MCTRL         |=    0x000F;
    PWM_OUTEN         =    0x0770;
    TMRB2_SCTRL&= ~0x8000;
}

void TimerB2_int(void){
    if(trigger_recorder>0 && trigger_counter<256){
        var1=(PRBS&0x1);
        var2=(PRBS&0x10)>>4;
        var3=(PRBS&0x20)>>5;
        var4=(PRBS&0x40)>>6;
        saida_P=((var1^var2)^var3)^var4;
        if(!(PRBS&0x1)){
            Set_Alt=1725;
        }
        else Set_Alt=2925;
        PRBS=(PRBS>>1)&0xff;
        if(saida_P){
            PRBS=PRBS|0x80;
        }
    }
}

```

```

    }
    trigger_counter++;
}
else{
    saida_P=0;
    Set_Alt=1725;
}
PWM_SM0_VAL3 = Set_Alt;
PWM_SM1_VAL3 = Set_Alt;
PWM_SM2_VAL3 = Set_Alt;
PWM_MCTRL |= 0x000F;
PWM_OUTEN = 0x0770;
}
TMRB2_SCTRL&= ~0x8000;
}

void TimerB2_int(void){
if(trigger_recorder>0){
    Erro = sub(Setpoint,Velo);
    P=L_mult(kp,Erro);
    add_erro=add(Erro,Erro_Ant);
    mult_L=L_mult_Is(ki,add_erro);
    I=L_add(mult_L,Iant);
    Iant=I;
    Erro_Ant=Erro;
    uk32=L_add(I,P);
    P2=div_Is(L_abs(uk32),24);
    DutyCycle =mult_r(3750,P2);
    if(DutyCycle<300 || (Erro&0x8000)) DutyCycle=300;
    PWM_SM0_VAL3 = DutyCycle;
    PWM_SM1_VAL3 = DutyCycle;
    PWM_SM2_VAL3 = DutyCycle;
    PWM_MCTRL |= 0x000F;
    PWM_OUTEN = 0x0770;
}
TMRB2_SCTRL&= ~0x8000;
}
}

```

5. REALIZANDO TESTES NO FREEMASTER

Para a obtenção dos dados para o processo de identificação do método estocástico é necessário à criação de uma visualização do Recorder no Freemaster e configurar alguns parâmetros como especificado na Figura 11.

A base de tempo deve ser configurada corretamente, pois a partir dela são identificados os instantes de amostragem, e baseando-se nestes dados, será realizado o processo de identificação. O “Board time base” é configurado na biblioteca

freemaster_cfggen.h alterando a variável FMSTR_REC_TIMEBASE de acordo com a taxa de amostragem da função Recorder do firmware.

A interrupção para o Freemaster é gerada no PWM de 16 kHz, assim a amostragem é de 62,5us aproximadamente.

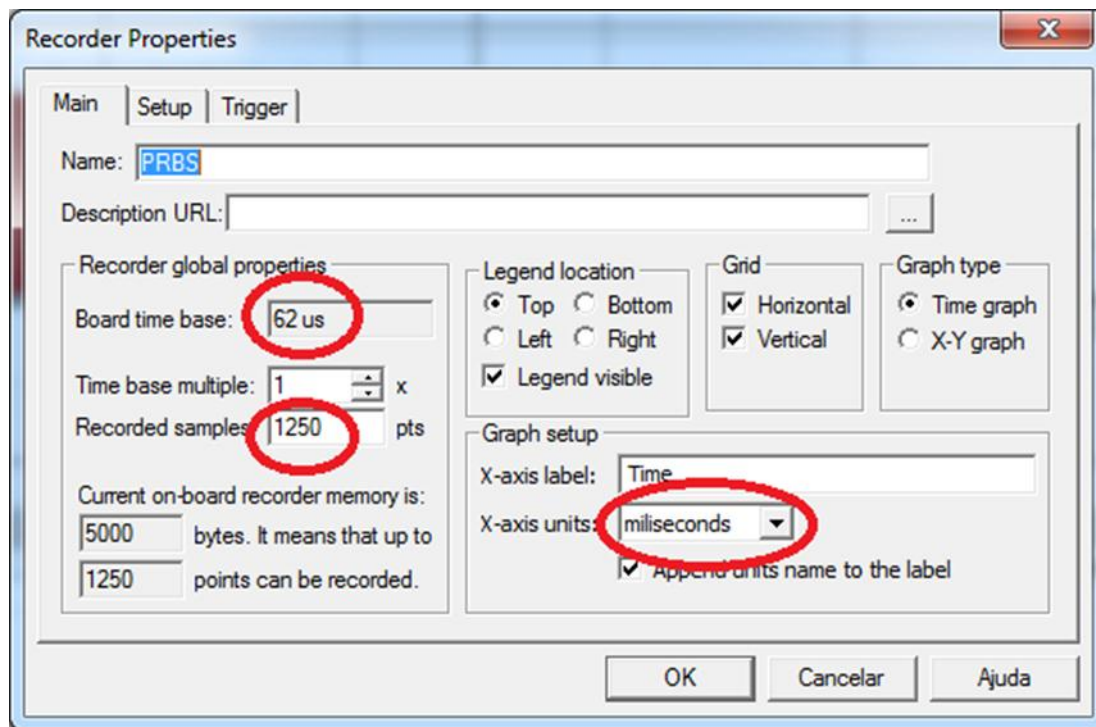


Figura 11 - Escolha das propriedades do Recorder

Em “Recorded samples” determina-se a quantidade de amostragens requerida. Sua limitação é proveniente do tamanho do buffer reservado no firmware.

É possível optar pela forma que o eixo x irá representar o período. Em nosso caso optamos por uma base de tempo em milissegundos devido às características do sistema.

Na opção “Setup” as variáveis que se desejam plotar são selecionadas. Nesta aplicação são o PRBS e a velocidade (entrada e saída) como a Figura 12.

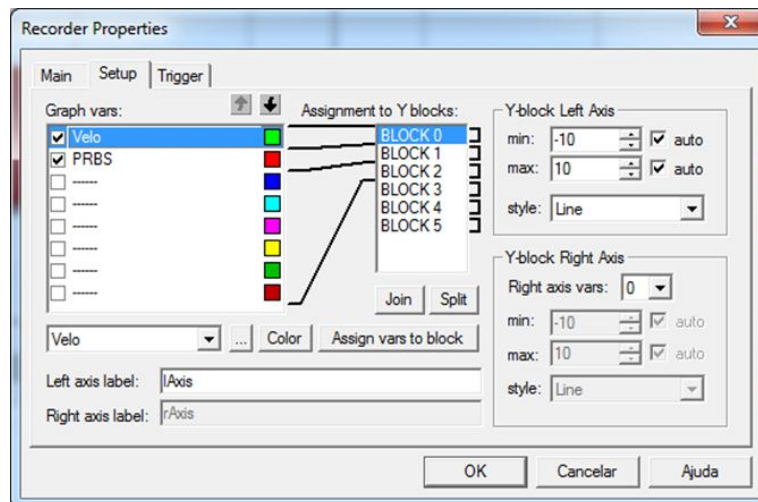


Figura 12 - Propriedades para exibição dos dados

No Recorder existe a necessidade de especificar um gatilho para a inicialização do armazenamento das amostras. No firmware foi criada uma variável (trigger_recorder) que no momento da aplicação do PRBS é incrementado, assim esta foi utilizada para o trigger.

Outra especificação importante é a quantidade de amostras a serem coletadas antes do trigger, que neste caso são descartadas (Figura 13).

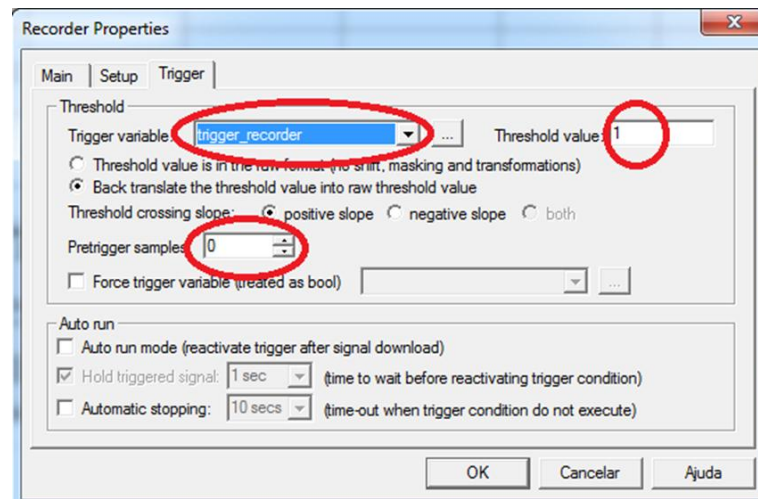


Figura 13 - Configurações do trigger

O gráfico resultante está apresentado na Figura 14.

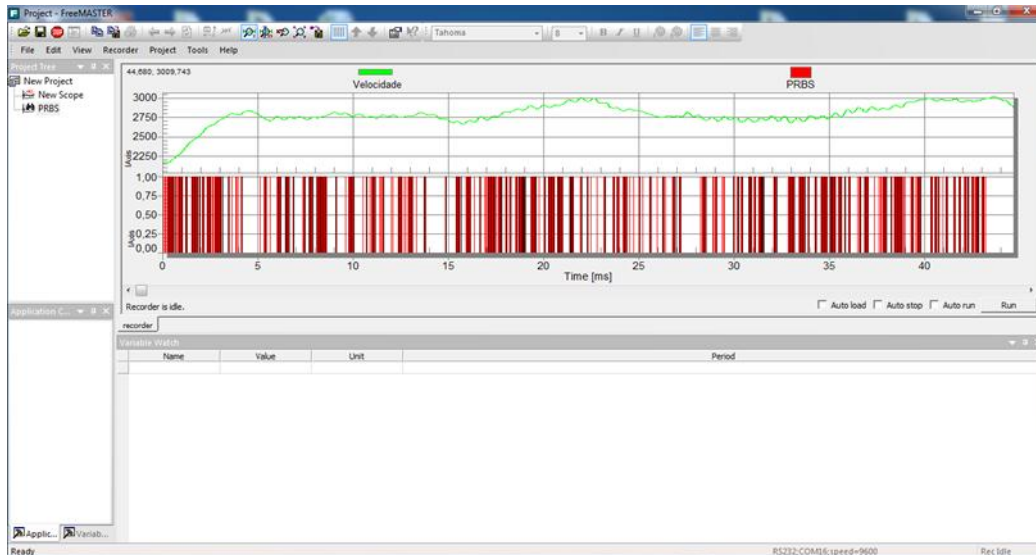


Figura 14 - Gráfico exibido utilizando o Recorder

Para a obtenção dos dados amostrados deve-se selecionar a opção “Toggle Data Capture” (Figura 15) e especificar o destino (Figura 16) do seu armazenamento no disco. Estes dados serão salvos em um arquivo texto.

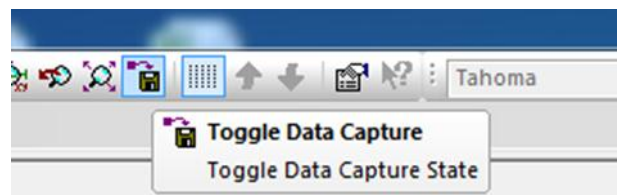


Figura 15 - Passo para obtenção dos dados amostrados

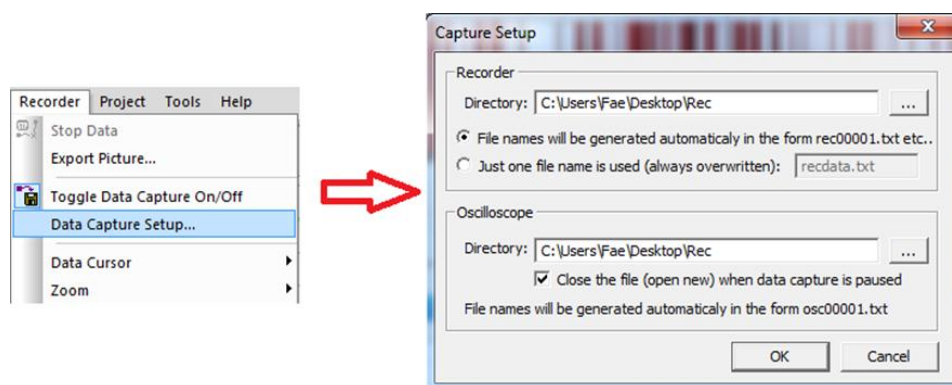


Figura 16 - Determinação do caminho para armazenamento dos dados

Na Figura 17 demonstra a resposta ao degrau (utilizando novamente o Recorder) do sistema utilizando um controlador PI com setpoint de 1000 rpm.

O processo de configuração do Recorder segue da mesma forma como utilizada na identificação pelo método estocástico.

Outra forma de visualizar a aquisição da velocidade é a através do Scope do Freemaster. Nesta ferramenta deve-se especificar o tempo em que é amostrado (Figura 18), pois os dados exportados pelo DSC são constantemente plotados. Também existe a necessidade de se especificar as variáveis amostradas em "Setup".

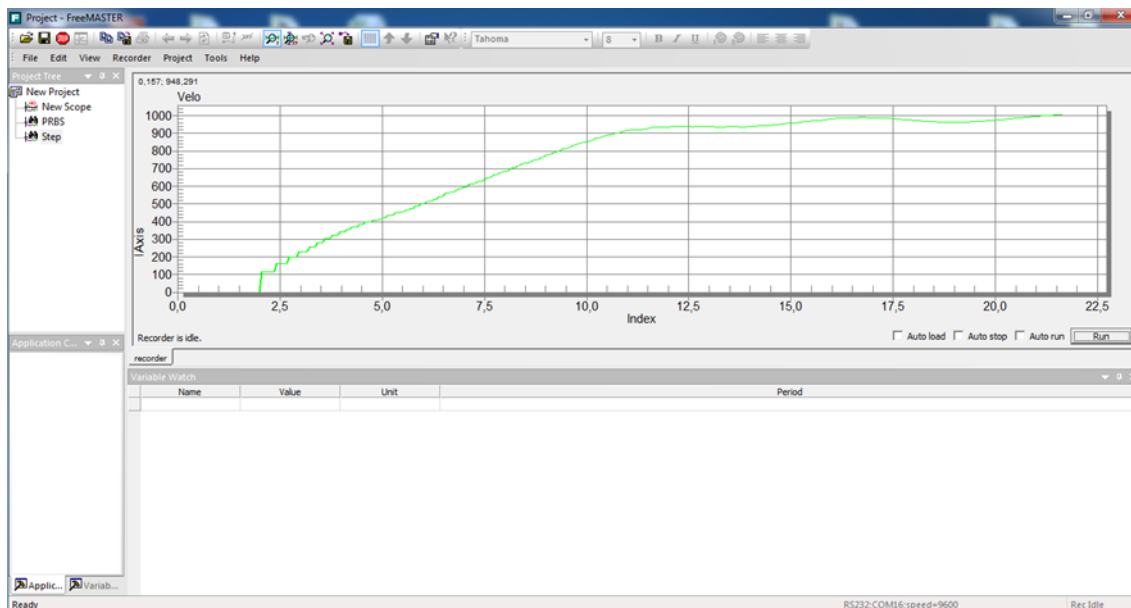


Figura 17 - Gráfico da resposta ao degrau com controle PI

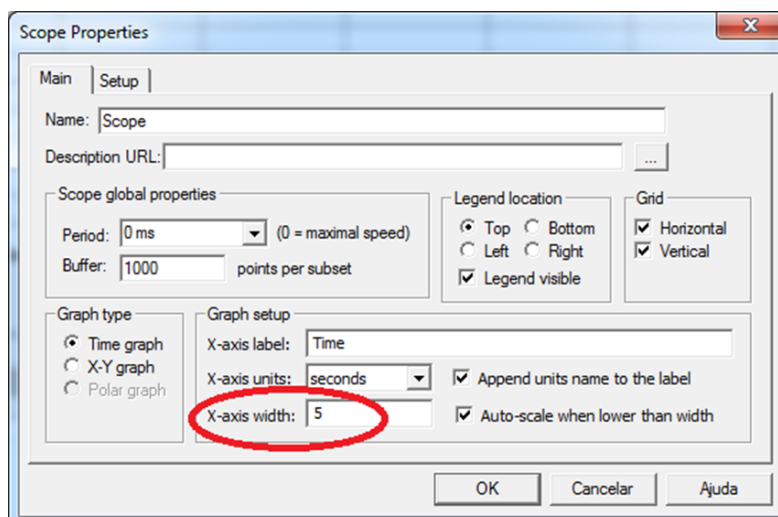


Figura 18 - Propriedades do Scope

Uma característica interessante do Freemaster é a possibilidade de se alterar os valores das variáveis durante a o funcionamento do sistema. Essa opção é de grande auxílio na obtenção dos ganhos do controlador quando não há o conhecimento do sistema, assim há possibilidade de se alterar os valores facilmente sem a necessidade de recompilar o firmware.

Este processo está exemplificado pela Figura 19 e Figura 20, mas neste caso o setpoint será alterado para especificar a velocidade requerida e assim visualizar a atuação do controlador (Figura 21).

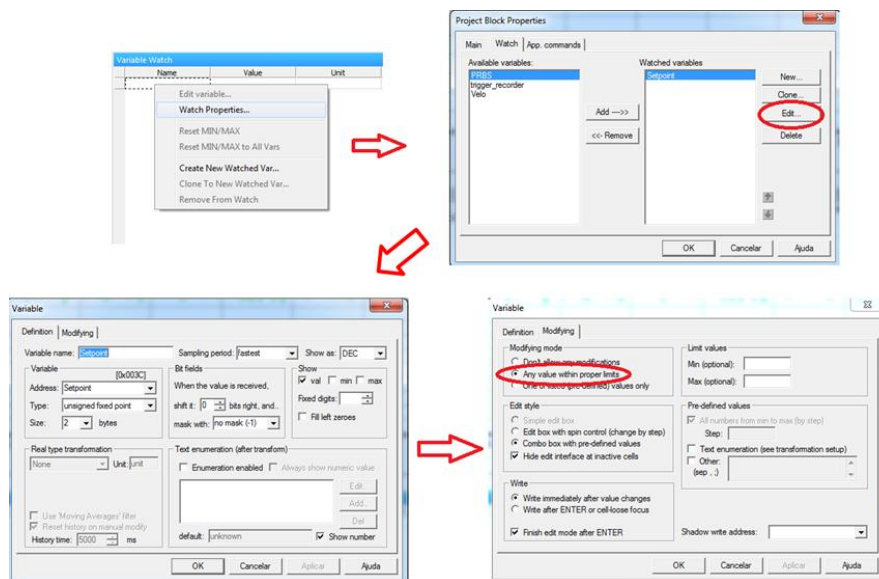


Figura 19 - Visualização da variável

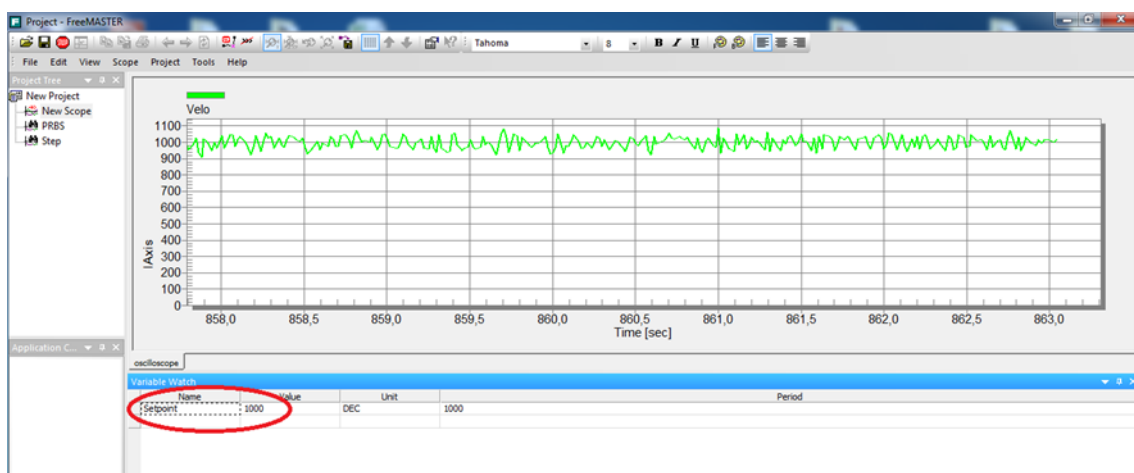


Figura 20 - Setpoint

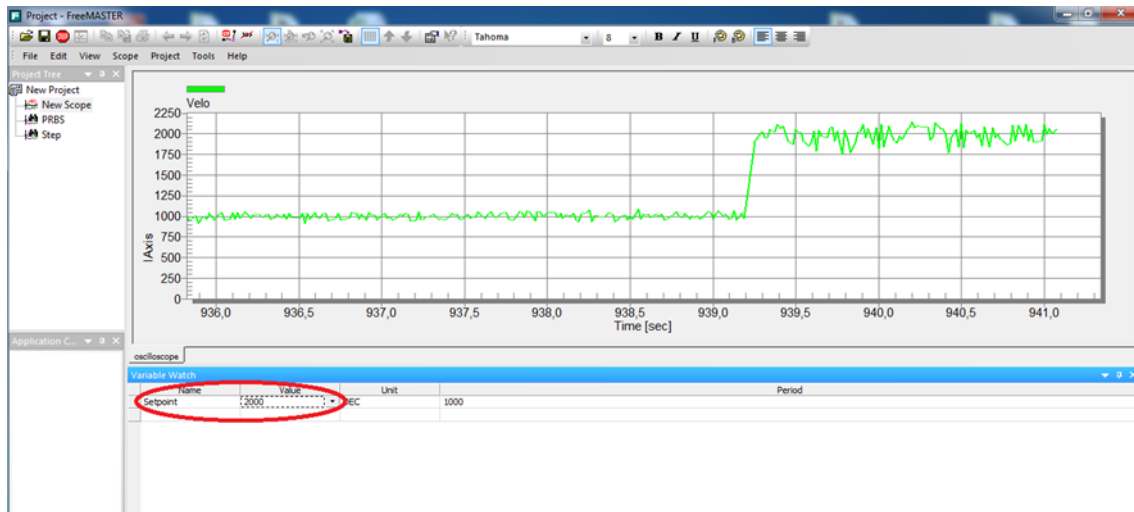


Figura 21 - Alteração do setpoint

6. IDENTIFICAÇÃO NO MATLAB

Para realizar a identificação no MATLAB, é necessário abrir a toolbox de identificação digitando “ident” na janela de comando conforme a Figura 22.

```

Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
>> ident
Opening System Identification Tool ..... done.
f3 >>

```

Figura 22 - Comando para abrir a ferramenta de identificação

O ambiente de identificação pode ser visto na Figura 23.

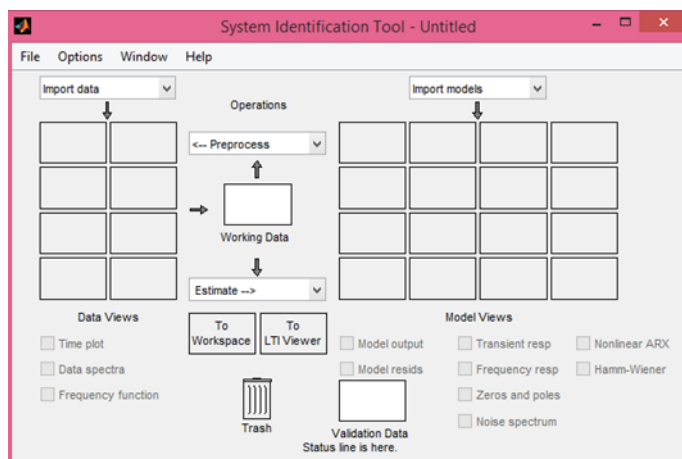


Figura 23 - Ambiente para identificação

Nele é possível importar novos dados ou então abrir uma sessão anteriormente criada. Para importar os dados, podem ser seleccionadas variáveis do workspace do MATLAB, seguindo os passos apresentados na Figura 24 e Figura 25. Em "Input" é colocado o nome da variável que contém as amostras dos dados de entrada, em "Output" escreve-se o nome da variável que armazena os dados de saída, o "Starting Time" deve ser zero e o "Sampling interval" deve corresponder ao tempo de amostragem utilizado.

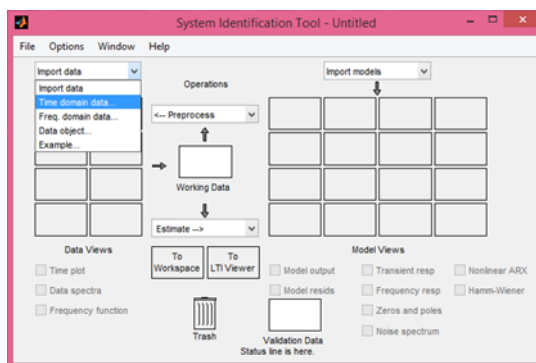


Figura 24 - Importar dados do workspace

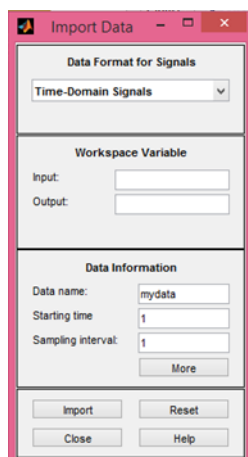


Figura 25 - Seleção de dados para importação

No caso de utilizar um novo conjunto de amostras pra a validação, deve-se repetir o procedimento.

O conjunto de dados importados irá aparecer na área de “Data Views” da ferramenta. Ao clicar sobre ela, ela fica selecionada e podem-se realizar visualizações do espectro e da resposta temporal, úteis para verificar a qualidade do sinal a ser utilizado na identificação.

No caso de se selecionar uma faixa do sinal para identificação e outra para validação, arrasta-se o “Data View” de interesse para o espaço denominado “Working Data” como apontado na Figura 26, e então se seleciona uma faixa do sinal até certo valor para criar um novo “Data View” através do passo apresentado na Figura 27, e mais uma vez para dados a partir desse valor para criar outro. As duas seleções realizadas também podem ser visualizadas na Figura 26.

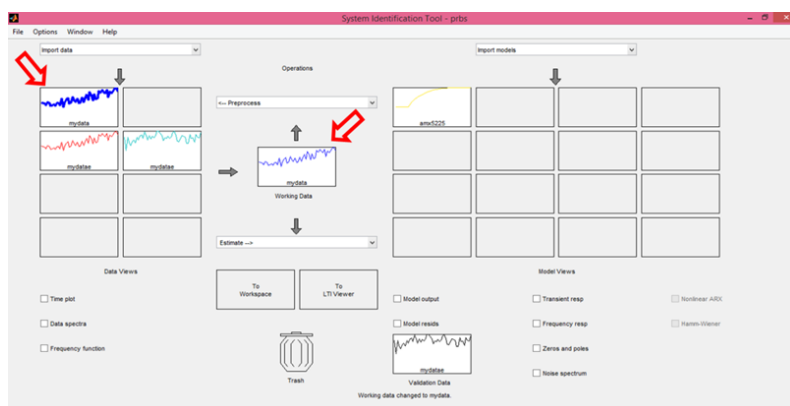


Figura 26 - Seleção dos dados a serem trabalhados

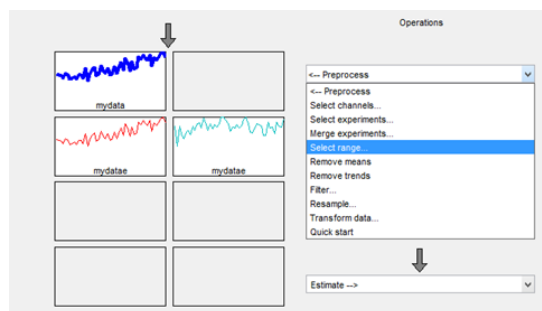


Figura 27 - Criação de novo conjunto a partir de faixa de dados

Com os dados preparados, o conjunto a ser utilizado na identificação deve ser arrastado sobre o campo "Working Data" e o destinado à validação sobre o campo "Validation Data".

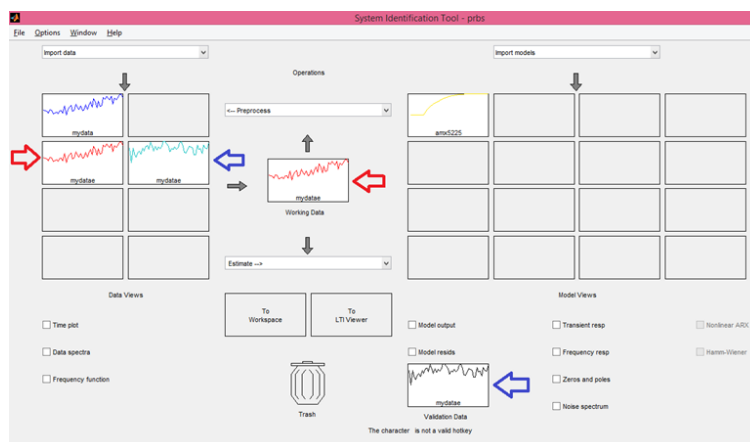


Figura 28 - Escolha dos dados para identificação e validação

Inicia-se então o processo de identificação do modelo matemático do sistema. No menu "Estimate ->", seleciona-se a opção de modelos polinomiais conforme a Figura 29.

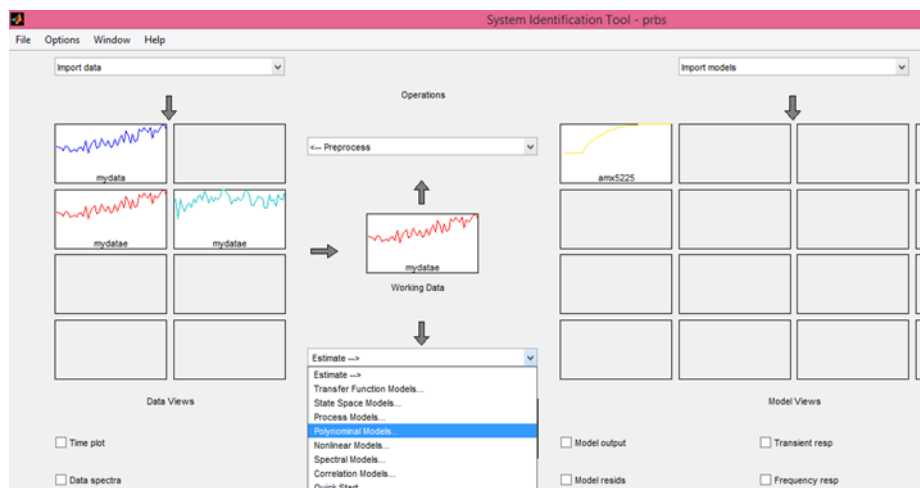


Figura 29 - Abertura da janela pra estimação do modelo polinomial

Na janela da Figura 30 seleciona-se o tipo de modelo a ser empregado, bem como a ordem dos polinômios.

Para o modelo ARX há uma ferramenta disponível, exibida na Figura 31, que estima a ordem dos polinômios. Ela fornece uma ideia de valor a ser aplicado, porém, é possível obter respostas mais satisfatórias através de tentativa e erro a partir da análise da validação do modelo. Para abri-la, basta clicar sobre o botão “Order Selection” antes de “Estimate”.

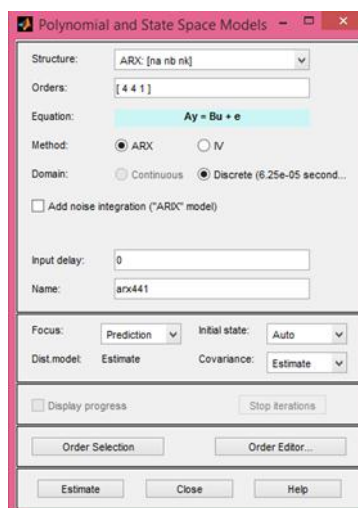


Figura 30 - Janela para seleção do modelo

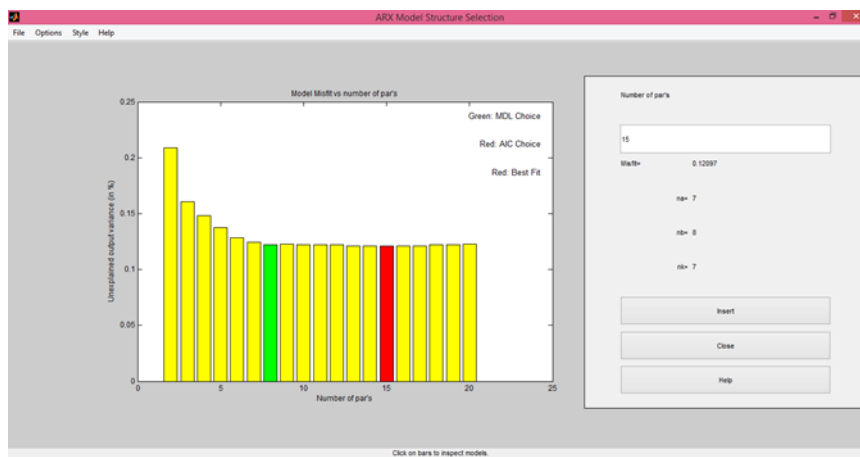


Figura 31 - Estimação do grau dos polinômios para o modelo ARX

Para cada modelo testado, observam-se os resíduos (Figura 32) e os polos e zeros (Figura 33), principalmente.

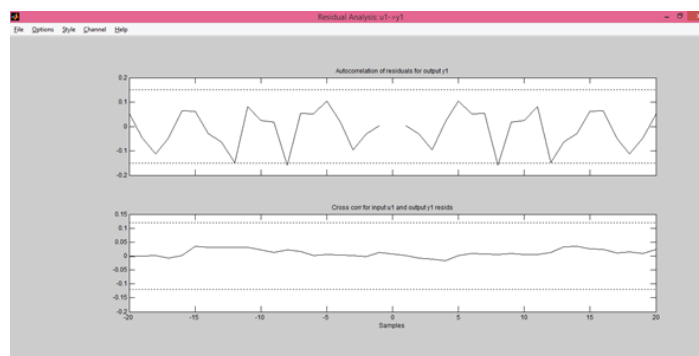


Figura 32 - Análise dos resíduos

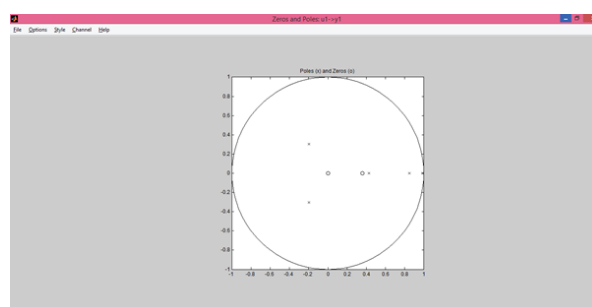


Figura 33 - Diagrama de polos e zeros

A observação da dinâmica do sistema capturada pelo modelo é verificada através da análise da saída do modelo em comparação à saída real selecionando a opção "Model Output".

As análises supracitadas são visualizadas selecionando um dos modelos em “Model Views”, e então marcando as que se deseja exibir.

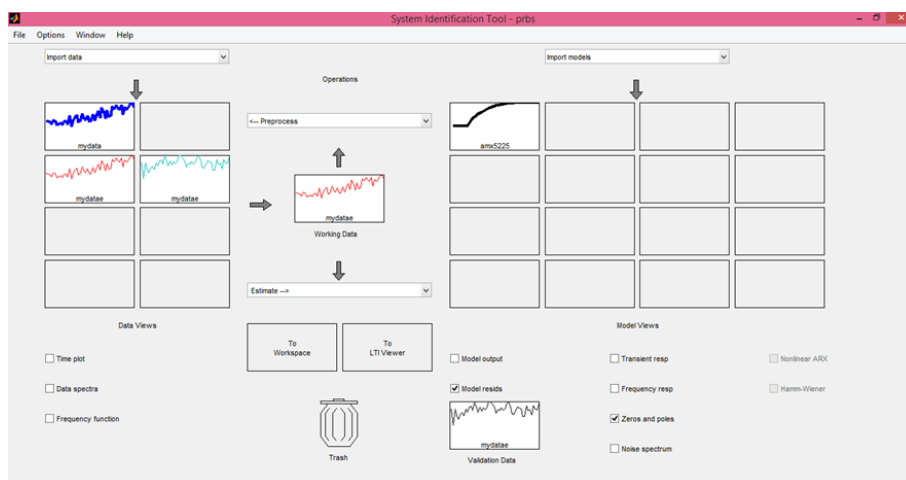


Figura 34 - Seleção de análises desejadas