



UNIVERSIDADE LUTERANA DO BRASIL
PRÓ-REITORIA DE GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA



Fábio Fischer de Oliveira

Controle de Velocidade de Motor Brushless DC

Canoas, Dezembro de 2008.



Fábio Fischer de Oliveira

Controle de Velocidade de Motor Brushless DC

Trabalho de Conclusão de Curso apresentado ao Departamento de Engenharia Elétrica da ULBRA como um dos requisitos obrigatórios para a obtenção do grau de Engenheiro Eletricista

Departamento:

Engenharia Elétrica

Área de Concentração

Eletrônica de Potência

Professor Orientador:

Dr. Eng. Eletr. Valner João Brusamarello – CREA-RS: 78.158-D

Canoas

2008



FOLHA DE APROVAÇÃO

Nome do Autor: Fábio Fischer de Oliveira

Matrícula: 992101311-4

Título: Controle de Velocidade de Motor Brushless DC

Trabalho de Conclusão de Curso apresentado ao Departamento de Engenharia Elétrica da ULBRA como um dos requisitos obrigatórios para a obtenção do grau de Engenheiro Eletricista

Professor Orientador:

Dr. Eng. Eletr. Valner J. Brusamarello

CREA-RS: 78.158-D

Banca Avaliadora:

Dr. Eng. Eletr. Marília A. da Silveira

CREA-RS: 50.909-D

Conceito Atribuído (A-B-C-D):

Dr. Eng. Eletr. Adriane Parraga

CREA-RS: 107.196-D

Conceito Atribuído (A-B-C-D):

Assinaturas:

Autor

Fábio Fischer de Oliveira

Orientador

Valner J. Brusamarello

Avaliador

Marília A. da Silveira

Avaliador

Adriane Parraga

Relatório Aprovado em:



DEDICATÓRIA

Dedico a minha família, pelo apoio e compreensão nos momentos de ausência.



AGRADECIMENTOS

Agradeço a Deus pela vida.

A minha família pelo incentivo e apoio incondicional que sempre colocaram à minha disposição, e pela compreensão nos momentos em que estive ausente.

Ao Professor Valner pelo apoio, dedicação e esforço pessoal proporcionado.

Ao colega Fábio Fernandes pelo empréstimo do encoder em um momento decisivo. Aos demais colegas por tudo que aprendi com cada um de vocês nesse período de convivência.

A todos os professores que tive o privilégio de ser aluno, não somente pelo ensino acadêmico, mas por proporcionarem uma outra visão de mundo durante a construção dos pilares da minha vida.



EPÍGRAFE

“Se você acha a educação cara, experimente a ignorância”

W.Bok

“Não se pode ensinar coisa alguma a alguém, pode-se apenas auxiliá-lo a descobrir por si mesmo”

Galileu Galilei



RESUMO

Autor: Oliveira, Fábio Fischer de. **Controle de Velocidade de Motor Brushless DC**. 87 p. Trabalho de Conclusão de Curso em Engenharia Elétrica - Departamento de Engenharia Elétrica. Universidade Luterana do Brasil. Canoas, RS. 2008.

Neste trabalho é apresentado o controle de velocidade com sensor de um motor brushless. Para fazer esse controle foi utilizado um encoder de quadratura acoplado ao eixo do motor e um circuito de acionamento composto de três meias-pontes H. O ajuste de velocidade é feito variando-se um potenciômetro ligado na entrada do A/D do microcontrolador.

Os resultados mostraram que o encoder de quadratura é uma boa opção para determinação de posição e velocidade angular do motor brushless, e que é possível o controle da velocidade do motor através da variação do *duty cycle* dentro de um período fixo do PWM.

Palavras chave: Controle de Velocidade. Circuito de Acionamento. Motores de alta eficiência. Motor Brushless. Microcontrolador DSPIC. Sensor de posição.



ABSTRACT

Autor: Oliveira, Fábio Fischer de. **Speed Control of Brushless DC Motor.**
XX p. Work of Conclusion of Course in Electrical Engineering - Electrical
Engineering Department. Lutheran University of Brazil. Canoas, RS. 2008.

In this work is presented a sensed speed control in a brushless DC motor. To make this control has quadrature encoder bound to the shaft engine and a circuit drive composed of three half - bridges (h). The adjustment speed is done varying a potentiometer connected in the A/ D of microcontroller.

The results showed that the quadrature encoder is a good choice for determination of position and angular speed of brushless motor, and the control of engine speed through the variation of the duty cycle within a fixed period of PWM conversion.

Keywords: Speed Control. Drive Circuit. High Efficiency Motors. Brushless DC Motor. DSPIC Microcontroller. Position Sensor.



LISTA DE ILUSTRAÇÕES

Figura 1 - Motor Síncrono AC.	6
Figura 2 - Motor de Passo.	7
Figura 3 - Motor de passo em visão de corte.	7
Figura 4 - Visão de corte de um motor BLDC.	9
Figura 5 - Diagrama simplificado do circuito acionador.	12
Figura 6 - Corte transversal de um MOSFET tipo N (NMOS).....	13
Figura 7 - Símbolo esquemático de um TJB, npn.....	16
Figura 8 - Símbolo esquemático de um TJB, pnp.....	16
Figura 9 - Sinais de Modulação por Largura de Pulso (PWM).....	18
Figura 10 - Diagrama de Pinos do DSPIC30F3010.	19
Figura 11 - Detecção ótica de Velocidade/Direção/Posição.....	21
Figura 12 - Compensador Proporcional Integral	22
Figura 13 - Diagrama de blocos simplificado do sistema.....	24
Figura 14 - Motor utilizado	25
Figura 15- Rotor circular com ímãs na periferia.....	26
Figura 16 - Sinais de interface do encoder de quadratura.....	26
Figura 17 - Circuito de acionamento motor trifásico	27
Figura 18 - Ligação dos enrolamentos	28
Figura 19 - Esquema elétrico atualizado	29
Figura 20 - PWM alinhado na borda	30
Figura 21 - Seqüência de energização dos enrolamentos.....	31
Figura 22 - Fluxograma do encoder com o envio pela serial.....	32
Figura 23 - Fluxograma do PWM de acordo com a leitura do encoder.....	34
Figura 24 - Foto do projeto desenvolvido.....	35
Figura 25 - Um sistema de controle de velocidade de malha fechada.....	36
Figura 26 - Forma de onda pinos PWM.....	37
Figura 27 - Forma de onda da tensão nas bobinas A e B do motor.....	38
Figura 28 - Circuito utilizado para o A/D.....	38
Figura 29 - Gráfico da velocidade em função da tensão no A/D.....	39
Figura 30 - Circuito para obtenção da tensão em relação à frequência.....	40
Figura 31 - Placa de aquisição utilizada.....	41
Figura 32 - Gráfico tensão x tempo.....	42
Figura 33 - Obtenção das constantes L e T.....	43
Figura 34 - Forma de onda da tensão no resistor de shunt.....	44



LISTA DE TABELAS

Tabela 1 – Atributos do MCU, DSP e DSC 20



LISTA DE ABREVIATURAS E SIGLAS

BLDC: Brushless Direct Current

DSPIC: Digital Signal Programmable Interface Controller

DC: Direct Current

AC: Alternate Current

MCU: Microcontroller

DSP: Digital Signal Processor

DSC: Digital Signal Controller

FPGA: Field Programmable Gate Array

PWM: Pulse Width Modulation

MOSFET: Metal Oxide Semiconductor Field Effect Transistor

FCEM: Força contra-eletromotriz

PI: Proporcional-Integral



LISTA DE SÍMBOLOS

V – Tensão [V]

μ_n – Mobilidade dos elétrons

W – Largura do canal

Co – Capacitância de placas paralelas do dielétrico de porta por unidade de área

x – Profundidade do canal

L – comprimento entre dreno e substrato



SUMÁRIO

1. INTRODUÇÃO.....	1
1.1. Objetivos do Trabalho	3
1.2. Justificativa Técnica e Econômica para a Solução Escolhida.....	3
2. REFERENCIAL TEÓRICO	5
2.1. Motores Elétricos	5
2.2. Ponte H	11
2.3. PWM.....	17
2.4. Controladores de Sinal Digital	18
2.5. Encoder Incremental	20
2.6. Controle Proporcional Integral	22
3. MATERIAIS E MÉTODOS	24
3.1. Descrição Geral do Projeto	24
3.2. Descrição dos Sistemas Mecânicos	25
3.3. Descrição dos Sistemas Eletroeletrônicos	27
3.4. Descrição dos Sistemas Informáticos e Computacionais	30
4. APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS.....	36
4.1. Controle da Velocidade pelo Potenciômetro	37
4.2. Obtenção da Curva de Resposta do Motor	40
5. CONSIDERAÇÕES FINAIS.....	46
5.1. Problemas Encontrados.....	46
5.2. Conclusões.....	47
5.3. Sugestões para Trabalhos Futuros.....	47
6. REFERÊNCIAS.....	49
OBRAS CONSULTADAS	50
APÊNDICE A – CÓDIGO FONTE	51
ANEXO A – DATASHEET DSPIC30F3010	59
ANEXO B – DATASHEET DO ENCODER.....	62
ANEXO C – DATASHEET DO MOSFET IRF640.....	66
ANEXO D – DATASHEET DO MOSFET IRF9640.....	70



1. INTRODUÇÃO

Já faz algum tempo que se ouve falar em preservação ambiental, e principalmente nos últimos anos tem crescido o interesse mundial pelo futuro do planeta e conseqüente futuro de cada habitante dele.

Como é sabido, os recursos naturais são finitos e neste contexto, há uma busca cada vez maior do uso racional destes recursos. Diversas são as fontes de energia que possibilitam sobrevivência, facilidades e conforto para os seres humanos. A água, o petróleo e seus derivados, enfim, diversos recursos naturais se fazem presentes, movimentam a economia e tornam a nossa vida mais fácil.

A busca pelo uso racional destes recursos finitos ou até mesmos escassos, tem motivado no mundo todo pesquisas de uso mais eficiente destes recursos, como o automóvel que consome menos combustível, fontes alternativas de energia, como o biocombustível, etc.

Na energia elétrica não é diferente e pensando nisso, busca-se neste trabalho a utilização de um motor elétrico de alta eficiência, um motor que disponibilize mais energia mecânica no eixo do motor com a mesma energia elétrica entregue ao sistema.

O motor escolhido para nortear o projeto foi o Brushless DC, pela alta eficiência e baixa necessidade de manutenção. O desafio é implementar o hardware e software necessário para controle de velocidade deste motor. Além da necessidade de um hardware de maior complexidade, este motor também tem um custo mais elevado frente aos motores tradicionais.

Este trabalho destina-se à Universidade Luterana do Brasil e poderá ser aproveitado no veículo elétrico de competição do curso de Engenharia Mecânica.



A grande maioria dos projetos de controle de velocidade de motores utiliza motores comuns, como motores de corrente contínua, pelo menor custo e facilidade de implementação de hardware.

O objetivo deste projeto é implementar o controle de velocidade em um motor de alta eficiência.

O curso de Engenharia Mecânica da Ulbra, no seu veículo de competição, utiliza um motor comum, de baixa eficiência. Busca-se melhorar a eficiência do motor do veículo de competição, ou seja, entregar mais energia mecânica no eixo do motor com a mesma energia elétrica entregue ao sistema.

O uso de um motor mais eficiente significa mais energia mecânica sendo entregue à carga/eixo do motor ou a necessidade de uma energia elétrica menor sendo entregue ao equipamento. Com o rendimento mais alto do equipamento aumenta-se a produtividade. Com a redução do consumo o custo diminui, e economicamente aumenta-se a rentabilidade do negócio.

O uso mais racional da energia elétrica diminui o consumo, ajuda a equilibrar a tênue relação oferta/demanda de energia. Menos energia demandada diminui o impacto ambiental causado pela construção de mais uma hidrelétrica, numa ótica social de larga escala.

Este projeto destina-se à Ulbra, como mencionado anteriormente, mas os motores têm larga utilização na indústria.

A preocupação com o meio ambiente gera muitas possibilidades de negócios, e a utilização de motores elétricos além de serem ecologicamente mais corretos do que a utilização de motores movidos a combustíveis fósseis, agrega valor a qualquer empresa pela imagem que passa ao grande público de empresa responsável com o meio ambiente.

A grande utilização de motores elétricos na indústria, a necessidade cada vez maior de automação industrial para aumentar a produtividade das empresas, constituem um universo de possibilidades do uso do motor a ser controlado neste projeto.



1.1. Objetivos do Trabalho

O trabalho tem por objetivo estudar um motor brushless DC e projetar, desenvolver, construir e testar um controle de velocidade para esse motor.

São objetivos específicos do trabalho:

- Estudo de um microcontrolador adequado;
- Construção de um hardware para o controle de um motor BLDC;
- Implementação de um controle PWM utilizando um PI.

A meta primária é tornar possível a utilização de um motor com eficiência mais elevada que um motor DC escovado. Os resultados previstos são a utilização do controle implementado no carro elétrico da ULBRA.

Poderia obter um controle de motor através do uso de diversos motores comuns e de largo uso industrial, como motores DC, AC, entre tantos outros que já possuem grande domínio público da forma de implementação de controle.

1.2. Justificativa Técnica e Econômica para a Solução Escolhida

Primordialmente, foi um projeto proposto pela Universidade e direcionado a esta, mas a solução de utilizar um motor BLDC para implementar um controle de velocidade mostra-se bastante acertada pela elevada eficiência e por não ter escovas, o que diminui a necessidade de manutenção e o ruído elétrico, e as perdas de potência.

Um motor BLDC é mais eficiente, mais rápido, mais confiável, tem ruído elétrico menor e baixa necessidade de manutenção. Em contrapartida, tem um custo maior e necessidade de um hardware mais complexo.

Um motor comum é mais barato, o hardware é mais simples, mas tem eficiência menor, e a presença de escovas resultam em necessidade de manutenção e maior ruído elétrico.



O projeto deve satisfazer as condições iniciais, ou seja, possibilitar o controle de velocidade de um motor BLDC. Também se deve avaliar a apresentação do projeto.



2. REFERENCIAL TEÓRICO

2.1. *Motores Elétricos*

A premissa básica deste projeto é desenvolver um sistema de controle de velocidade em um motor de alta eficiência. O controle de motores ditos “comuns” já é largamente difundido e esses motores possuem como vantagens o baixo custo do motor e a facilidade de implementação de hardware.

2.1.1 Motores AC

Os motores AC de ímã permanente são referidos freqüentemente como motores sem escova ou motores CC sem escovas. Essa terminologia surge devido à similaridade com motores CC, quando combinada com um sistema de acionamento de freqüência e tensão variáveis, entre suas características de velocidade versus conjugado, e também devido ao fato de que é possível ver esses motores como sendo motores CC às avessas, em que o circuito de campo está no rotor e a armadura é comutada eletronicamente pelo sensor de posição e por interruptores conectados aos seus enrolamentos [Fitzgerald, 2006].

O motor síncrono AC possui o enrolamento da armadura no estator, e este por indução faz girar o rotor que é um ímã permanente. O rotor gira com freqüência igual ou múltipla daquela da AC aplicada, como pode ser visto na figura 1.

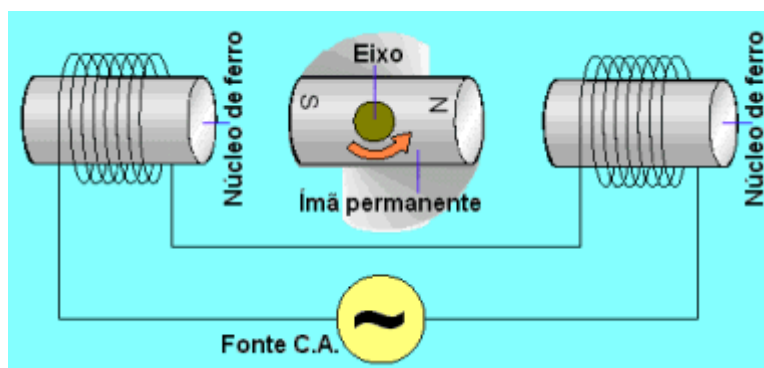


Figura 1 – Motor Síncrono AC
Fonte: Feira de Ciências (2008)

O motor síncrono AC é essencialmente um motor DC virado do avesso. As escovas e o comutador são eliminados e os enrolamentos são conectados ao controle eletrônico. O controle eletrônico substitui a função do comutador e energiza o enrolamento adequado.

2.1.2 Motores DC

2.1.2.1 Motores de passo

O motor de passo é um tipo de motor elétrico utilizado em movimentos que exigem rotação em um ângulo exato. São encontrados com ímã permanente (divididos em: unipolar, bipolar e multifase) e relutância variável. A figura 2 mostra um motor de passo.



Figura 2 – Motor de Passo

Fonte: Permuta Livre (2008)

Os motores de relutância variável, usualmente, têm três ou às vezes quatro bobinas, com um ponto comum. Já os motores de ímã permanente, usualmente, têm duas bobinas independentes, com ou sem o comum central. São controlados por uma série de campos eletromagnéticos ativados e desativados eletronicamente. A figura 3 ilustra um motor de passo em corte.

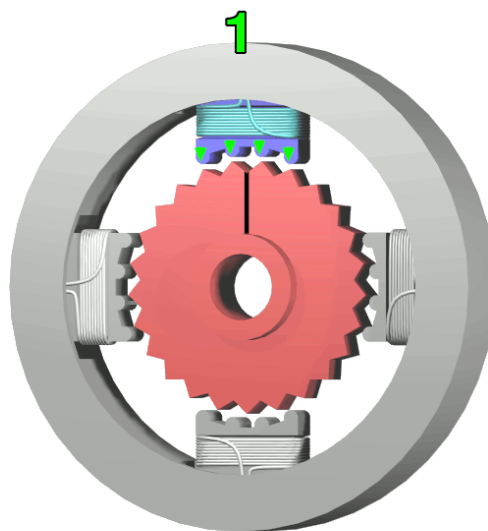


Figura 3 – Motor de passo em visão de corte

Fonte: Wikipédia (2008)



Motores de passo não usam escovas ou comutadores, possuem um número fixo de pólos magnéticos que determinam o número de passos por revolução. Ao se alimentar uma das bobinas, é gerado um campo magnético no estator, que induz o rotor a alinhar-se com o campo, gerando um movimento chamado “passo”. Normalmente, estes motores possuem 200 passos por volta. Sendo 360° de rotação, cada passo avança em $1,8^\circ$. Com alguns controladores, pode-se posicionar o estator em micropassos para obter uma resolução melhor no movimento.

Quanto ao torque, uma característica exclusiva deste tipo de motor é a sua habilidade de poder manter o eixo em uma posição, segurando o torque sem estar em movimento. Para atingir todo o seu torque, as bobinas de um motor de passo devem receber, durante cada passo, toda a corrente marcada proveniente do controlador do motor. O torque do motor é uma grandeza vetorial da física. É definido como a fração de força aplicada sobre um objeto que é efetivamente utilizado para fazer ele girar em torno de um eixo, conhecido como ponto pivô. A distância do ponto do pivô ao ponto onde atua uma força F é chamada braço do momento e é denotada por r , dado pela rotação vetorial modular $T = R F \text{ sen } (\varphi)$. [Silva, 2006].

2.1.2.2 Motores Brushless DC

A construção do motor brushless DC é similar a de um motor AC, conhecido como motor síncrono de ímã permanente. No motor brushless, os enrolamentos de armadura são parte do estator, e o rotor é composto de um ou mais ímãs.

Os motores brushless DC diferem dos motores AC síncrono, na detecção da posição do rotor (ou pólos magnéticos) para produzir sinais para controlar as chaves eletrônicas.

Motores BLDC são basicamente motores DC ao avesso. Em um motor DC o circuito de campo está no estator. O rotor tem os enrolamentos da armadura, os quais são excitados com uma corrente. A corrente no rotor é invertida para criar uma rotação ou campo elétrico em movimento por meio de divisão do comutador e escovas. Por outro lado, em um motor BLDC os enrolamentos estão no estator e os ímãs permanentes estão localizados no rotor. Portanto o termo motor DC ao avesso.

Para fazer o rotor girar, deve haver um campo elétrico em rotação. Tipicamente um motor BLDC trifásico tem um estator trifásico com duas bobinas excitadas por vez para criar um campo elétrico girante. Esse método é fácil de implementar, mas para evitar que o rotor de ímã permanente fique travado com o estator, a excitação do estator deve ser chaveada de uma maneira específica enquanto se sabe a exata posição do rotor.

A informação de posição pode ser obtida por um encoder acoplado ao eixo do motor, ou mais freqüentemente, por sensores de efeito Hall que detectam a posição do rotor. Para um motor trifásico típico, do tipo BLDC há seis regiões distintas ou setores nas quais dois enrolamentos específicos são excitados. [AN957]

O motor BLDC é geralmente operado com um ou mais sensores de posição uma vez que a excitação elétrica deve ser sincronizada com a posição do rotor. Por razões de custo, confiabilidade, embalagem mecânica e especialmente se o rotor trabalha imerso em fluido, é desejável usar o motor sem sensores de posição – chamado assim de operação sem sensor.

Na figura 4 tem-se uma ilustração representativa do motor BLDC em corte.

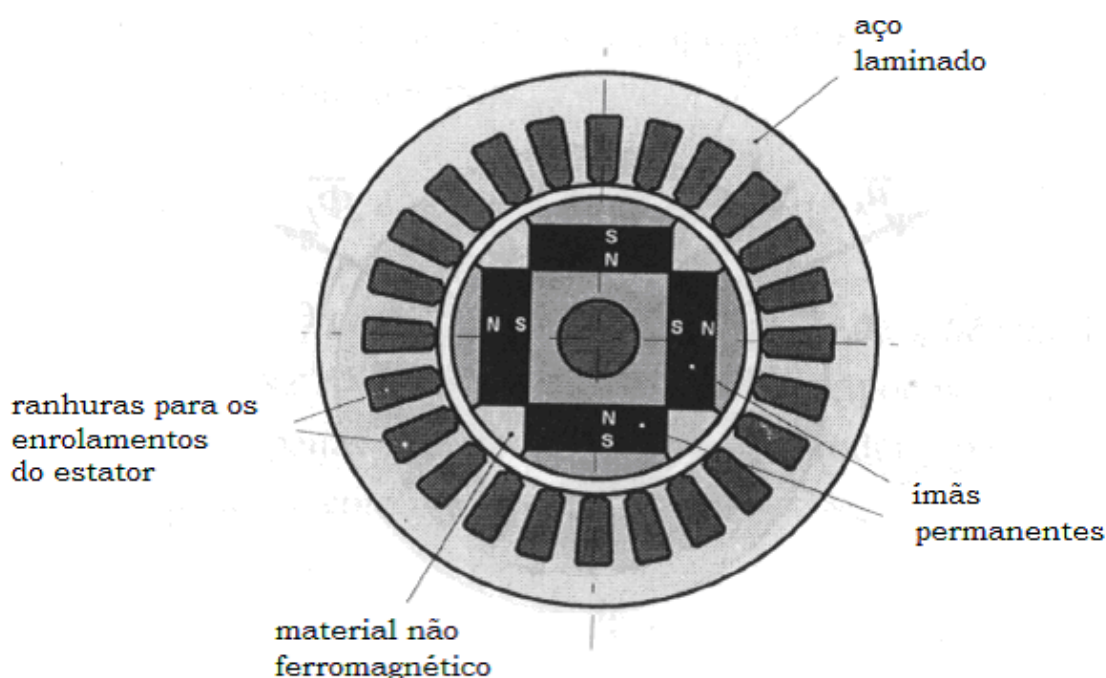


Figura 4 – Visão de corte de um motor BLDC

Fonte: UFMG (2008)



A maneira mais fácil de saber o momento correto de comutar as correntes nos enrolamentos é através da utilização de um sensor de posição. Muitos fabricantes de motores BLDC fornecem motores com três sensores de efeito hall para determinação de posição do rotor.

É possível determinar quando comutar a tensão de acionamento do motor pela detecção da força contra-eletromotriz em um terminal do motor não acionado durante uma das fases de acionamento. A vantagem óbvia do custo do controle sem sensor é a eliminação dos sensores de posição Hall. Há diversas desvantagens no controle sem sensor:

- O motor deve estar girando a uma velocidade mínima para gerar suficiente FCEM para ser sentida;
- mudanças repentinas na carga do motor podem fazer o loop de acionamento FCEM perder o sincronismo;
- a FCEM só pode ser medida quando a velocidade do motor está dentro de uma faixa limitada de taxa de comutação ideal para a tensão aplicada;
- comutações a taxas mais rápidas que a taxa ideal resultará em uma resposta descontínua do motor.

Se o baixo custo é uma preocupação primária e a operação do motor em baixas velocidades não é requisito e não é esperada a mudança brusca na carga do motor então o controle sem sensor pode ser a melhor escolha para a sua aplicação[AN857].

Alternativamente à utilização de sensores de efeito Hall para determinar a posição do rotor, pode-se utilizar um encoder de quadratura, a informação do encoder pode ser obtida de registradores de hardware e lógica de software dentro de um microcontrolador PIC. Por exemplo, o DSPIC30F3010 tem uma lógica de Interface de Encoder de Quadratura integrada ao processador[AN894].

Os motores de corrente contínua sem escovas ou BLDC (Brushless DC) oferecem diversas vantagens sobre os motores de corrente contínua com escovas, dentre as quais se podem destacar a confiabilidade mais elevada, o ruído reduzido, a vida útil mais longa (devido a ausência de desgaste da escova), a eliminação da



ionização do comutador, e a redução total de interferência eletromagnética (EMI). Também não há centelhas ou faíscas. A não utilização de anéis e escovas eliminam as perdas do rotor. Não há perdas por efeito joule, uma vez que os ímãs não são condutores. Apresenta grande adaptação às características de projeto desejadas pelo tipo e arranjo dos ímãs.

Para a mesma potência de entrada, os motores de BLDC converterão mais energia elétrica em energia mecânica do que um motor de corrente contínua escovado. A eficiência é maior na região de "baixa-carga" e "à vazio" na curva característica do motor. Sob cargas mecânicas elevadas, os motores de BLDC e os motores escovados de alta qualidade são equivalentes em eficiência.

A desvantagem principal do motor sem escovas é o custo mais elevado, a qual se deve a dois fatores: primeiramente, estes motores requerem dispositivos MOSFET de alta potência na fabricação do controlador eletrônico de velocidade. Por outro lado, os motores de corrente contínua com escovas podem ser regulados por um resistor variável simples (potenciômetro ou reostato), mesmo sendo ineficiente, ele também pode ser satisfatório para algumas aplicações dependendo do custo-benefício. Os motores BLDC necessitam de um controlador eletrônico de velocidade para oferecer o mesmo tipo de controle variável.

2.2. Ponte H

Para acionar este tipo de motor é necessário um circuito acionador (driver) e um circuito controlador. O primeiro é basicamente um inversor/amplificador que transforma corrente contínua em corrente alternada e o segundo é responsável por gerar sinais de acionamento e monitorar o funcionamento do motor.

O driver a ser utilizado consiste de três meias-pontes H idênticas. Um diagrama simplificado do circuito acionador é mostrado na figura 5.

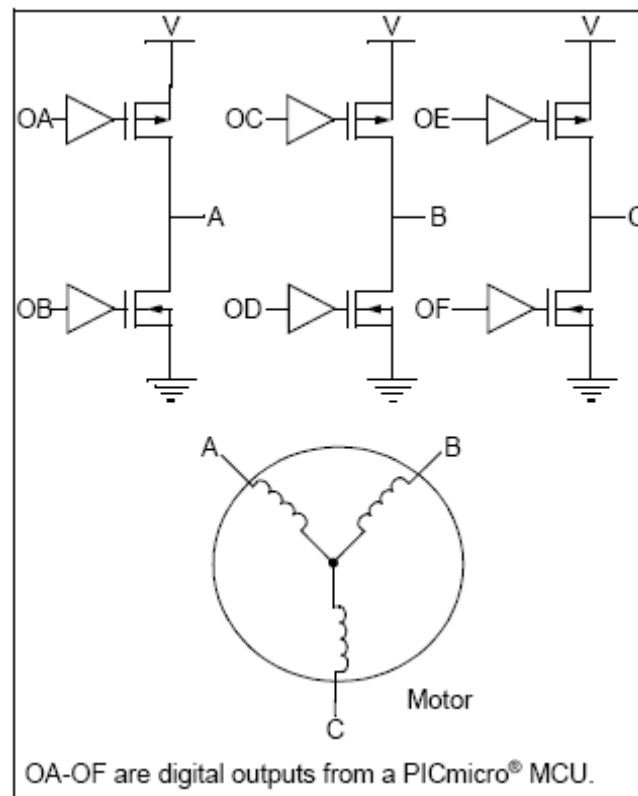


Figura 5 – Diagrama simplificado do circuito acionador

Fonte: Microchip (2004)

2.2.1 Transistores

2.2.1.1 MOSFET

O transistor MOSFET (transistor de efeito de campo de semicondutor de óxido metálico), é, de longe, o tipo mais comum de transistores de efeito de campo em circuitos tanto digitais quanto analógicos.

A palavra "metal" no nome é um anacronismo vindo dos primeiros chips, onde as comportas (gates) eram de metal. Os chips modernos usam comportas de polisilício, mas ainda são chamados de MOSFETs. Um MOSFET é composto de um canal de material semicondutor de tipo N ou de tipo P e é chamado respectivamente de NMOSFET ou PMOSFET. Geralmente o semicondutor escolhido é o silício, mas

alguns fabricantes, principalmente a IBM, começaram a usar uma mistura de silício e germânio (SiGe) nos canais dos MOSFETs.

O terminal de comporta é uma camada de polisilício (silício policristalino) colocada sobre o canal, mas separada do canal por uma fina camada de dióxido de silício isolante. Quando uma tensão é aplicada entre os terminais comporta (gate) e fonte (source), o campo elétrico gerado penetra através do óxido e cria uma espécie de "canal invertido" no canal original abaixo dele. O canal invertido é do mesmo tipo P ou tipo N, como o da fonte ou do dreno, assim, ele cria um condutor através do qual a corrente elétrica possa passar. Variando-se a tensão entre a comporta e a fonte, modula-se a condutividade dessa camada e torna-se possível controlar o fluxo de corrente entre o dreno e a fonte.

A figura 6 mostra o corte transversal de um MOSFET.

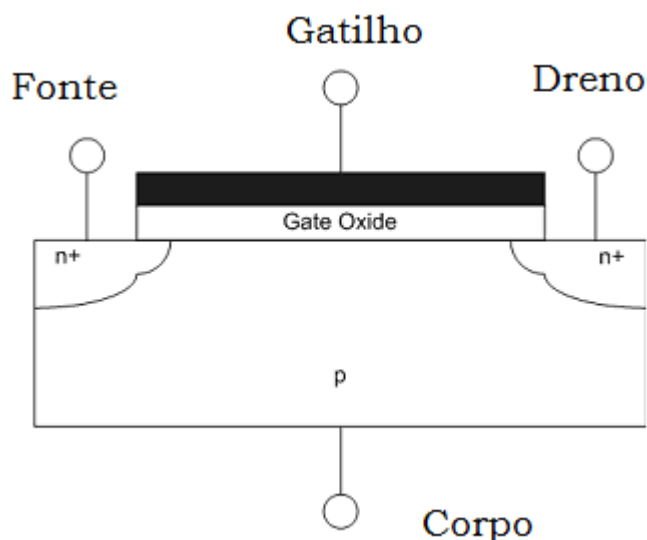


Figura 6 - Corte transversal de um MOSFET tipo N (NMOS)

Fonte: Wikipédia (2008)

A operação de um MOSFET pode ser dividida em três diferentes modos, dependendo das tensões aplicadas sobre seus terminais. Para o NMOSFET os modos são:

- Região de Corte: quando $V_{gs} < V_{th}$

onde V_{gs} é a tensão entre a comporta (gate) e a fonte (source) e V_{th} é a Tensão de threshold (limiar) de condução do dispositivo. O transistor permanece desligado, e



não há condução entre o dreno e a fonte. Enquanto a corrente entre o dreno e fonte deve idealmente ser zero devido à chave estar desligada, há uma fraca corrente invertida.

- Região de Triodo (ou região linear): quando $V_{gs} > V_{th}$ e $V_{ds} < V_{gs} - V_{th}$

onde V_{ds} é a tensão entre dreno e fonte. O transistor é ligado, e o canal que é criado permite o fluxo de corrente entre o dreno e fonte. O MOSFET opera como um resistor, controlado pela tensão na comporta. A corrente do dreno para a fonte é ,

$$I_D = \frac{\mu_n C_{ox}}{2} \frac{W}{L} (2(V_{gs} - V_{th})V_{ds} - V_{ds}^2) \quad (\text{Eq. 1})$$

- Região de Saturação: quando $V_{gs} > V_{th}$ e $V_{ds} > V_{gs} - V_{th}$

O transistor fica ligado, e um canal que é criado permite o fluxo de corrente entre o dreno e a fonte. Como a tensão de dreno é maior do que a tensão na comporta, uma parte do canal é desligado. A criação dessa região é chamada de pinçamento (*pinch-off*). A corrente de dreno é agora relativamente independente da tensão de dreno (numa primeira aproximação) e é controlada somente pela tensão da comporta de tal forma que :

$$I_D = \frac{\mu_n C_{ox}}{2} \frac{W}{L} (V_{gs} - V_{th})^2 \quad (\text{Eq. 2})$$

Em circuitos digitais, os MOSFETs são usados somente em modos de corte e de saturação. O modo de triodo é usado mais em aplicações de circuitos analógicos.

2.2.1.2 IGBT

O transistor bipolar de porta isolada (IGBT) destaca-se pelas características de baixa queda de tensão no estado ligado do BJT com as excelentes características

de chaveamento, que traz um circuito de acionamento da porta bem simplificado e com alta impedância de entrada do mosfet. Existem no mercado transistores IGBTs com os valores nominais de corrente e de tensão bem acima dos valores encontrados para Mosfets de potência.

Os IGBTs estão gradativamente substituindo os mosfets que se dizem em aplicações de alta tensão, onde as perdas na condução precisam ser mantidas em valores baixos. Mesmo que as velocidades de chaveamento dos IGBTs sejam maiores (até 50 kHz) do que as do BJTs e as do mosfets.

Ao contrário do ocorrido no MOSFET, o IGBT não tem nenhum diodo reverso internamente, sendo assim este fator torna sua capacidade de bloqueio para tensões inversas muito baixa, podendo suportar uma tensão inversa máxima em menos de 10 volts.

A operação do IGBT é muito similar à dos MOSFETs de potência. Para colocá-lo no estado ligado, basta polarizá-lo positivamente no terminal do coletor (C+) em relação ao terminal do emissor (E -). De igual maneira, uma tensão positiva V_G aplicada na porta (G) fará o dispositivo passar para o estado ligado (ON), quando a tensão no gate (G) exceder a tensão de limiar. O IGBT passara para o estado desligado (OFF) quando houver o corte de tensão do terminal da porta (G).

2.2.1.3 BJT

O Transistor de Junção Bipolar, TJB (BJT), é um dispositivo semicondutor, composto por três regiões de semicondutores dopados (base, coletor e emissor), separadas por duas junções p-n. A junção p-n entre a base e o emissor tem uma tensão de barreira (V_0) de 0,6 V, que é um parâmetro importante do TJB (BJT). Contrariamente ao transistor de efeito de campo, TEC (FET), no qual a corrente é produzida apenas por um único tipo de portador de cargas (elétrons ou lacunas), no TJB (BJT) a corrente é produzida por ambos os tipos de portadores de cargas (elétrons e lacunas), daí a origem do nome bipolar.

Existem dois tipos de TJB (BJT): npn e pnp. O tipo npn consiste em duas regiões n separadas por uma região p. O tipo pnp consiste em duas regiões p

separadas por uma região n. As Figuras 7 e 8 representam os seus respectivos símbolos esquemáticos.

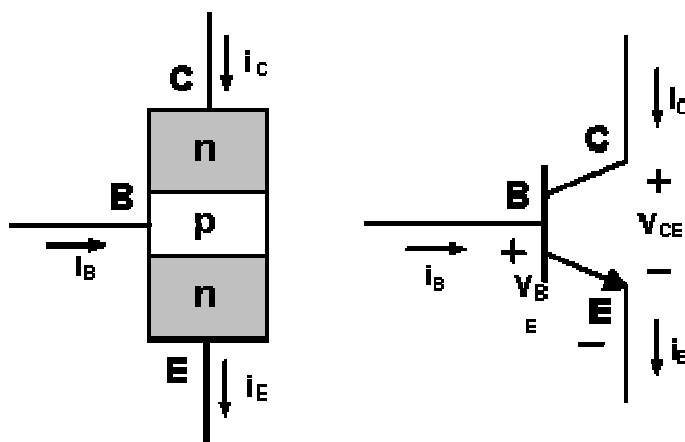


Figura 7 - Símbolo esquemático de um TJB, npn

Fonte: Museu das Comunicações (2008)

O TJB (BJT) opera em três modos diferentes: modo de corte, modo de amplificação linear e modo de saturação.

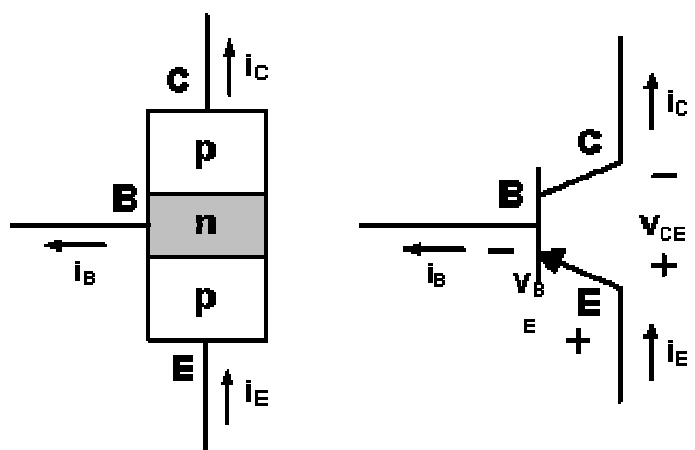


Figura 8 - Símbolo esquemático de um TJB, pnp

Fonte: Museu das Comunicações (2008)



2.3. PWM

A modulação por largura de pulso é largamente utilizada no controle de velocidade de motores. Contudo, para obter velocidade variável de um motor BLDC, deve-se aplicar uma tensão variável aos terminais dos enrolamentos. Colocando isso em termos digitais, a tensão variável pode ser obtida por diferentes ciclos ativos (*duty cycles*) de um sinal PWM aplicado aos enrolamentos do motor BLDC.

O módulo da geração de Modulação por Largura de Pulso (PWM) é um poderoso recurso, muito utilizado para o controle de motores. A partir dele é possível gerar um sinal analógico (tensão variável), apesar de a saída ser um sinal digital que assume apenas os níveis lógicos: alto (um) e baixo (zero). A saída gerada é uma onda quadrada, com a frequência constante e uma largura de pulso variável. Esses conceitos estão diretamente relacionados com o período fixo e o *Duty Cycle* respectivamente.

A frequência de uma onda pode ser definida como o número de ciclos por unidade de tempo. O tempo decorrido para um ciclo é definido como período.

O *Duty Cycle* define a duração do sinal em nível lógico alto em um ciclo. Assim, quando se tem um *Duty Cycle* de 100%, tem-se nível lógico alto por todo o período e quando se tem 0%, tem-se nível lógico baixo por todo o período. Um *Duty Cycle* de 50% define a metade do período em nível lógico alto e a outra metade em nível lógico baixo, dando uma tensão média. Se se opera em níveis TTL, em que a tensão em nível alto é 5V, esse ciclo ativo fornece uma tensão média de 2,5V [Oliveira, 2006]. Estes conceitos são demonstrados na figura 9:

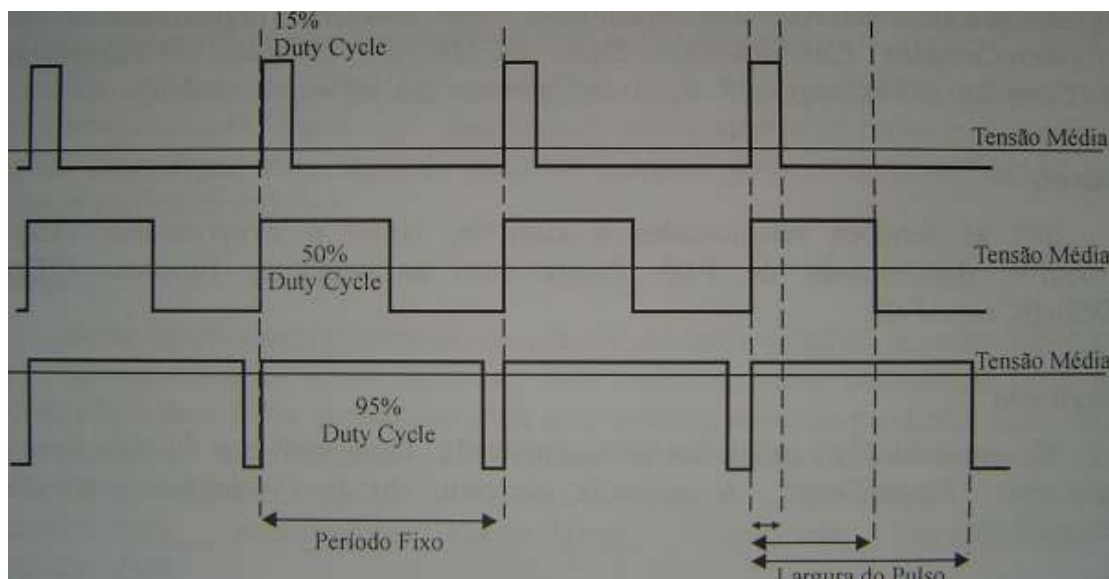


Figura 9 – Sinais de Modulação por Largura de Pulso (PWM)

Fonte: Oliveira, 2006

2.4. Controladores de Sinal Digital

O circuito controlador terá como base o microcontrolador DSPIC.

A Microchip possui uma família de microcontroladores chamada de dsPIC. Dentre as diversas vantagens desta família frente às famílias PIC16 e PIC18 pode-se citar algumas, entre outras:

- MCU de 16 bits;
- Poder de processamento de um DSP, utilizando a arquitetura e instruções dedicadas;
- A idéia da Microchip é desenvolver ferramentas de suporte para simplificar aplicações sem a necessidade de muita base matemática. Um exemplo é o software para projeto de filtros que já gera o código C ou assembly e os AN;
- 86% das instruções são executadas em 1 ciclo de máquina (algumas instruções como divisão especial em 18 ciclos);

- Flexibilidade de clock que possui *PLL*, divisor de frequência, RC interno e possibilidade de oscilação por cristal do timer1;
- Proteção da memória. Como o componente pode se auto-programar, se o dsPIC tentar gravar em uma área de memória protegida, uma interrupção de *Trap* é gerada;

A seguir o diagrama de pinos do microcontrolador a ser utilizado, o DSPIC30F3010.

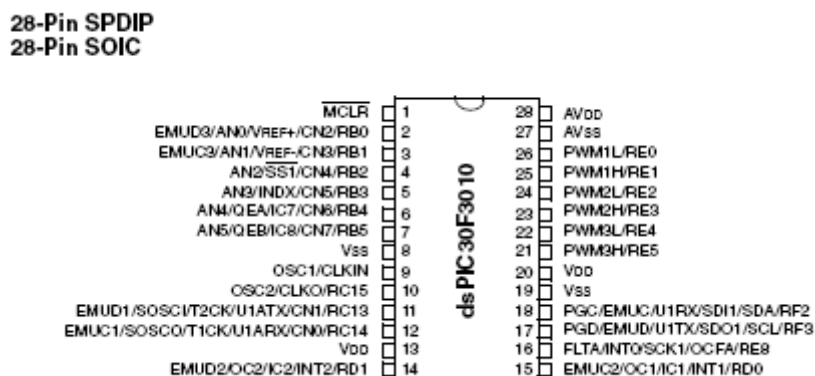


Figura 10 – Diagrama de Pinos do DSPIC30F3010

Fonte: Microchip (2005)

Microcontroladores (*MCUs*) e Processadores de Sinal Digital (*DSPs*) têm se adequado historicamente a diferentes categorias de aplicação.

Um Controlador de Sinal Digital (*DSC*) é um controlador desenvolvido que integra a força de processamento de algoritmo de um DSP com a robustez, habilidades de controle em tempo real de um microcontrolador. *MCUs* oferecem tipicamente uma solução em um chip único. *DSPs* freqüentemente necessitam de memória externa, um *FPGA*, um controlador, e outra lógica colada.

O mercado dos *DSCs* foi inicialmente direcionado ao controle de motores – para obter velocidades mais rápidas ou redução do custo do motor. Mas seu uso pode também se dar em aplicações de controle tais como comandos de voz, reconhecimento de fala, modems, criptografia, reconhecimento de padrões, etc.

O DSC está emergindo como um segmento de mercado próprio, tal como o MCU evoluiu do microprocessador para sua própria categoria de produto.

Microcontroladores desenvolvem-se sobre interrupções, DSPs as evitam. A primazia de um MCU é o controle em tempo real e resposta determinística. A primazia de um DSP é a performance. A tabela 1 lista os atributos do MCU, DSP e DSC.

Tabela 1 – Atributos do MCU, DSP e DSC

Fonte: Microchip (2005)

Atributo	MCUs	DSPs	DSCs
Executável da Flash	x		x
Set de Registro Grande	x		x
Capacidade Interrupção Robusta	x		x
Abundante mistura de sinal	x		x
MAC de ciclo único		x	x
Alcance Operando Duplo		x	x
Elevado <i>Loops</i> por zero		x	x
Saturação/ <i>Rounding</i>		x	x
Modos <i>bit</i> -reverso		x	x

2.5. Encoder Incremental

Um encoder incremental é feito de 2 partes principais: o disco e o sensor. O disco de um encoder incremental é estampado com linhas idênticas repetidas perto da beira exterior do disco. O número de linhas pares no disco determina a resolução do encoder. Praticamente todos os encoders incrementais produzem saída em quadratura as quais indicam a velocidade, ângulo e direção do eixo de transmissão. Encoders incrementais são comumente utilizados como equipamentos de retorno para controladores de motor.

Praticamente todos os encoders incrementais fornecem sinais de saída em quadratura. Um encoder de disco ótico de trilha simples pode somente fornecer informação de velocidade. Adicionando uma segunda trilha ao disco ótico e ajustando ele para $\frac{1}{4}$ de ciclo da primeira trilha fornece duas vantagens:

1) Direção de rotação pode ser determinada examinando a fase relativa dos dois canais (se A está à frente de B ou B está à frente de A).

2) A resolução do encoder é efetivamente aumentada já que há agora 4 transições para cada par de linha, ao invés de 2.

No microcontrolador utilizado neste projeto, o módulo QEI (Quadrature Encoder Interface) fornece a interface para os encoders incrementais obterem dados de posição mecânica.

Os dois canais, Fase A (QEA) e Fase B (QEB), tem uma relação única. Se Fase A está à frente da Fase B, então a direção (do motor) é julgada positiva ou para frente. Se Fase A está atrás da Fase B, então a direção é julgada negativa ou para trás.

Um terceiro canal, chamado pulso de referência, ocorre uma vez por revolução e é utilizado como uma referência para estabelecer uma posição absoluta.

A figura 11 exemplifica o encoder.

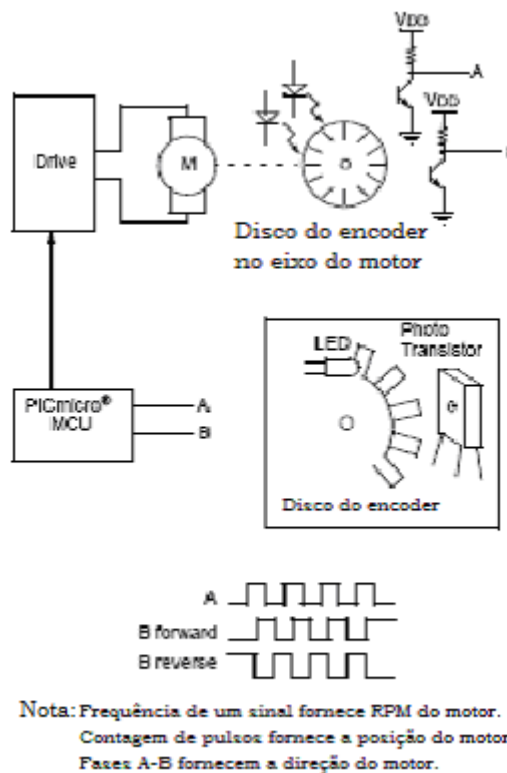


Figura 11 – Detecção ótica de Velocidade/Direção/Posição

Fonte: Microchip (2004)

2.6. Controle Proporcional Integral

Essa ação é definida por: (K. Ogata, 2005).

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt \quad (\text{Eq. 3})$$

ou, então, a função de transferência do controlador é:

$$\frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} \right) \quad (\text{Eq. 4})$$

onde T_i é chamado de tempo integrativo, K_p é o ganho proporcional.

Uma maneira de implementar um compensador integral ideal, ou controlador PI, está mostrado na figura 12.

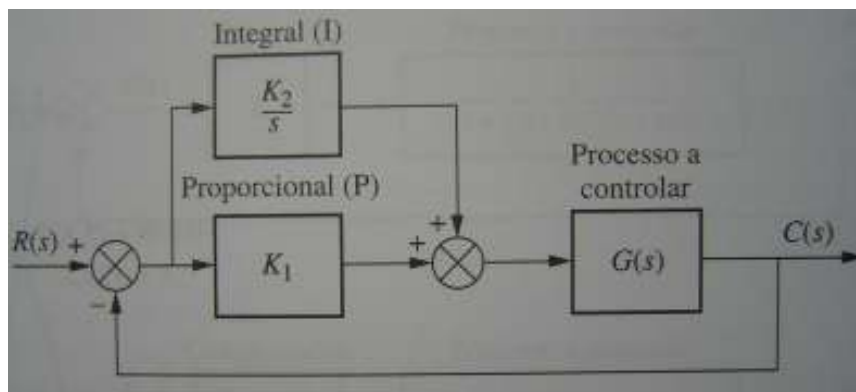


Figura 12 – Compensador Proporcional Integral

Fonte: Nise, 2000

Os efeitos de um controlador PI são:

- 1) Adiciona um zero em $S = -K_I/K_P$ no caminho direto
- 2) Adiciona um pólo em $S = 0$, o sistema tem o tipo aumentado por 1



- 3) Um controlador PI é essencialmente um passa-baixa
- 4) Se o zero em $-KI/KP$ é selecionado corretamente pode-se melhorar o amortecimento e o erro em regime permanente.
- 5) Sistemas compensados terão um tempo de subida maior.
- 6) Diminui a largura de banda
- 7) Melhora a margem de ganho e a margem de fase
- 8) Filtra ruídos de alta frequência
- 9) Uma aproximação para a localização do zero é fazê-lo o menor possível. Com isto ele ficará próximo da origem e longe dos outros pólos significantes do sistema, e os valores de KP e de KI serão relativamente pequenos.

3. MATERIAIS E MÉTODOS

3.1. Descrição Geral do Projeto

O projeto proposto consiste em gerar um sinal PWM com *duty cycle* variável, para controlar a variação de velocidade de um motor síncrono de ímã permanente. Um encoder acoplado ao eixo do rotor lê esta velocidade, e transfere a informação para um microcontrolador. Para esse projeto, um controle em malha fechada (proporcional integral) foi implementado.

A interface entre o microcontrolador e a etapa de potência se dá através de um circuito constituído de três meias ponte H, com uma lógica digital na entrada para isolar eletricamente o microcontrolador da etapa de potência e evitar que um comando errôneo do microcontrolador provoque curto-circuito na alimentação do motor.

A figura 13 mostra um diagrama de blocos simplificado do sistema implementado.

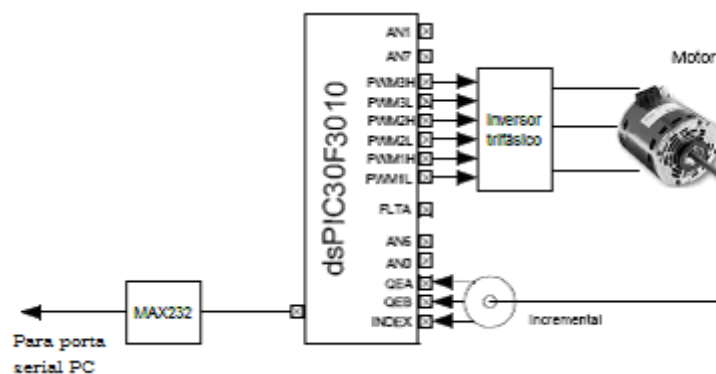


Figura 13 – Diagrama de blocos simplificado do sistema

A implementação do sinal de estímulo da variação de velocidade se dará por um potenciômetro, o qual será lido com uso do AD do microcontrolador.

O controle PI será ativado por software.

Para resolver o problema de engenharia, o primeiro passo foi definir o circuito de acionamento para o motor, o qual deveria ser adequado para acionamento de um motor síncrono de ímã permanente trifásico e suportar os valores de tensão e corrente característicos do motor empregado. Foi realizado simulação de funcionamento do circuito de acionamento no software Proteus, antes de ser implementado o circuito fisicamente.

A seguir definiu-se o microcontrolador a ser utilizado, partindo da premissa que o microcontrolador deveria ser específico para controle de motores e ter entrada para os sinais do encoder.

3.2. Descrição dos Sistemas Mecânicos

O motor utilizado neste projeto é um motor que estava disponível na própria instituição de ensino, do qual não se tinha nenhum dado, como tensão, velocidade e corrente máxima. A certeza era que se tratava de um motor síncrono de ímã permanente. Na figura 14 tem-se uma foto do motor.



Figura 14 – Motor utilizado

Ao abrir o motor, constatou-se que se tratava de um motor de 4 pólos, de núcleo circular com ímãs na periferia. A figura 15 mostra uma foto do motor aberto.

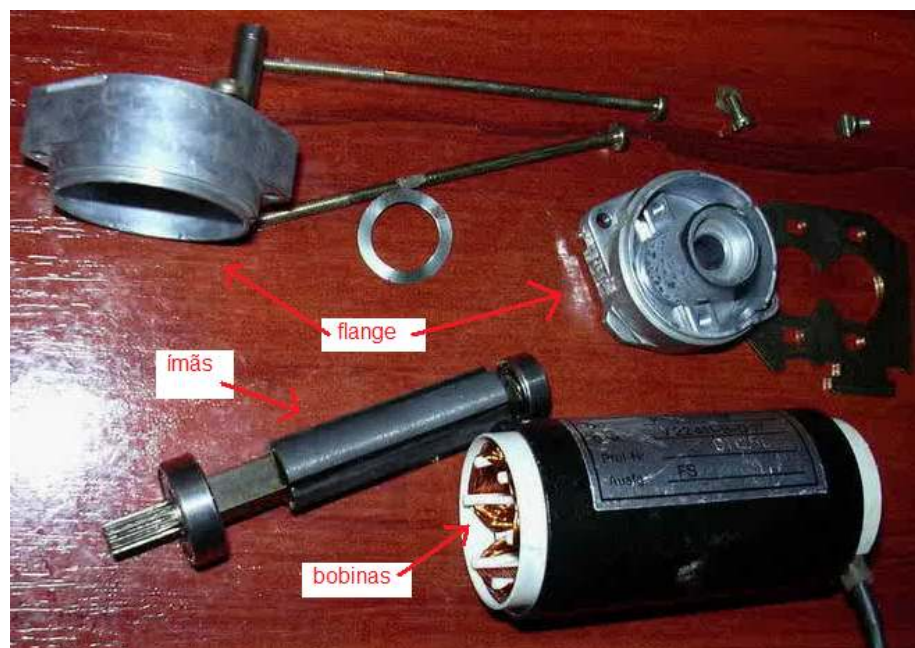


Figura 15– Foto do motor aberto

A figura 16 ilustra o rotor circular com ímãs na periferia.

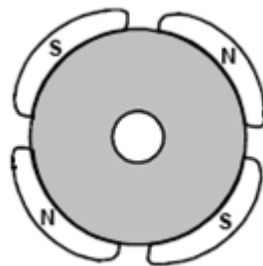


Figura 16 – Rotor circular com ímãs na periferia

Fonte: Microchip (2003)

A máquina síncrona de 4 pólos tem como característica:

$$\theta_m = 90^\circ$$

$$\theta_e = 180^\circ$$

Onde θ_m é o ângulo mecânico e θ_e é o ângulo elétrico.

Isto significa que para cada ciclo mecânico do motor têm-se dois ciclos elétricos.

Para definir uma posição de referência do rotor em relação ao estator foi necessário alinhar o estator com o rotor, através da alimentação de duas bobinas com a tensão de operação do motor (12V). Ao realizar esta operação, o rotor alinha-se com o estator. Para determinar o ponto inicial (90° entre rotor/estator), foi necessário retornar a posição do rotor em 45° mecânicos. Este ponto foi alinhado com o pulso de referência do encoder para ser possível saber a posição do rotor a cada instante.

3.3. Descrição dos Sistemas Eletroeletrônicos

3.3.1. Encoder de Quadratura

O encoder de quadratura utilizado neste projeto é o modelo CDD 50, do fabricante STEGMANN. O mesmo possui 4096 pulsos por revolução. Fornece informação de velocidade e posição do motor através dos pinos de sinal A e B e do pino de sinal de referência INDX.

Os dois canais, Fase A (QEA) e Fase B (QEB), estão inter-relacionados. Se a fase A está à frente da fase B, então a direção do motor é julgada positiva ou para frente. Se a fase B está à frente da fase A, então a direção do motor é julgada negativa ou reversa. Um terceiro canal, chamado *Index Pulse*, ocorre uma vez por revolução e é usado como uma referência para estabelecer uma posição absoluta.

A figura 17 mostra os sinais do encoder de quadratura.

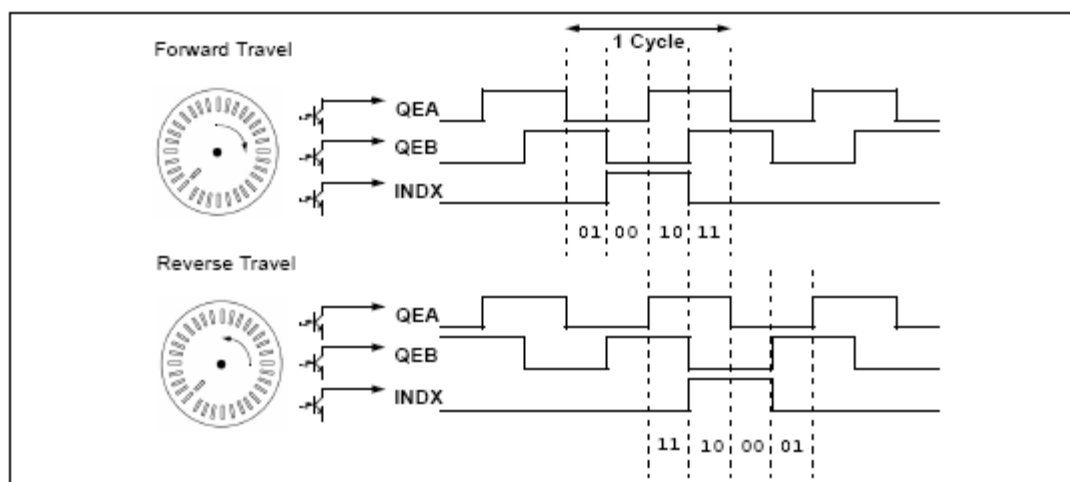


Figura 17 – Sinais de interface do encoder de quadratura

Fonte: Microchip (2005)

Foram realizados testes de funcionamento do encoder, o qual após alimentado corretamente com +5V foi monitorado no osciloscópio os sinais QEA, QEB e INDX, tanto na direção para frente como reversa, o mesmo apresentou funcionamento perfeito.

3.3.2. Ponte H

Neste projeto foram utilizadas três meias ponte H para o acionamento do motor trifásico.

Os dispositivos para chaveamento utilizados nas pontes foram os Mosfet's IRF9640 e IRF640, os quais suportam 11A/200V e 18A/200V, respectivamente.

Como mencionado anteriormente, o papel da lógica digital na entrada é isolar o microcontrolador da etapa de potência e evitar que um comando errôneo do microcontrolador danifique o circuito de potência.

A figura 18 mostra o circuito implementado para o acionamento do motor trifásico.

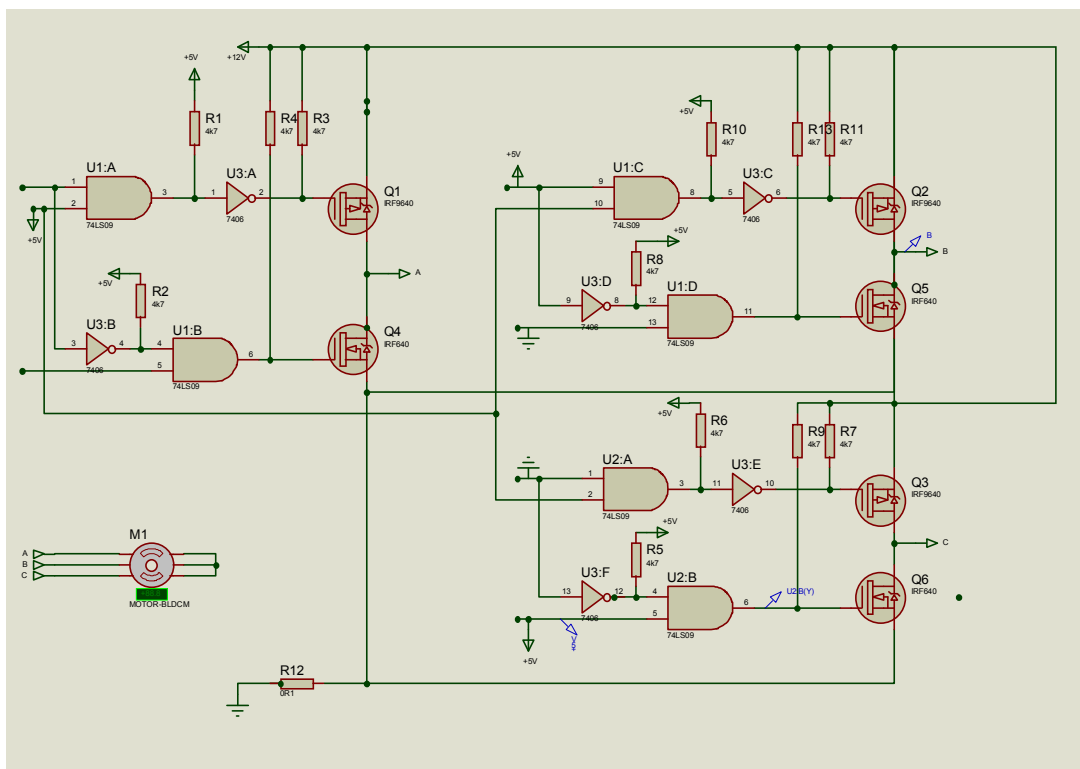


Figura 18 – Circuito de acionamento motor trifásico

Os pinos AL, AH, BL, BH, CL e CH são conectados respectivamente aos pinos 26, 25, 24, 23, 22 e 21 do microcontrolador. Onde AL, BL e CL são os acionamentos dos MOSFET'S que chaveiam a tensão negativa das meias pontes, enquanto AH, BH e CH chaveiam a tensão positiva. Os pinos 26 a 21 do microcontrolador são os pinos de controle PWM do DSPIC. O pino ENAB/PWM é deixado em nível lógico alto (+5V).

Os pontos A, B e C são ligados aos centros das 3 meias pontes, que são conectadas ao motor em estrela, como mostra a figura 19.

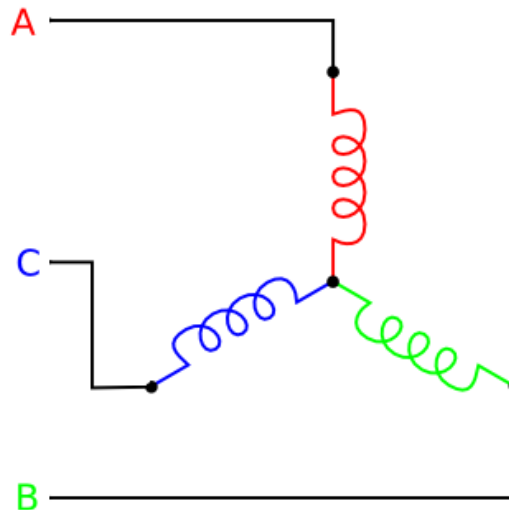


Figura 19– Ligação dos enrolamentos

3.4. Descrição dos Sistemas Informáticos e Computacionais

O desenvolvimento do programa foi realizado utilizando-se as ferramentas de desenvolvimento da Microchip, MPLAB IDE v8.10 e a linguagem de programação utilizada foi a linguagem C conjuntamente com o compilador C30.

3.4.1. Firmware

3.4.1.1 PWM

O microcontrolador DSPIC30F3010 possui um módulo dedicado à modulação por largura de pulso, o MCPWM (Motor Control Pulse Width Modulation).

Há diversos modos os quais o módulo MCPWM pode ser configurado. Saída alinhada na borda é provavelmente o modo mais comum. A figura 20 descreve a operação de um PWM alinhado por borda. No início do período, as saídas vão para nível 1. Como o PTMR (registrador que armazena o valor da contagem) incrementa, uma combinação com os registradores *duty cycle* ocasiona a saída do *duty cycle* correspondente ir para nível 0, desse modo marcando o final do *duty cycle*. Quando o PTMR alcançar o valor setado no registrador PTPER (registrador onde é setado o período PWM), um novo período é iniciado e todas as saídas vão para nível 1 para início de um novo ciclo completo. [AN957]

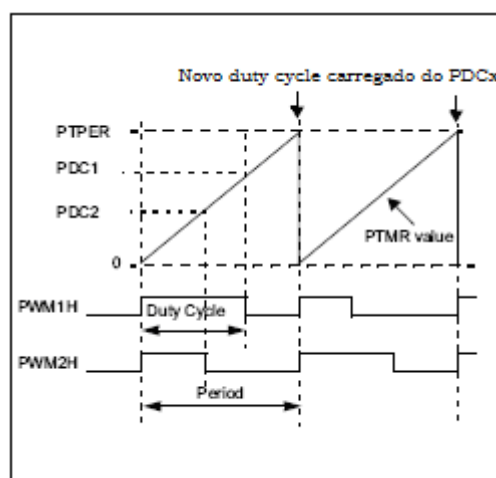


Figura 20 – PWM alinhado na borda

Fonte: Microchip (2004)

O firmware desenvolvido deverá seguir uma sequência de acionamento das saídas PWM, afim de comutar os enrolamentos ativados na ordem correta.

Esta sequência de acionamento pode ser vista na figura 21.

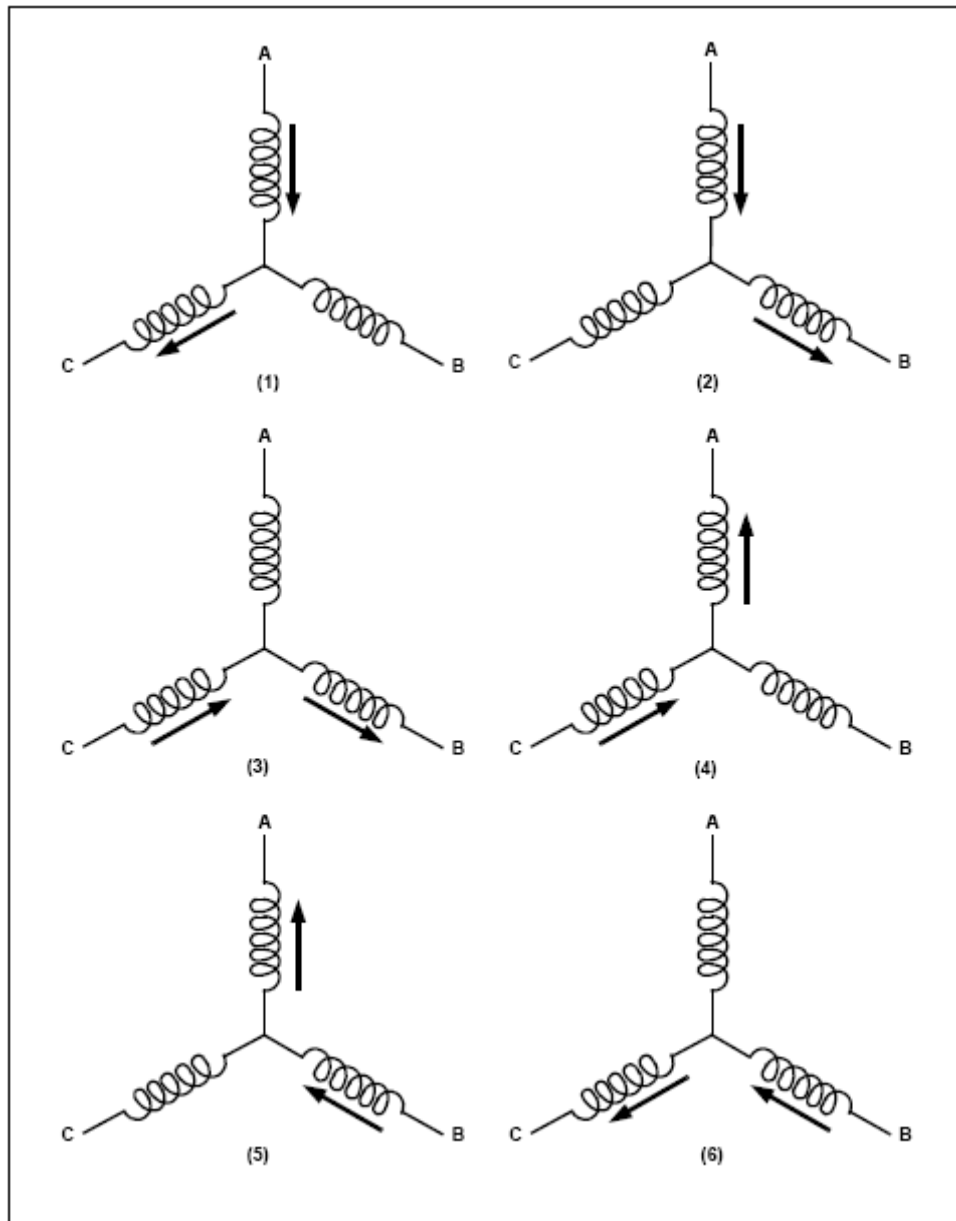


Figura 21 – Sequência de energização dos enrolamentos

Fonte: Microchip (2004)

Para isso, foi desenvolvido um software que lê a posição do encoder e energiza as 3 bobinas (2 a 2) em seis passos dentro de um ciclo elétrico.

Interfaces de I/O e Fluxogramas

O fluxograma da figura 22 descreve a rotina de inicialização do encoder com o envio do dado de velocidade do motor para porta serial de um

microcomputador. O envio de dados pela porta serial foi utilizado para depurar o software já que não se tinha display no projeto.

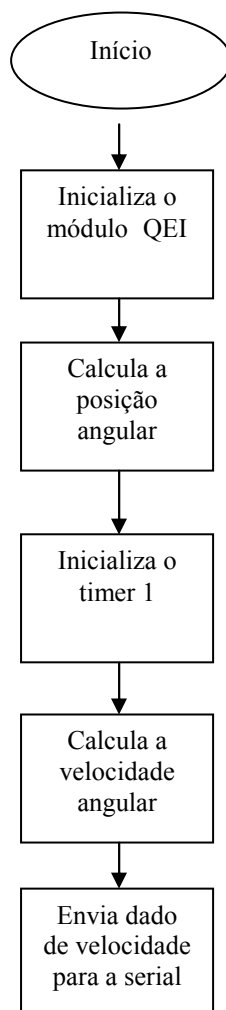
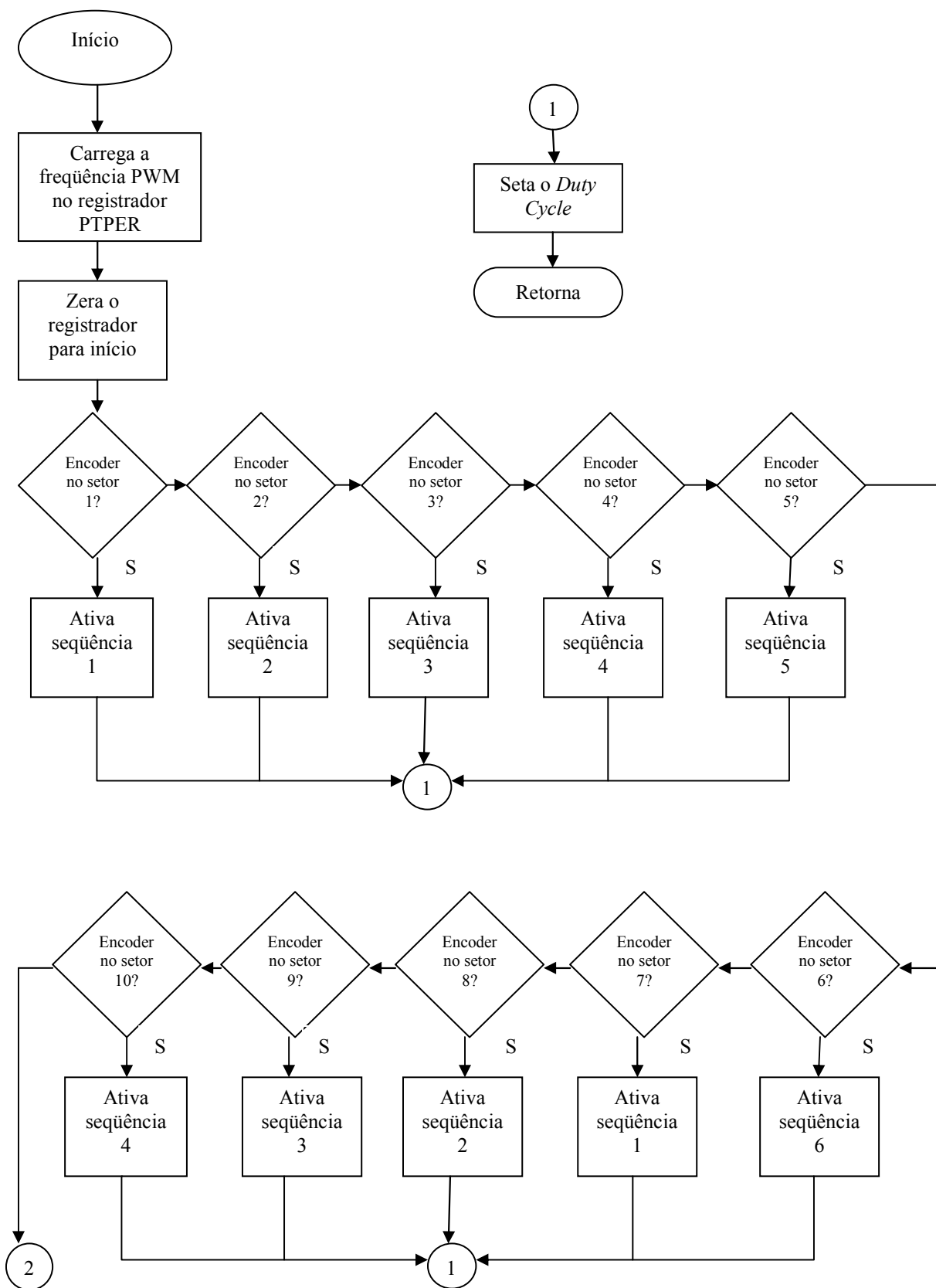


Figura 22 – Fluxograma do encoder com o envio pela serial

O fluxograma do acionamento PWM é mostrado na figura 23, de acordo com a posição do encoder.



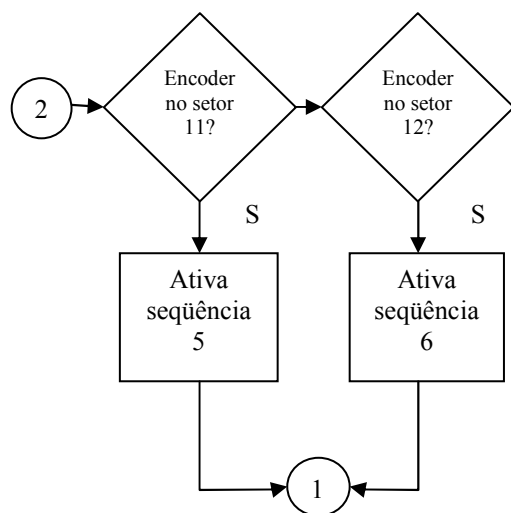


Figura 23 – Fluxograma do PWM de acordo com a leitura do encoder

Descrevendo a figura 23, primeiro é setado o período PWM no registrador PTPER, zerado o registrador PWMCON1 para início. Então é lido a posição do encoder para energização dos enrolamentos corretos. Conforme o motor é acionado, a posição do rotor muda e o encoder troca de setor, energizando os enrolamentos pertencentes a cada setor. O Duty Cycle seta o tempo em que se tem nível 1 durante o período.

A velocidade será controlada por software, através de um controle proporcional-integral, que tem como *set point* a velocidade setada no potenciômetro (velocidade desejada). O software compara a velocidade atual lida através do encoder com a desejada. A diferença entre a velocidade desejada e a velocidade atual é o erro, que será compensado através dos ganhos K_p e K_i definidos por software. O compensador atuará no *Duty Cycle* a fim de manter a velocidade desejada.

4. APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

O controle de velocidade foi desenvolvido utilizando-se um microcontrolador DSPIC30F3010, um motor síncrono de ímã permanente disponível na universidade e um encoder de quadratura de 1024 pulsos por revolução. O ajuste do *set point* de velocidade é feito por um potenciômetro, através da utilização de um pino de A/D do microcontrolador. O encoder de quadratura, através de contagens de pulsos, informa ao microcontrolador a posição do rotor a cada instante de tempo e a velocidade angular do motor.

Na figura 24 mostra-se uma fotografia do projeto desenvolvido.

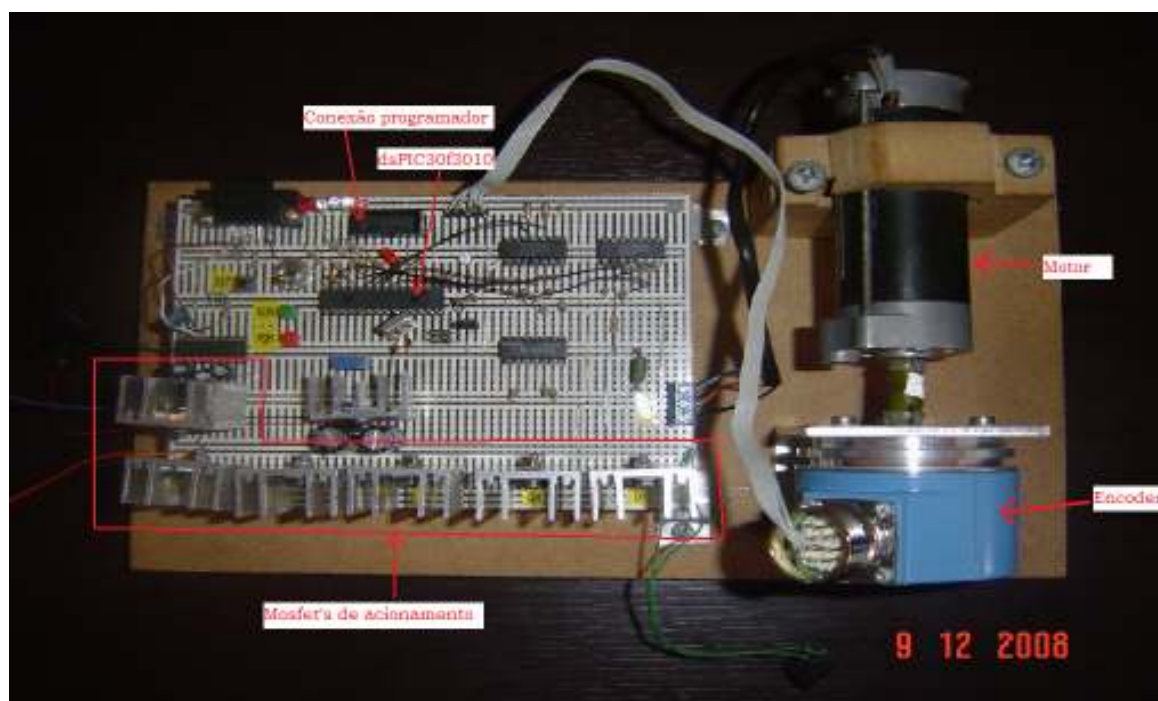


Fig. 24 – Foto do projeto desenvolvido

4.1. Controle da Velocidade pelo Potenciômetro

Na sua forma elementar, um controle em malha fechada consiste em um atuador (motor), um comparador, um amplificador e um sensor (gerador). A figura 25 ilustra estes elementos e como a expressão “malha fechada” é extraída: o amplificador aciona o motor, o qual é acoplado ao gerador; o gerador envia um sinal ao comparador, que compara o sinal de retorno com o sinal de comando; o comparador mantém os dois sinais em balanço dando ao amplificador o comando apropriado. [Electro-Craft Corporation, 1978]

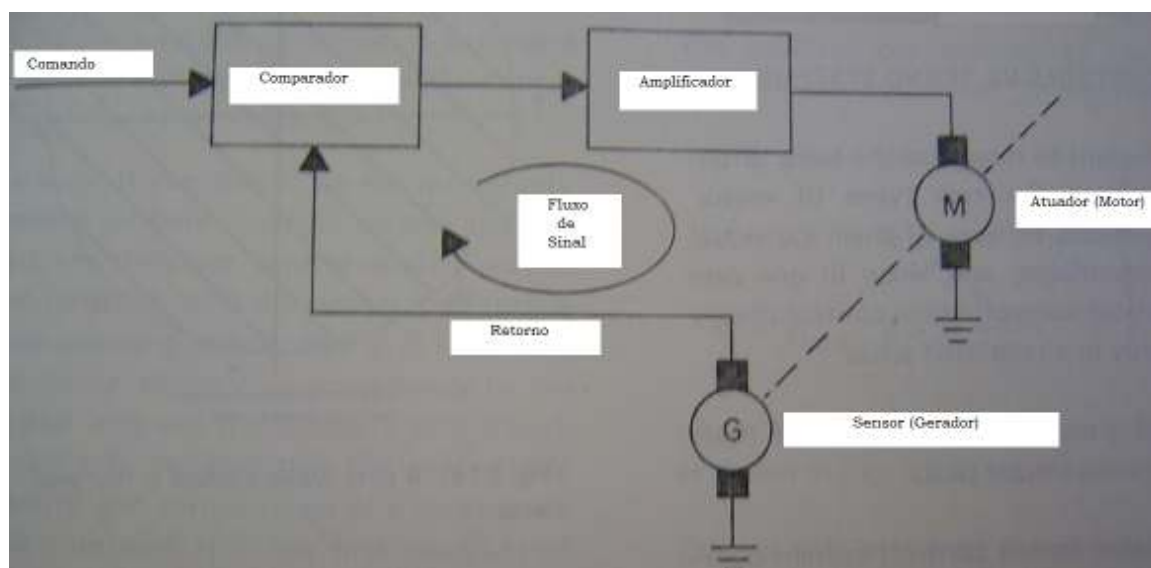


Fig. 25 – Um sistema de controle de velocidade de malha fechada

Fonte: Electro-Craft Corporation (1978)

A velocidade do motor foi determinada a partir da variação do *duty cycle* dentro de um período fixo do PWM. Quanto maior o duty cycle, maior a tensão média disponibilizada ao motor, conseqüentemente, maior a velocidade angular do motor.

Na figura 26 é visto uma amostra obtida com o osciloscópio da forma de onda nos pinos PWM1L e PWM1H do microcontrolador.

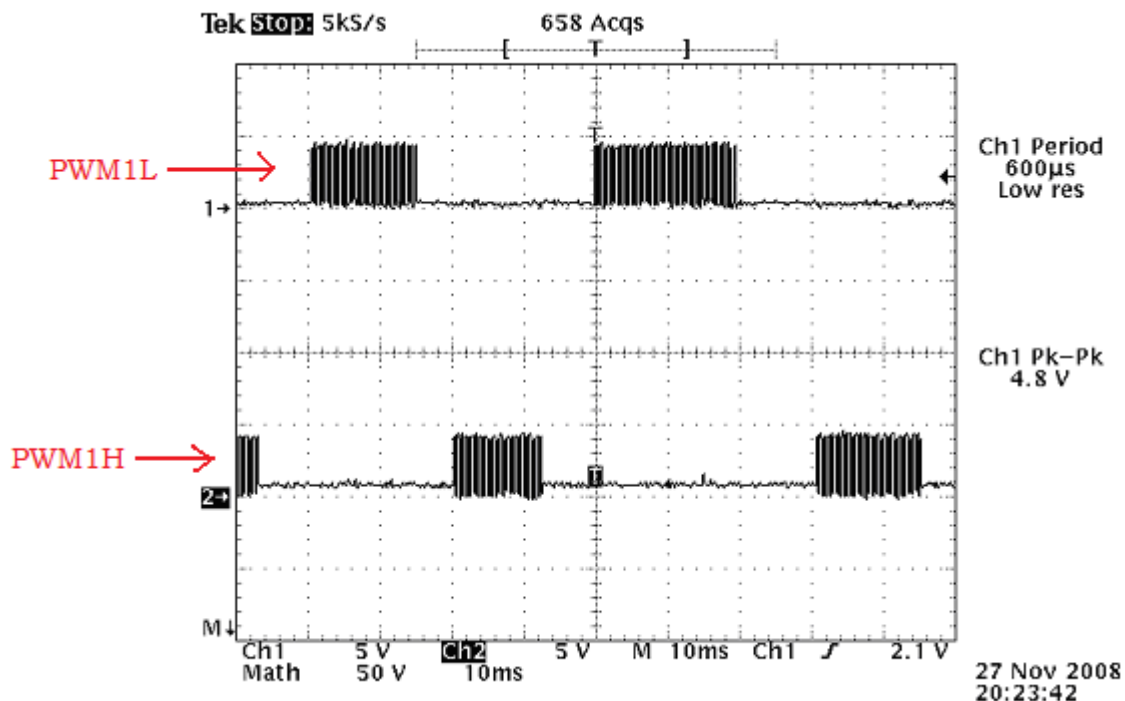


Fig. 26 – Forma de onda pinos PWM

As formas de onda mostradas na figura 26 foram obtidas com o *duty cycle* máximo. A figura 27 mostra a forma de onda da tensão fornecida ao motor, nas bobinas A e B.

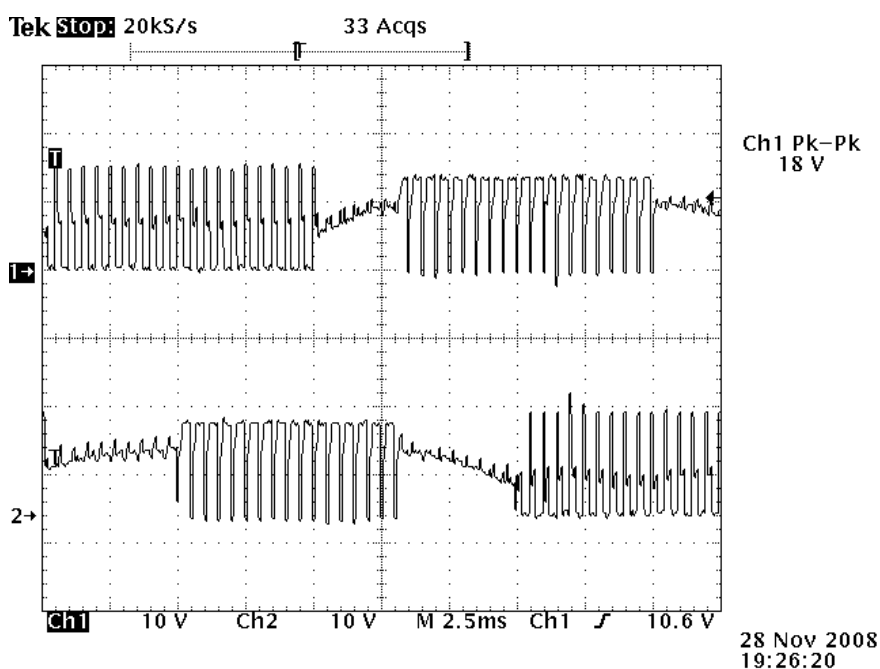


Fig. 27 – Forma de onda da tensão nas bobinas A e B do motor

O set point de velocidade deu-se pelo valor de tensão no pino do A/D do microcontrolador. Um potenciômetro multivoltas faz variar a tensão na entrada do pino de A/D do microcontrolador. O valor de tensão no pino do A/D tem variação de 2,45V até 4,7V. A figura 28 mostra o circuito utilizado para o A/D.

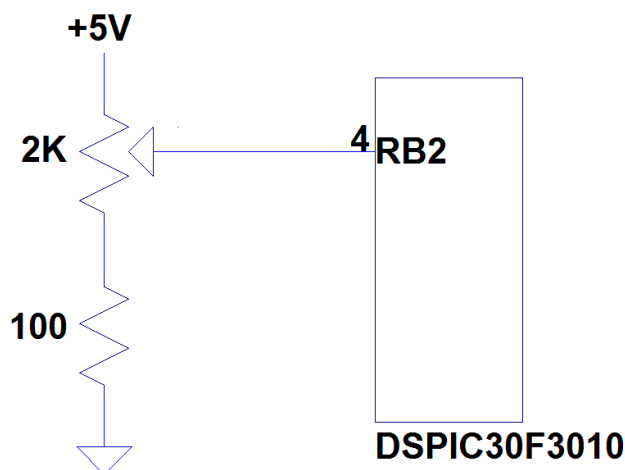


Figura 28 – Circuito utilizado para o A/D

Com o motor a vazio, e repetindo diversos pontos de tensão no A/D, foi medida a velocidade do motor com o tacômetro Lutron DT-2236. Foi repetido o procedimento duas vezes e os valores mostraram-se praticamente iguais ao encontrado anteriormente. A velocidade foi medida através da leitura ótica do tacômetro.

A figura 29 apresenta o gráfico obtido no experimento. Os valores de velocidade angular correspondem ao da primeira medição.

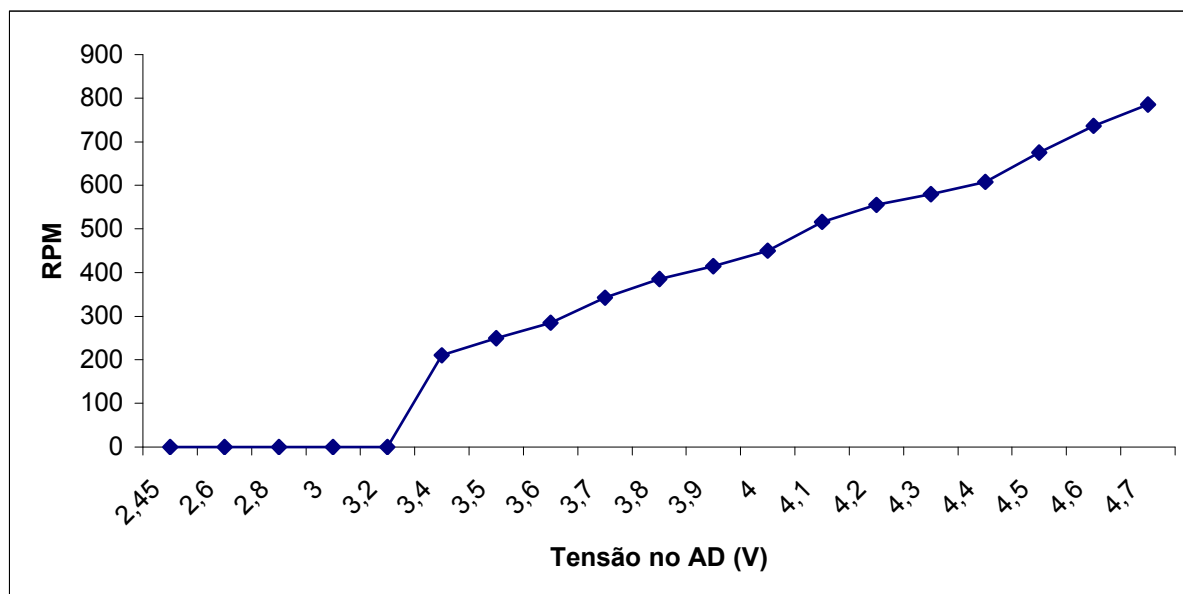


Figura 29 – Gráfico da velocidade em função da tensão no A/D

Nota-se que para o motor começar a girar é necessário uma tensão mínima aplicada em seus enrolamentos, para vencer a inércia. A variação de velocidade deu-se de 210 a 785 rpm, conforme a figura 29.

4.2. Obtenção da Curva de Resposta do Motor

Para obtenção da curva de resposta do motor foi necessário a utilização de uma placa de aquisição de dados, visto a rapidez com que o processo atingia o regime permanente (em torno de 60-70ms).

Para se obter a leitura, utilizou-se um conversor de frequência para tensão (LM2907). O sinal de entrada em frequência utilizado foi o sinal QEA do encoder, e a saída do circuito montado gerou um valor de tensão proporcional à frequência de entrada. Este valor de tensão foi lido pela placa de aquisição MyLabPC, da Novus. Com os valores de tensão e o tempo de amostragem, construiu-se um gráfico com os valores obtidos no experimento.

A figura 30 mostra o circuito utilizado para obtenção do valor de tensão em função do tempo, em resposta ao salto aplicado no motor.

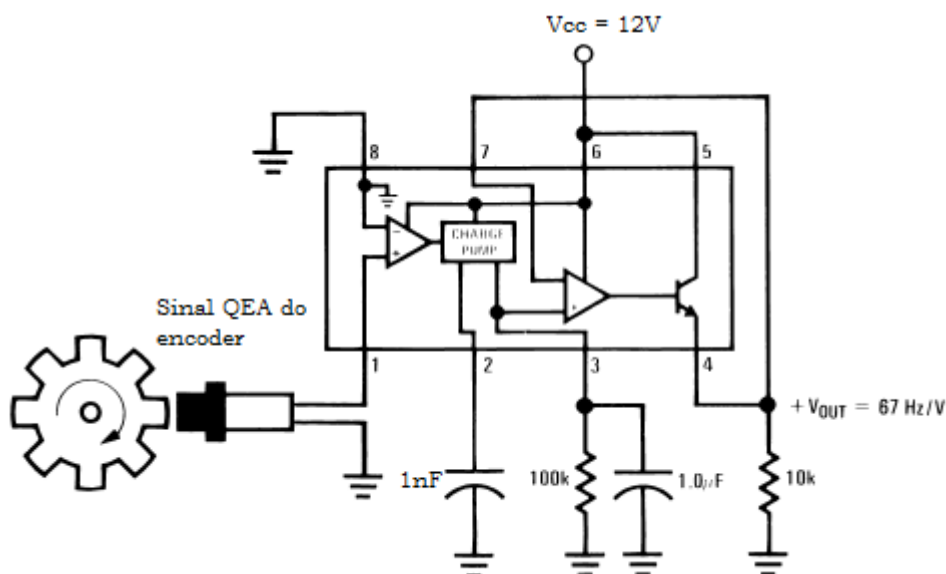


Figura 30 – Circuito para obtenção da tensão em relação à frequência

Fonte: National (2003)

O circuito foi adequado para a frequência de entrada do sistema, através da alteração dos valores de C1 (o capacitor de 1nF) e Vcc.

Para a máxima velocidade do motor foi calculada a frequência máxima de entrada, como segue:

$$\frac{785rpm}{60s} = 13,08rps = 13,08Hz$$

$$13,08 \times 1024 \text{ pulsos} = 13397$$

Assim o circuito teria que ler pelo menos 13397 Hz. Através da utilização da fórmula de máxima frequência disponível no datasheet do LM2907, chegou-se a um valor de 15000Hz com os valores de C1=1nF e Vcc=12V.

$$F_{m\acute{a}x} = \frac{I_2}{C1 \cdot V_{cc}}$$

(Eq. 5)

$$F_{m\acute{a}x} = \frac{180\mu A}{1nF \cdot 12V} = 15000Hz$$

Onde I2 é a corrente de saída.

Na figura 31 mostra-se a placa de aquisição utilizada. A taxa de amostragem da placa de aquisição é de 8 a 128 amostras por segundo, e a resolução do A/D é configurável de 15 a 11 bits.



Figura 31 – Placa de aquisição utilizada

Fonte: Novus (2008)

Na figura 32 mostra-se o gráfico da tensão em relação ao tempo, o qual foi utilizado para o cálculo dos ganhos K_p e K_i através do método 1 de Ziegler-Nichols.

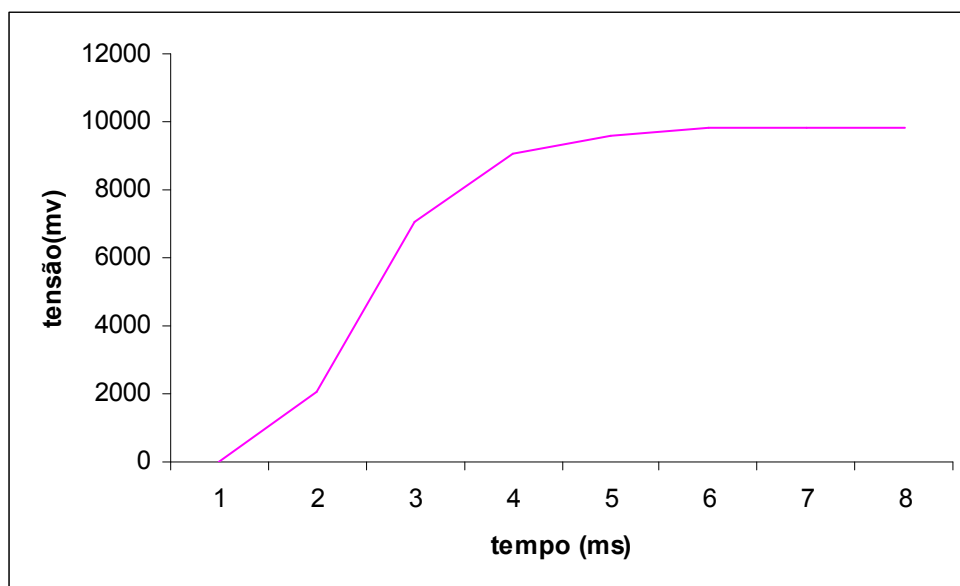


Figura 32 – Gráfico tensão x tempo

Para chegar aos valores de ganho K_p e K_i através do método 1 de Ziegler-Nichols, utilizou-se o método descrito [Ogata, 2005], em que é necessário conhecer o tempo de retardo L e a constante de tempo T .

O tempo de retardo e a constante de tempo podem ser determinados traçando-se uma reta tangente à curva em forma de S no ponto de inflexão e determinando-se as intersecções com o eixo dos tempos e com a reta $c(t)=K$. [Ogata, 2005]

Como é mostrado na figura 33.

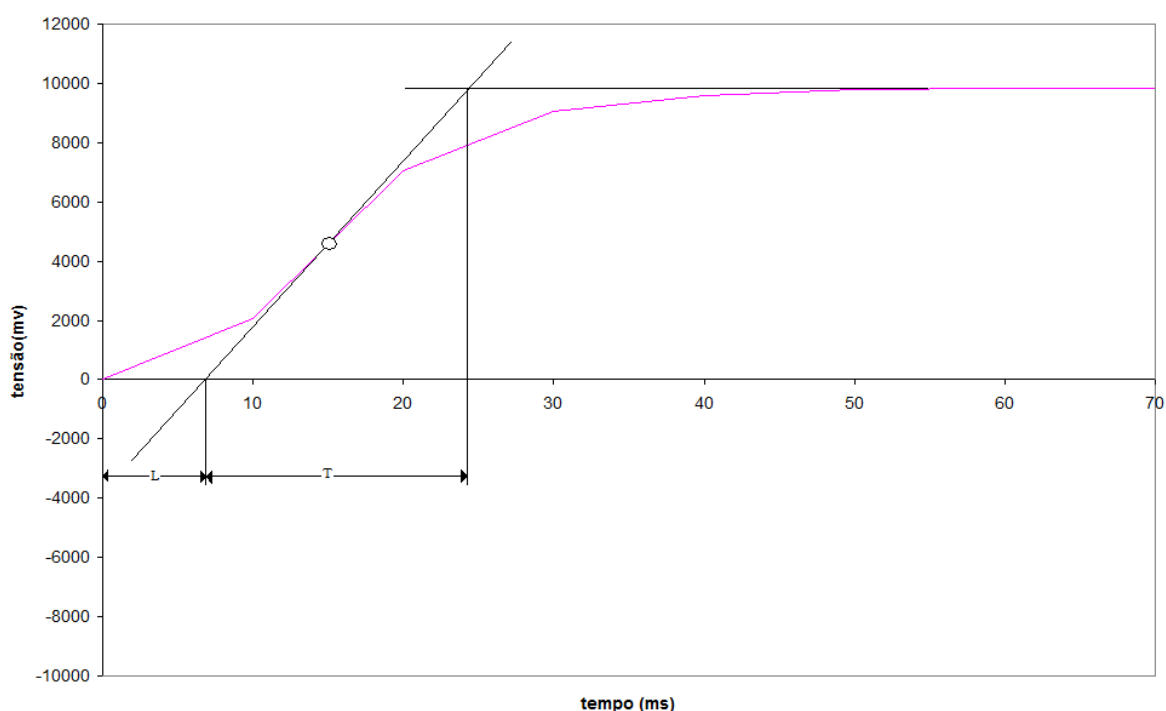


Figura 33 – Obtenção das constantes L e T

Segundo [Ogata,2005], para controlador PI:

$$K_p = 0,9 \frac{T}{L} \quad (\text{Eq. 6})$$

e

$$T_i = \frac{L}{0,3} \quad (\text{Eq. 7})$$

Da equação 6:

$$K_p = 0,9 \times \frac{17,5 \text{ms}}{6,78 \text{ms}} = 2,3$$

E da equação 7, têm-se:

$$T_i = \frac{6,76 \text{ms}}{0,3} = 0,0226$$

$$K_i = \frac{1}{T_i} = \frac{1}{0,0226} = 44,2$$

Como resultados observados do projeto, têm-se a variação de velocidade do motor de 210 a 785 rpm, para uma tensão de alimentação do motor de 12V.

Foi realizado uma averiguação da forma de onda da tensão no resistor de shunt (valor 0,1 ohm) para se ter uma idéia da corrente média no motor. A figura 34 mostra essa forma de onda.

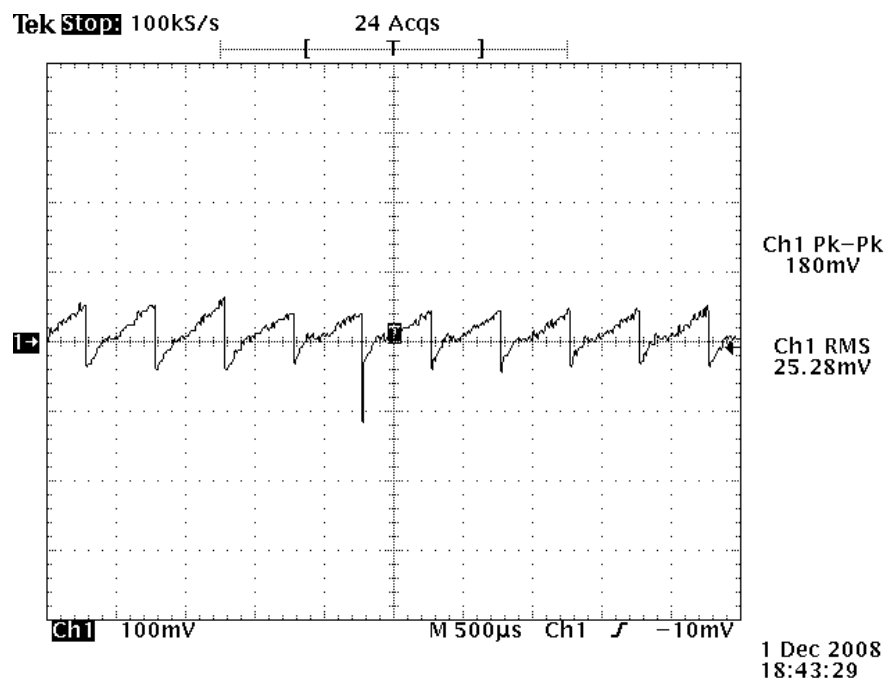


Fig. 34 – Forma de onda da tensão no resistor de shunt



A solução para melhorar o torque é, conhecendo a tensão de operação do motor, implementar um circuito de acionamento específico para esta tensão, com o cuidado de dimensionar os transistores de acordo com os valores de tensão e corrente de operação do motor.

Um motor com seus parâmetros elétricos e mecânicos conhecidos também facilitaria os cálculos dos ganhos do compensador, e aqui pelo não conhecimento desses parâmetros optou-se pela obtenção experimental da curva do motor.

5. CONSIDERAÇÕES FINAIS

5.1. *Problemas Encontrados*

Durante o desenvolvimento de um projeto é considerado normal o aparecimento de problemas, porém é necessário superá-los para que os objetivos sejam atingidos.

O principal problema durante a execução do projeto foi a constatação de defeito no encoder utilizado, que apesar de testes iniciais demonstrarem o funcionamento do equipamento, ao ser colocado em funcionamento contínuo apresentou defeito intermitente, precisando ser substituído. Como não se tinha o mesmo modelo sobressalente, optou-se pelo modelo DGS60, da Sick|Stegmann, de 1024 pulsos por revolução.

Outro problema de relevante magnitude foi a falta de dados sobre o motor. Realizou-se um teste preliminar no gerador trifásico e constatou-se que o motor funcionou com 12V, mas nesse teste não foi levado em consideração o torque. A partir desta constatação optou-se por um circuito de acionamento de 12V. Mas com esse valor de tensão o torque não é alto e aumentando o valor de tensão o torque aumenta.

Outros problemas que foram encontrados são:

- queima de Mosfet's;
- necessidade de colocação de resistores de *pull-down* nos pinos do PWM do microcontrolador,
- queima de um DSPIC;
- falta de literatura em português.



5.2. Conclusões

Para desenvolvimento deste projeto foram necessários conhecimentos em três áreas distintas da engenharia: hardware/eletrônica de potência, programação e conhecimento de máquinas elétricas, mais precisamente do funcionamento do motor síncrono com rotor de ímã permanente.

O objetivo principal deste trabalho, que consistia em desenvolver um controle de velocidade para um motor brushless, foi alcançado.

A partir do estudo do funcionamento do motor brushless e a constatação de que se faz necessário conhecer a posição do rotor a cada instante de tempo, e com a disponibilidade na universidade de um encoder de quadratura, optou-se por este equipamento para controle com sensor.

A escolha do microcontrolador DSPIC30F3010 mostrou-se acertada, pelo baixo custo que este microcontrolador apresenta, pequenas dimensões e recursos disponíveis voltados para o controle de motores.

A variação de velocidade foi conseguida através da variação da tensão na entrada A/D do microcontrolador. Esta variação da tensão fez variar o *duty cycle* do período PWM, fazendo variar a tensão média aplicada nos enrolamentos do motor.

O encoder permitiu saber a posição do rotor a cada instante de tempo, necessário para o chaveamento correto dos MOSFET'S. Também permitiu ler a velocidade angular do motor, através da posição angular do rotor e uma base de tempo fixada pelo *timer* 1 do microcontrolador.

5.3. Sugestões para Trabalhos Futuros



Pode-se utilizar um motor com sensor de efeito hall embutido, o que eliminaria a necessidade de encoder e simplificaria um pouco o projeto, já que a literatura para este tipo de controle com sensor é mais abundante.

Com o encoder utilizado no projeto, pode-se também implementar um controle de posição e/ou determinação de distância através das contagens de pulsos.

Com a utilização de um circuito de acionamento adequado, pode-se utilizar motores de maior capacidade de carga, para utilização em veículos elétricos, por exemplo.



6. REFERÊNCIAS

FITZGERALD, A. E.; KINGSLEY, JR.; UMANS, STEPHEN – **Máquinas Elétricas** – 6a. edição – Editora Bookman – RS – Ano 2006.

SILVA, RENATO A. – **Programando Microcontroladores PIC** – Editora Ensino Profissional, 2006.

OLIVEIRA, ANDRÉ S. DE; ANDRADE, FERNANDO S. DE – **Sistemas Embarcados** – 1a. Ed. – Editora Érica – 2006.

OGATA, KATSUHIKO – **Engenharia de Controle Moderno** – 4ª. Ed. – São Paulo: Editora Prentice Hall do Brasil, 2005.

ELECTRO-CRAFT CORPORATION – **DC Motors Speed Controls Servo Systems** – 4ª. Ed. 1978.

SOUZA, STAN D' - **Sensored BLDC Motor Control Using dsPIC30F2010 (AN957)** – Microchip Technology, 2004.

BROWN, WARD – **Brushless DC Motor Control Made Easy (AN857)** – Microchip Technology, 2002.

LEPKOWSKI, JIM – **Motor Control Sensor Feedback Circuits (AN894)** – Microchip Technology, 2003.

MITRA, SUMIT – **When MCUs and DSPs Collide: Digital Signal Controllers** – Microchip Technology, 2005.

Wikipédia, a enciclopédia livre.



OBRAS CONSULTADAS

NISE, NORMAN S. – **Engenharia de Sistemas de Controle** - 3ª. Ed. – Rio de Janeiro: Editora LTC – Livros Técnicos e Científicos Editora S.A., 2002.

KLAFTER, RICHARD D.; CHMIELEWSKI, THOMAS A.; NEGIN, MICHAEL – **Robotic Engineering, An Integrated Approach** – Editora Prentice Hall, 1989.

COCIAN, LUIS F. E. – **Manual da Linguagem C** – 1ª. Ed. – Editora da Ulbra, 2004.

Cerne Tecnologia e Treinamento - **Apostila de C para dsPIC.**

Apostila “Curso de Liguagem C UFMG”.

MICROCHIP – **dspic30f3010 Datasheet.**

MICROCHIP – **dspic30f Programmer’s Reference Manual.**

YEDAMALE, PADMARAJA – **Brushless DC Motor Fundamentals (AN885)** – Microchip Technology, 2003.

ZAMBADA, JORGE – **Measuring Speed and Position with the QEI Module** – Microchip Technology, 2005.

ZAMBADA, JORGE – **Driving an ACIM with the dsPIC DSC MCPWM Module** – Microchip Technology, 2005.

SOUZA, STAN D’ – **Getting Started with BLDC Motors and dsPIC30F Devices** – Microchip Technology, 2005.



APÊNDICE A – CÓDIGO FONTE

**Encoder.c**

```
#include <p30f3010.h>
```

```
#define FCY 7372800
```

```
#define MAX_CNT_PER_REV (1024 * 2 - 1)
```

```
#define MAXSPEED (unsigned int)((((unsigned long)MAX_CNT_PER_REV*2000)/125)
```

```
#define HALFMXSPEED (MAXSPEED>>1)
```

```
#define Ksp 2
```

```
#define Ksi 44
```

```
#define RPMConstant 60*(FCY/256)
```

```
void CalculateDC(void);
```

```
void GetSpeed(void);
```

```
int ActualSpeed;
```

```
int SpeedError;
```

```
int DutyCycle;
```

```
int SpeedIntegral;
```

```
struct {
```

```
unsigned RunMotor : 1;
```

```
unsigned Minus : 1;
```

```
unsigned unused : 14;
```

```
} Flags;
```

```
int Speed;
```

```
int DesiredSpeed;
```

```
unsigned int Timer1;
```

```
unsigned char Count;
```

```
unsigned char SpeedCount;
```

```
int AngPos[2] = {0,0}; // Two variables are used for Speed Calculation
```

```
int POSCNTcopy = 0;
```

```
char nume=0x30;
```

```
////////////////////////////////////
```

```
void InitQEI(void)
```

```
{
```

```
ADPCFG |= 0xFFFF8; // Configure QEI pins as digital inputs
```

```
QEICONbits.QEIM = 0; // Disable QEI Module
```

```
QEICONbits.CNTERR = 0; // Clear any count errors
```



```
QEICONbits.QEISIDL = 0; // Continue operation during sleep
QEICONbits.SWPAB = 0; // QEA and QEB not swapped
QEICONbits.PCDOUT = 0; // Normal I/O pin operation
QEICONbits.POSRES = 1; // Index pulse resets position counter
DFLTCONbits.CEID = 1; // Count error interrupts disabled
DFLTCONbits.QEOUT = 1; // Digital filters output enabled for QEn pins
DFLTCONbits.QECK = 2;
POSCNT = 0; // Reset position counter
QEICONbits.QEIM = 4; // X4 mode with position counter reset by Index
return;
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
void InitTMR1(void)
{
TMR1 = 0; // Reset timer counter
T1CONbits.TON = 0; // Turn off timer 1
T1CONbits.TSIDL = 0; // Continue operation during sleep
T1CONbits.TGATE = 0; // Gated timer accumulation disabled
T1CONbits.TCS = 0; // use Tcy as source clock
T1CONbits.TCKPS = 1;
PR1 = 5750; // Interrupt period = 0.1 sec with a 64 prescaler
IFS0bits.T1IF = 0; // Clear timer 1 interrupt flag
IEC0bits.T1IE = 1; // Enable timer 1 interrupts
T1CONbits.TON = 1; // Turn on timer 1
return;
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
void PositionCalculation(void)
{
POSCNTcopy = (int)POSCNT;
if (POSCNTcopy < 0)
POSCNTcopy = -POSCNTcopy;
AngPos[1] = AngPos[0];
AngPos[0] = (unsigned int)((((unsigned long)POSCNTcopy * 2000)/125);
// 0 <= POSCNT <= 2047 to 0 <= AngPos <= 32752
return;
}
```

```
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```

```
void __attribute__((__interrupt__)) _T1Interrupt (void)
{
IFS0bits.T1IF = 0; // Clear timer 1 interrupt flag
PositionCalculation();
Speed = AngPos[0] - AngPos[1];
}
```



```
if (Speed >= 0)
{
    if(Speed >= (HALFMAXSPEED))
    Speed = Speed - MAXSPEED;
}
else
{
    if (Speed < -(HALFMAXSPEED))
    Speed = Speed + MAXSPEED;
}
Speed *= 2;
return;
}

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

void _ISR_ADCInterrupt(void)
{
    IFS0bits.ADIF = 0;
    DesiredSpeed = 900+(ADCBUF0/2);
}

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

void InitADC10(void)
{
    ADPCFG = 0xFFF8; // all PORTB = Digital;RB0 to RB2 = analog
    ADCON1 = 0x0064; // PWM starts conversion
    ADCON2 = 0x0000; // sample CH0 channel
    ADCHS = 0x0002; // Connect RB2/AN2 as CH0 = pot.
    ADCON3 = 0x0080; // Tad = internal RC (4uS)
    IFS0bits.ADIF = 0; // clear flag
    IEC0bits.ADIE = 1; // enable interrupt
    ADCON1bits.ADON = 1; // turn ADC ON
}

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

void CalculateDC(void)
{
    DesiredSpeed = DesiredSpeed*3;
    Flags.Minus = 0;
    if (ActualSpeed > DesiredSpeed)
    SpeedError = ActualSpeed - DesiredSpeed;
    else
    {
        SpeedError = DesiredSpeed - ActualSpeed;
        Flags.Minus = 1;
    }
    SpeedIntegral += SpeedError;
}
```



```
if (SpeedIntegral > 9000)
SpeedIntegral = 0;
DutyCycle = (((long)Ksp*(long)SpeedError + (long)Ksi*(long)SpeedIntegral) >> 12);
DesiredSpeed = DesiredSpeed/3;
if (Flags.Minus)
DutyCycle = DesiredSpeed + DutyCycle;
else DutyCycle = DesiredSpeed - DutyCycle;
if (DutyCycle < 1000)
DutyCycle = 1000;
if (DutyCycle > 5000)
{DutyCycle = 5000;SpeedIntegral = 0;}
```

```
PDC1 = DutyCycle;
PDC2 = PDC1;
PDC3 = PDC1;}
```

```
////////////////////////////////////
```

```
char dezmil(int num)
{
int dezmil;
dezmil=num/10000; // 1
return (dezmil+48); //retorna o 1
}
```

```
char milhar(int num)
{
int dezmil;
int milhar;
dezmil=num/10000; // 1
milhar=num-dezmil*10000; // 12345-10000 = 2345
milhar=milhar/1000; // 2
return (milhar+48); // retorna 2
}
```

```
char cent(int num)
{
int dezmil;
int milhar;
int cent;
dezmil=num/10000; // 1
milhar=num-dezmil*10000; // 2345
milhar=milhar/1000; // 2
cent=num-dezmil*10000-milhar*1000; // 12345-10000-2000=345
cent=cent/100;
return (cent+48); // retorna 3
}
```

```
char dez(int num)
{
```



```
int dezmil;
int milhar;
int cent;
int dez;
dezmil=num/10000; // 1
milhar=num-dezmil*10000; // 2345
milhar=milhar/1000; // 2
cent=num-dezmil*10000-milhar*1000; // 12345-10000-2000=345
cent=cent/100; //3
dez=num-dezmil*10000-milhar*1000-cent*100; // 12345-10000-2000-300=45
dez=dez/10; // 4
return (dez+48); // retorna 4
}
```

```
char uni(int num)
{
int dezmil;
int milhar;
int cent;
int dez;
int uni;
dezmil=num/10000; // 1
milhar=num-dezmil*10000; // 2345
milhar=milhar/1000; // 2
cent=num-dezmil*10000-milhar*1000; // 12345-10000-2000=345
cent=cent/100;
dez=num-dezmil*10000-milhar*1000-cent*100; // 12345-10000-2000-300=45
dez=dez/10; // 4
uni=num-dezmil*10000-milhar*1000-cent*100-dez*10; // 12345-10000-2000-300-40=5
return (uni+48);
}
```

```
////////////////////////////////////
```

```
void GetSpeed(void)
{
if (Timer1 > 23000) // if TMR1 is large ignore reading
return;
if (Timer1 > 0)
Speed = RPMConstant/(long)Timer1;// get speed in RPM
ActualSpeed += Speed;
ActualSpeed = ActualSpeed >> 1;
if (++SpeedCount == 1)
{SpeedCount = 0;CalculateDC();}
}
```

```
////////////////////////////////////
```



```
int main (void)
{

InitTMR1();
InitQEI();
InitADC10();

TRISBbits.TRISB1=0;
TRISBbits.TRISB2=0;

/* Set up I/O Port */
ADPCFG=0xFFFF8;          /* Analog ports as Digital I/O */
TRISB=0x0000;           /* Port B output Data as well as serial port */

/* Main Program Loop */
while(1)
{

PTPER = 0x0397;

{

if (++Count == 2) // do this for 2 electrical revolutions or 1
                // mechanical revolution for a 4 pole motor
{
Timer1 = TMR1; // read latest tmr1 value
TMR1 = 0;
Count = 0;
GetSpeed(); // determine speed
}

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

PWMCON1 = 0x0700; // zera o registrador para reinício

while ((POSCNT <= 170.58) || ((POSCNT > 1023.5) && (POSCNT <= 1194.08)))
// SETOR 1
{
PTCON = 0x8000; // Enabling the PWM module
PWMCON1 = 0x0714; // seta o pwm1 no modo complementar
}

while (((POSCNT > 170.58) && (POSCNT <= 341.16)) || ((POSCNT > 1194.08) &&
(POSCNT <= 1364.66))) // SETOR 2
{
```



```
PTCON = 0x8000;          // Enabling the PWM module
PWMCON1 = 0x0712;        // seta o pwm2 no modo complementar
}

while (((POSCNT > 341.16) && (POSCNT <= 511.75)) || ((POSCNT > 1364.66) &&
(POSCNT <= 1535.25))) // SETOR 3
{
PTCON = 0x8000;          // Enabling the PWM module
PWMCON1 = 0x0742;        //seta o pwm3 no modo complementar
}

while (((POSCNT > 511.75) && (POSCNT <= 682.33)) || ((POSCNT > 1535.25) &&
(POSCNT <= 1705.83))) // SETOR 4
{
PTCON = 0x8000;          // Enabling the PWM module
PWMCON1 = 0x0741;        // seta o pwm1 no modo complementar
}

while (((POSCNT > 682.33) && (POSCNT <= 852.91)) || ((POSCNT > 1705.83) &&
(POSCNT <= 1876.41))) // SETOR 5
{
PTCON = 0x8000;          // Enabling the PWM module
PWMCON1 = 0x0721;        // seta o pwm2 no modo complementar
}

while (((POSCNT > 852.91) && (POSCNT <= 1023.5)) || ((POSCNT > 1876.41) &&
(POSCNT <= 2047))) // SETOR 6
{
PTCON = 0x8000;          // Enabling the PWM module
PWMCON1 = 0x0724;        //seta o pwm3 no modo complementar
}

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

}

}

return(0);

}
```



ANEXO A – DATASHEET DSPIC30F3010



dsPIC30F3010/3011

dsPIC30F3010/3011 Enhanced Flash 16-bit Digital Signal Controller

Note: This data sheet summarizes features of this group of dsPIC30F devices and is not intended to be a complete reference source. For more information on the CPU, peripherals, register descriptions and general device functionality, refer to the dsPIC30F Family Reference Manual (DS70046). For more information on the device instruction set and programming, refer to the dsPIC30F Programmer's Reference Manual (DS70090).

High Performance Modified RISC CPU:

- Modified Harvard architecture
- C compiler optimized instruction set architecture with flexible addressing modes
- 84 base instructions
- 24-bit wide instructions, 16-bit wide data path
- 24 Kbytes on-chip Flash program space (8K instruction words)
- 1 Kbytes of on-chip data RAM
- 1 Kbytes of non-volatile data EEPROM
- 16 x 16-bit working register array
- Up to 30 MIPS operation:
 - DC to 40 MHz external clock input
 - 4 MHz-10 MHz oscillator input with PLL active (4x, 8x, 16x)
- 29 interrupt sources
 - 3 external interrupt sources
 - 8 user selectable priority levels for each interrupt source
 - 4 processor trap sources

DSP Engine Features:

- Dual data fetch
- Accumulator write back for DSP operations
- Modulo and Bit-Reversed Addressing modes
- Two, 40-bit wide accumulators with optional saturation logic
- 17-bit x 17-bit single cycle hardware fractional/integer multiplier
- All DSP instructions single cycle
- \pm 16-bit single cycle shift

Peripheral Features:

- High current sink/source I/O pins: 25 mA/25 mA
- Timer module with programmable prescaler:
 - Five 16-bit timers/counters; optionally pair 16-bit timers into 32-bit timer modules
- 16-bit Capture Input functions
- 16-bit Compare/PWM output functions
- 3-wire SPI™ modules (supports 4 Frame modes)
- I²C™ module supports Multi-Master/Slave mode and 7-bit/10-bit addressing
- 2 UART modules with FIFO Buffers

Motor Control PWM Module Features:

- 6 PWM output channels
 - Complementary or Independent Output modes
 - Edge and Center-Aligned modes
- 3 duty cycle generators
- Dedicated time base
- Programmable output polarity
- Dead-time control for Complementary mode
- Manual output control
- Trigger for A/D conversions

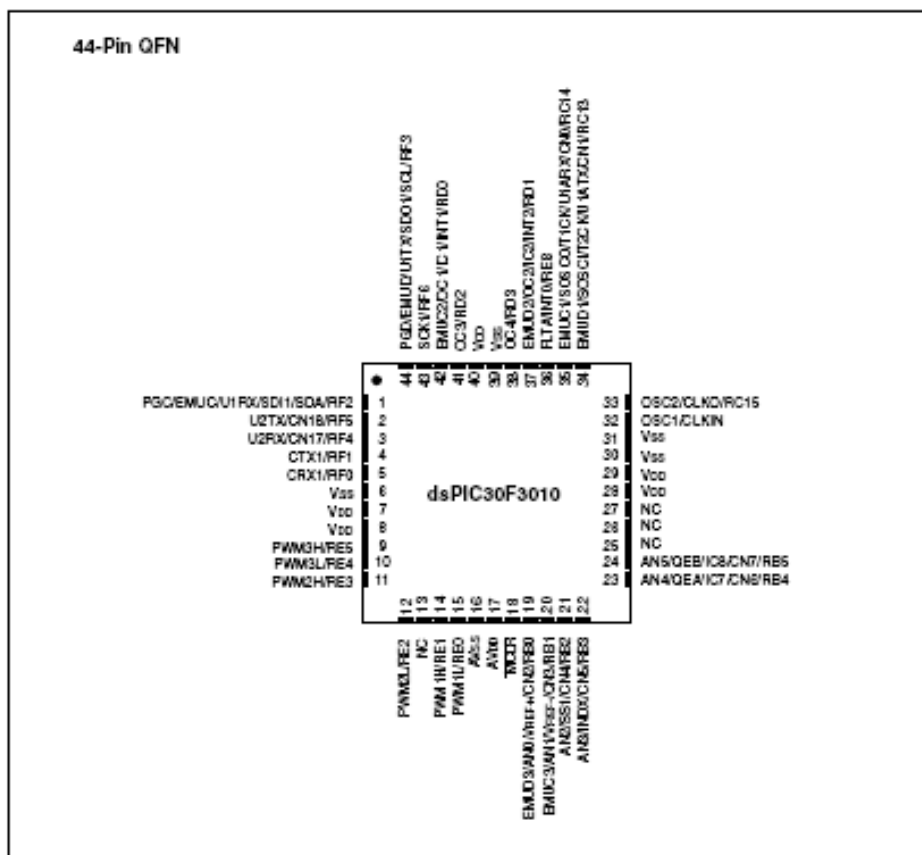
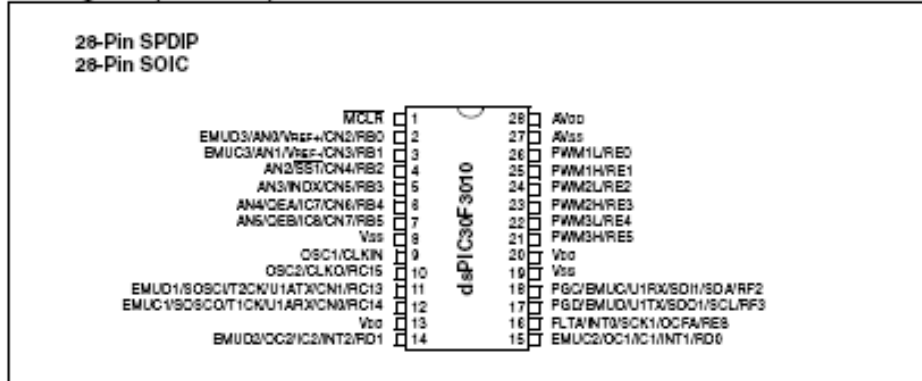
Quadrature Encoder Interface Module Features:

- Phase A, Phase B and Index Pulse Input
- 16-bit up/down position counter
- Count direction status
- Position Measurement (x2 and x4) mode
- Programmable digital noise filters on inputs
- Alternate 16-bit Timer/Counter mode
- Interrupt on position counter rollover/underflow



dsPIC30F3010/3011

Pin Diagrams (Continued)





ANEXO B – DATASHEET DO ENCODER



DGS60, DGS65 and DGS66: Incremental Encoders for rough environmental conditions

DATA SHEET



Number of lines
100 to 10,000
Incremental Encoder

Incremental encoders in the DGS60, DGS65 and DGS66 series are in use world-wide under the toughest environmental conditions.

The rugged construction – up to IP 67 degree protection – and the individual adaptation of the design to the requirements of the user are the outstanding features of this series.

Resolutions up to 10,000 lines are available.

Select your individual encoder!
Possible product variants:
6 and 10 mm solid shafts with servo flange or face mount flange, through or blind hollow shafts with connector or cable outlet, TTL or HTL interface.

Thanks to this wide variety of products, there are numerous possible uses, for example in:

- machine tools
- textile machines
- woodworking machines
- packaging machines

SICK | STEGMANN

Incremental Encoder DGS60, face mount and servo flange

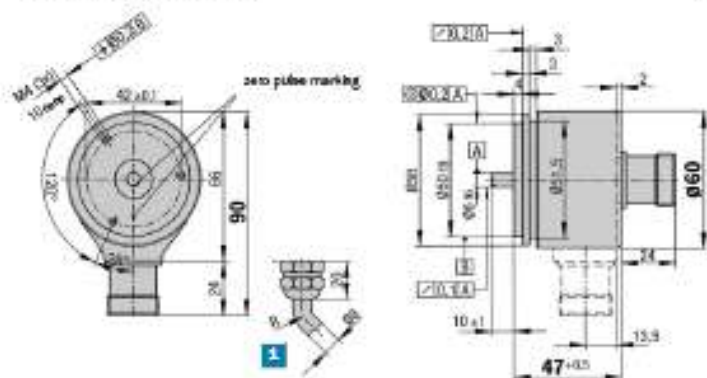
Number of lines
100 to 10,000
Incremental Encoder

- Servo or face mount flange
- Connector or cable outlet
- Protection class up to IP 67
- Electrical Interfaces
TTL and HTL



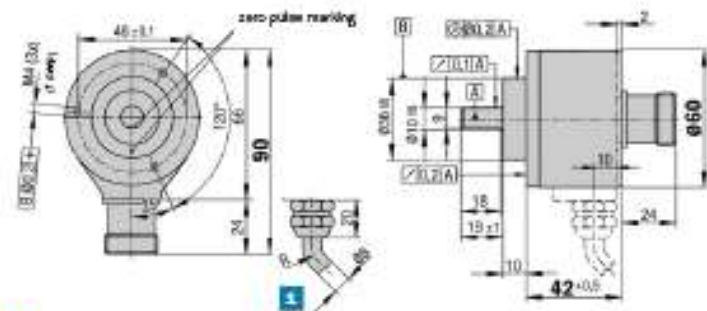
Accessories
Connection systems
Mounting systems

Dimensional drawing servo flange



R = bending radius min. 40 mm
General tolerances according to DIN ISO 2768-mk

Dimensional drawing face mount flange



R = bending radius min. 40 mm
General tolerances according to DIN ISO 2768-mk

PIN and wire allocation/cable 11 core

PIN	Signal HTL	Signal TTL	Core colour (cable outlet)	Explanation
1	N. C.	B'	black	Signal line
2	N. C.	Sense +	gray	Connected internally to U ₂
3	Z	Z	blue	Signal line
4	N. C.	Z'	yellow	Signal line
5	A	A	white	Signal line
6	N. C.	A'	brown	Signal line
7	N. C.	N. C.	orange	N. C.
8	B	B	pink	Signal line
9	Screen	Screen		Housing potential
10	GND	GND	blue	Ground connection
11	N. C.	Sense -	green	Connected internally to ground
12	U ₂	U ₂	red	Power supply ¹⁾



¹⁾ Potential free to housing
N. C. = Not Connected



ANEXO C – DATASHEET DO MOSFET IRF640




IRF640, RF1S640, RF1S640SM
Data Sheet
January 2002
18A, 200V, 0.180 Ohm, N-Channel Power MOSFETs

These are N-Channel enhancement mode silicon gate power field effect transistors. They are advanced power MOSFETs designed, tested, and guaranteed to withstand a specified level of energy in the breakdown avalanche mode of operation. All of these power MOSFETs are designed for applications such as switching regulators, switching converters, motor drivers, relay drivers, and drivers for high power bipolar switching transistors requiring high speed and low gate drive power. These types can be operated directly from integrated circuits.

Formerly developmental type TA17422.

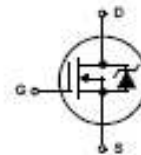
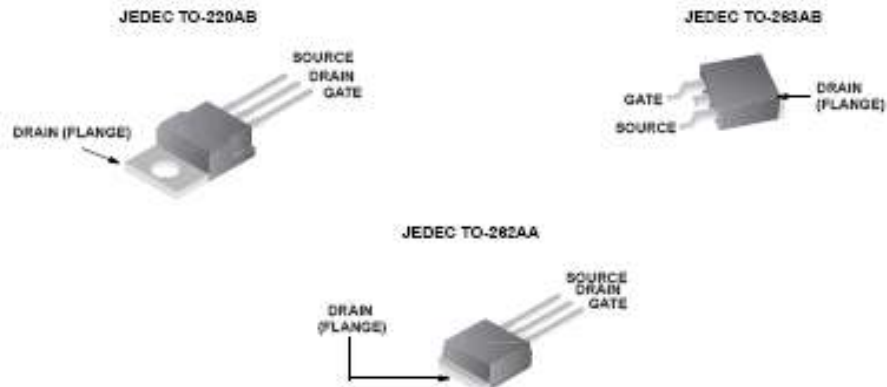
Ordering Information

PART NUMBER	PACKAGE	BRAND
IRF640	TO-220AB	IRF640
RF1S640	TO-262AA	RF1S640
RF1S640SM	TO-263AB	RF1S640

NOTE: When ordering, use the entire part number. Add the suffix 3A to obtain the TO-263AB variant in the tape and reel, i.e., RF1S640SM3A.

Features

- 18A, 200V
- $r_{DS(ON)} = 0.180\Omega$
- Single Pulse Avalanche Energy Rated
- SOA is Power Dissipation Limited
- Nanosecond Switching Speed
- Linear Transfer Characteristics
- High Input Impedance
- Related Literature
 - TB334 "Guidelines for Soldering Surface Mount Components to PC Boards"

Symbol

Packaging


IRF640, RF1S640, RF1S640SM
Absolute Maximum Ratings $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

	IRF640, RF1S640, RF1S640SM	UNITS	
Drain to Source Breakdown Voltage (Note 1)	V_{DS}	200	V
Drain to Gate Voltage ($R_{GS} = 20k\Omega$) (Note 1)	V_{DGR}	200	V
Continuous Drain Current	I_D	18	A
$T_C = 100^\circ\text{C}$	I_D	11	A
Pulsed Drain Current (Note 3)	I_{DM}	72	A
Gate to Source Voltage	V_{GS}	± 20	V
Maximum Power Dissipation	P_D	125	W
Dissipation Derating Factor		1.0	$\text{W}/^\circ\text{C}$
Single Pulse Avalanche Energy Rating (Note 4)	E_{AS}	580	mJ
Operating and Storage Temperature	T_J, T_{STG}	-55 to 150	$^\circ\text{C}$
Maximum Temperature for Soldering			
Leads at 0.063in (1.6mm) from Case for 10s	T_L	300	$^\circ\text{C}$
Package Body for 10s, See TB334	T_{pkg}	250	$^\circ\text{C}$

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

- $T_J = 25^\circ\text{C}$ to 125°C .

Electrical Specifications $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS	
Drain to Source Breakdown Voltage	BV_{DSS}	$I_D = 250\mu\text{A}, V_{GS} = 0\text{V}$ (Figure 10)	200	-	-	V	
Gate Threshold Voltage	$V_{GS(TH)}$	$V_{GS} = V_{DS}, I_D = 250\mu\text{A}$	2	-	4	V	
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS} = \text{Rated } BV_{DSS}, V_{GS} = 0\text{V}$	-	-	25	μA	
		$V_{DS} = 0.8 \times \text{Rated } BV_{DSS}, V_{GS} = 0\text{V}, T_J = 125^\circ\text{C}$	-	-	250	μA	
On-State Drain Current (Note 1)	$I_{D(ON)}$	$V_{DS} > I_{D(ON)} \times r_{DS(ON)MAX}, V_{GS} = 10\text{V}$ (Figure 7)	18	-	-	A	
Gate to Source Leakage Current	I_{GSS}	$V_{GS} = \pm 20\text{V}$	-	-	± 100	nA	
Drain to Source On Resistance (Note 1)	$r_{DS(ON)}$	$I_D = 10\text{A}, V_{GS} = 10\text{V}$ (Figures 8, 9)	-	0.14	0.18	Ω	
Forward Transconductance (Note 1)	g_{fs}	$V_{DS} \geq 10\text{V}, I_D = 11\text{A}$ (Figure 12)	6.7	10	-	S	
Turn-On Delay Time	$t_{d(ON)}$	$V_{DD} = 100\text{V}, I_D = 18\text{A}, R_{GS} = 9.1\Omega, R_L = 5.4\Omega$, MOSFET Switching Times are Essentially Independent of Operating Temperature	-	13	21	ns	
Rise Time	t_r		-	50	77	ns	
Turn-Off Delay Time	$t_{d(OFF)}$		-	46	68	ns	
Fall Time	t_f		-	35	54	ns	
Total Gate Charge (Gate to Source + Gate to Drain)	$Q_g(\text{TOT})$	$V_{GS} = 10\text{V}, I_D = 18\text{A}, V_{DS} = 0.8 \times \text{Rated } BV_{DSS}$ (Figure 14) Gate Charge is Essentially Independent of Operating Temperature	-	43	64	nC	
Gate to Source Charge	Q_{gs}	$I_{G(REF)} = 1.5\text{mA}$	-	8	-	nC	
Gate to Drain "Miller" Charge	Q_{gd}		-	22	-	nC	
Input Capacitance	C_{iss}	$V_{DS} = 25\text{V}, V_{GS} = 0\text{V}, f = 1\text{MHz}$ (Figure 11)	-	1275	-	pF	
Output Capacitance	C_{oss}		-	400	-	pF	
Reverse Transfer Capacitance	C_{rss}		-	100	-	pF	
Internal Drain Inductance	L_D	Measured From the Contact Screw on Tab to Center of Die	Modified MOSFET Symbol Showing the Internal Devices Inductances	-	3.5	-	nH
		Measured From the Drain Lead, 6mm (0.25in) From Package to Center of Die		-	4.5	-	nH
Internal Source Inductance	L_S	Measured From the Source Lead, 6mm (0.25in) from Header to Source Bonding Pad	-	7.5	-	nH	
Thermal Resistance Junction to Case	$R_{\theta JC}$		-	-	1	$^\circ\text{C}/\text{W}$	
Thermal Resistance Junction to Ambient	$R_{\theta JA}$	Free Air Operation, IRF640	-	-	62	$^\circ\text{C}/\text{W}$	
	$R_{\theta JA}$	RF1S640SM Mounted on FR-4 Board with Minimum Mounting Pad	-	-	62	$^\circ\text{C}/\text{W}$	



ANEXO D – DATASHEET DO MOSFET IRF9640



IRF9640, RF1S9640SM
Data Sheet
July 1999
File Number 2284.2
11A, 200V, 0.500 Ohm, P-Channel Power MOSFETs

These are P-Channel enhancement mode silicon-gate power field-effect transistors. They are advanced power MOSFETs designed, tested, and guaranteed to withstand a specified level of energy in the breakdown avalanche mode of operation. All of these power MOSFETs are designed for applications such as switching regulators, switching converters, motor drivers, relay drivers and as drivers for other high-power switching devices. The high input impedance allows these types to be operated directly from integrated circuits.

Formerly developmental type TA17522.

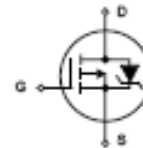
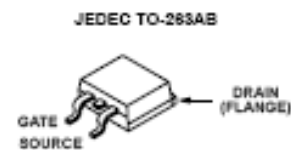
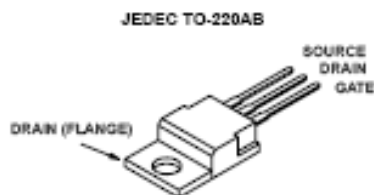
Ordering Information

PART NUMBER	PACKAGE	BRAND
IRF9640	TO-220AB	IRF9640
RF1S9640SM	TO-263AB	RF1S9640

NOTE: When ordering, use the entire part number. Add the suffix 9A to obtain the TO-263AB variant in the tape and reel, i.e., RF1S9640SM9A.

Features

- 11A, 200V
- $r_{DS(ON)} = 0.500\Omega$
- Single Pulse Avalanche Energy Rated
- SOA Is Power Dissipation Limited
- Nanosecond Switching Speeds
- Linear Transfer Characteristics
- High Input Impedance
- Related Literature
 - TB334, "Guidelines for Soldering Surface Mount Components to PC Boards"

Symbol

Packaging


IRF9640, RF1S9640SM
Absolute Maximum Ratings $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

	IRF9640, RF1S9640SM	UNITS
Drain to Source Breakdown Voltage (Note 1)	-200	V
Drain to Gate Voltage ($R_{GS} = 20k\Omega$) (Note 1)	-200	V
Continuous Drain Current	-11	A
$T_C = 100^\circ\text{C}$	-7	A
Pulsed Drain Current (Note 2)	-44	A
Gate to Source Voltage	± 20	V
Maximum Power Dissipation	125	W
Linear Derating Factor	1	W/ $^\circ\text{C}$
Single Pulse Avalanche Energy Rating (Note 3, 4)	750	mJ
Operating and Storage Temperature	-55 to 150	$^\circ\text{C}$
Maximum Temperature for Soldering		
Leads at 0.063in (1.6mm) from Case for 10s	300	$^\circ\text{C}$
Package Body for 10s, See Techbrief 334	260	$^\circ\text{C}$

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. $T_J = 25^\circ\text{C}$ to 125°C

Electrical Specifications $T_C = 25^\circ\text{C}$, Unless Otherwise Specified

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Drain to Source Breakdown Voltage	BV_{DSS}	$I_D = -250\mu\text{A}$, $V_{GS} = 0\text{V}$ (Figure 10)	-200	-	-	V
Gate Threshold Voltage	$V_{GS(TH)}$	$V_{GS} = V_{DS}$, $I_D = -250\mu\text{A}$	-2	-	-4	V
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS} = \text{Rated } BV_{DSS}$, $V_{GS} = 0\text{V}$	-	-	25	μA
		$V_{DS} = 0.8 \times \text{Rated } BV_{DSS}$, $V_{GS} = 0\text{V}$, $T_C = 125^\circ\text{C}$	-	-	250	μA
On-State Drain Current (Note 2)	$I_{D(ON)}$	$V_{DS} > I_{D(ON)} \times r_{DS(ON)MAX}$, $V_{GS} = -10\text{V}$	-11	-	-	A
Gate to Source Leakage Current	I_{GSS}	$V_{GS} = \pm 20\text{V}$	-	-	± 100	nA
Drain to Source On Resistance (Note 2)	$r_{DS(ON)}$	$I_D = -6\text{A}$, $V_{GS} = -10\text{V}$ (Figures 8, 9)	-	0.350	0.500	Ω
Forward Transconductance (Note 2)	g_{fs}	$V_{DS} > I_{D(ON)} \times r_{DS(ON)MAX}$, $I_D = -6\text{A}$ (Figure 12)	4	6	-	S
Turn-On Delay Time	$t_{d(ON)}$	$V_{DS} = 0.5 \times \text{Rated } BV_{DSS}$, $I_D = -11\text{A}$, $R_G = 9.1\Omega$	-	18	22	ns
Rise Time	t_r	$V_{GS} = -10\text{V}$ (Figures 17, 18)	-	45	68	ns
Turn-Off Delay Time	$t_{d(OFF)}$	$R_L = 8.4\Omega$ for $V_{DSS} = -100\text{V}$	-	75	90	ns
Fall Time	t_f	$R_L = 5.1\Omega$ for $V_{DSS} = -75\text{V}$	-	29	44	ns
Total Gate Charge (Gate to Source + Gate to Drain)	$Q_g(\text{TOT})$	$V_{GS} = -10\text{V}$, $I_D = -11\text{A}$, $V_{DS} = 0.8 \times \text{Rated } BV_{DSS}$ $I_{G(\text{REF})} = -1.5\text{mA}$ (Figures 14, 15, 20)	-	70	90	nC
			Gate Charge is Essentially Independent of Operating Temperature	-	55	-
Gate to Source Charge	Q_{gs}	MOSFET Switching Times are Essentially Independent of Operating Temperature	-	15	-	nC
Gate to Drain "Miller" Charge	Q_{gd}		-	1100	-	pF
Input Capacitance	C_{iss}		$V_{DS} = -25\text{V}$, $V_{GS} = 0\text{V}$, $f = 1\text{MHz}$	-	375	-
Output Capacitance	C_{oss}	(Figure 11)	-	150	-	pF
Reverse Transfer Capacitance	C_{rss}		-	-	-	pF
Internal Drain Inductance	L_D	Measured From the Contact Screw on Tab To Center of Die	-	3.5	-	nH
		Measured From the Drain Lead, 6mm (0.25in) from Package to Center of Die	-	4.5	-	nH
Internal Source Inductance	L_S	Measured From the Source Lead, 6mm (0.25in) from Header to Source Bonding Pad	-	7.5	-	nH
Thermal Resistance Junction to Case	$R_{\theta JC}$		-	-	1.0	$^\circ\text{C/W}$
Thermal Resistance Junction to Ambient	$R_{\theta JA}$	Typical Socket Mount	-	-	62.5	$^\circ\text{C/W}$

