

**UNIVERSIDADE DE CAXIAS DO SUL
ÁREA DO CONHECIMENTO DE CIÊNCIAS EXATAS E ENGENHARIA**

VINÍCIUS SILVA VELHO

**DRIVE SENSORLESS PARA MOTORES DE CORRENTE CONTÍNUA SEM
ESCOVAS DE BAIXO CUSTO PARA COMPRESSORES DE GELADEIRA
AUTOMOTIVA**

**Caxias do Sul
2023**

Vinícius Silva Velho

**DRIVE SENSORLESS PARA MOTORES DE CORRENTE CONTÍNUA SEM
ESCOVAS DE BAIXO CUSTO PARA COMPRESSORES DE GELADEIRA
AUTOMOTIVA**

Trabalho de Conclusão de Curso apresentado
como parte do requisito para obtenção do título
de Bacharel em Engenharia Elétrica à
Universidade de Caxias do Sul.

Orientador:
Prof. Me. Bruno Fensterseifer Dias

**Caxias do Sul
2023**

Ficha catalográfica.

SILVA VELHO, Vinícius

Drive Sensorless para motores de corrente contínua sem escovas de baixo custo para compressores de geladeira automotiva / Vinícius Silva Velho; orientador: Bruno Fensterseifer Dias, Caxias do Sul: UCS, Área do Conhecimento de Ciências Exatas e Engenharias, 2023.

95 folhas, 30 cm.

Trabalho de Conclusão de Curso – Universidade de Caxias do Sul, 2023

Inclui referências bibliográficas.

1. Motor CC. 2. Sem escovas. 3. Sem sensor. 4. Drive. 5. BLDC. I. Costa, Guilherme H. II. Universidade de Caxias do Sul. III. Drive Sensorless para motores de corrente contínua sem escovas de baixo custo para compressores de geladeira automotiva.

Vinícius Silva Velho

**DRIVE SENSORLESS PARA MOTORES DE CORRENTE CONTÍNUA SEM
ESCOVAS DE BAIXO CUSTO PARA COMPRESSORES DE GELADEIRA
AUTOMOTIVA**

Trabalho de Conclusão de Curso apresentado
como parte do requisito para obtenção do título
de Bacharel em Engenharia Elétrica à
Universidade de Caxias do Sul.

Orientador:
Prof. Me. Bruno Fensterseifer Dias

Aprovado em 28/11/2023

Banca Examinadora

Prof. Me. Bruno Fensterseifer Dias (orientador)
Universidade de Caxias do Sul - UCS

Prof. Me. Ricardo Leal Costi
Universidade de Caxias do Sul - UCS

Prof. Emerson Samuel de Oliveira
Universidade/Faculdade Técnica - UNIFTEC

AGRADECIMENTOS

Primeiramente a meu pai por ter me propiciado a oportunidade de estudar que ele nunca teve.
À meu irmão Sidnei por sempre ter me incentivado em minhas decisões.

Ao professor Bruno Fensterseifer Dias, pela confiança e orientação no desenvolvimento deste projeto, contribuindo com seu conhecimento e dedicação.

A todos os amigos que sempre entenderam os longos momentos de ausência durante todo o período de graduação, especialmente ao amigo G. Bresolin.

“[...] Saiba bem o que te apetece, e não recue ante nenhum pretexto, porque o universo se organizará para te dissuadir.”
Friedrich Nietzsche

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema de acionamento sem sensoramento de baixo custo para motores CC sem escova aplicáveis em compressores de refrigeradores automotivos. Para seu correto funcionamento, o controlador proposto conta com tanto um estimador discreto de posição quanto um estimador de velocidades. Os estimadores são baseados na leitura de corrente do barramento CC e na leitura da FEM (Força Eletromotriz) gerada pela interação dos ímãs do rotor com o campo magnético gerado pelas bobinas do estator. O controle de velocidade apresenta dois estágios: um baseado no acionamento em open-loop e um completamente realimentado, baseado na leitura da FEM. É apresentado um método de determinação da posição inicial utilizando a leitura de corrente das bobinas, através de doze acionamentos distintos das bobinas. O trabalho apresenta o desenvolvimento do hardware proposto, trazendo aspectos de reusabilidade e otimização de custos. Além do desenvolvimento do hardware para o acionamento do motor, o presente projeto engloba também o desenvolvimento do firmware para o microcontrolador utilizado. Por fim, são apresentados os resultados obtidos com os testes práticos do sistema.

Palavras-chave: Motor CC. Sem escovas. Sem sensor. Drive. BLDC.

ABSTRACT

This work presents the development of a low-cost sensorless drive system for Brushless DC motors used in automotive refrigerator compressors. For its correct operation, the controller features both a discrete position estimator and a velocity estimator, based on the DC bus current reading and the EMF reading generated by the interaction of the rotor magnets with the magnetic field generated by the stator coils. The speed control has two stages: one based on open-loop drive and one fully feedback based on the EMF reading. A method for determining the initial position is presented using the current reading of the coils, through twelve different activations of the coils. The work presents the development of the proposed hardware, bringing aspects of reusability and cost optimization. In addition to the development of the hardware to drive the motor, this project also includes the development of the firmware for the microcontroller used, applying good programming practices. In addition, the results obtained with the practical tests of the system are presented at the end.

Keywords: Motor. Brushless. Sensorless. Drive. BLDC.

LISTA DE FIGURAS

1	Relação torque <i>VS</i> Velocidade para diferentes aplicações do <i>BLDC</i>	12
2	Partes motor CC	14
3	Representação do rotor no interior do motor CC com escovas	15
4	Escovas de um motor CC	16
5	Escova danificada pelo uso constante	17
6	Comutador mecânico motor CC	17
7	Rotor de uma espira envolto por estator de ímã permanente	18
8	Tensão induzida em um espira em rotação	19
9	Modelo equivalente do motor CC	20
10	Representação física do BLDC	22
11	Modelo equivalente do motor BLDC	23
12	Forma de onda esperada para o ensaio com máquina primária	24
13	Representação em blocos do <i>hardware</i> de acionamento do <i>BLDC</i>	26
14	Acionamento em <i>Half-Bridge</i>	27
15	Acionamento <i>Full-Bridge</i> com motor Y	27
16	Acionamento <i>Full-Bridge</i> com motor em D	28
17	Acionamento com quatro chaves	28
18	Forma de acionamento do <i>BLDC</i>	29
19	Curva de acionamento do MOSFET	31
20	Representação das capacitâncias internas do MOSFET	31
21	Processo de alinhamento do rotor	34
22	Medição de corrente para detecção da posição inicial	35
23	Aplicação das quatro formas de acionamento para as 3 fases do motor	35
24	Fusível de vidro em um circuito eletrônico	37
25	Trilha fusível em zigue-zague	39
26	Fluxograma de desenvolvimento do projeto	40
27	Chaves de potência	42
28	Circuito dos <i>gate drivers</i>	43
29	Circuito de leitura de corrente	45
30	Circuito de leitura da <i>BEMF</i>	46
31	Representação simplificada do sistema de controle	46
32	Circuito do microcontrolador	47
33	Circuito da fonte de +15 V	48
34	Circuito da fonte de +5 V	49
35	Trilha fusível desenhada na PCI	50
36	Circuito de proteção contra sobretensão e <i>load-dump</i>	51
37	Diagrama simplificado do processamento não crítico	52
38	Diagrama simplificado do processamento crítico	52
39	Resultado obtido do ensaio de resistência entre as fases A e C	54
40	Resultado obtido do ensaio de resistência entre as fases A e B	55
41	Resultado obtido do ensaio de resistência entre as fases B e C	55
42	Forma de onda obtida no ensaio com máquina primária	56
43	Bancada de testes para determinação dos parâmetros	57
44	Formas de onda obtidas no ensaio de corrente alternada	57
45	Formas de onda da tensão entre as fases	58
46	Formas de onda da tensão entre as fases	59

47	Corrente e tensão com o motor submetido ao Torque nominal	59
48	Análise termográfica na região do <i>fuse-track</i> para diferentes correntes	60
49	Danos ocasionadas na PCI pela atuação da proteção	60
50	Opção de religamento de <i>fuse-track sobressalente</i>	61

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVOS	13
1.1.1	Objetivo Geral	13
1.1.2	Objetivos Específicos	13
1.2	ESCOPO E RESTRIÇÕES	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	MOTORES ELÉTRICOS DE CORRENTE CONTÍNUA	14
2.1.1	Estator do motor CC	15
2.1.2	Rotor do motor CC	15
2.1.3	Escovas	16
2.1.4	Comutador	17
2.1.5	Relação entre campo e armadura	18
2.1.6	Modelo do motor CC	20
2.1.7	Motor BLDC	21
2.1.8	Modelo equivalente do motor CC de três terminais	22
2.2	PARÂMETROS	23
2.2.1	Resistência entre dois enrolamentos	23
2.2.2	Força eletromotriz	24
2.2.3	Indutância	25
2.3	ACIONAMENTO DO MOTOR	25
2.3.1	Inversor	26
2.3.1.1	Half-bridge	26
2.3.1.2	Full-bridge	26
2.3.1.3	Quatro chaves	28
2.3.2	Formas de onda	28
2.3.3	Chaves de potência	30
2.3.3.1	Mosfet	30
2.3.3.2	Perdas no Mosfet	32
2.4	CONTROLE DO MOTOR	33
2.4.1	Determinação da posição inicial	33
2.4.1.1	Posicionamento forçado	33
2.4.1.2	Estimador de posição por corrente	34
2.4.1.3	Procedimento de aceleração	36
2.5	CIRCUITO DE CONTROLE E PROTEÇÃO	36
2.5.1	Corrente	36
2.5.2	Sobretensão	39
2.5.3	Subtensão	39
3	MÉTODO	40
3.1	DESENVOLVIMENTO DO HARDWARE	41
3.1.1	Circuito de potência	41
3.1.1.1	Chaves de potência	41
3.1.1.2	Gate drivers	42
3.1.2	Circuitos de controle	43
3.1.2.1	Leitura de corrente	43

3.1.2.2	Leitura de BEMF	45
3.1.2.3	Microcontrolador	47
3.1.3	Fontes	48
3.1.4	Proteções	49
3.1.4.1	Curto-circuito	49
3.1.4.2	<i>Load-Dump</i>	50
3.2	DESENVOLVIMENTO DO FIRMWARE	51
4	RESULTADOS	54
4.1	PARÂMETROS DO MOTOR	54
4.2	FORMAS DE ONDA OBTIDAS NO ACIONAMENTO	58
4.3	PROTEÇÕES	60
5	CONSIDERAÇÕES FINAIS	62
5.1	DISCUSSÃO DOS RESULTADOS	62
5.2	TRABALHOS FUTUROS	62
	REFERÊNCIAS	64
	APÊNDICE A - PROJETO DA PLACA DE CIRCUITO IMPRESSO	66
	APÊNDICE B - PROJETO MONTADO	68
	APÊNDICE C - ESQUEMÁTICO COMPLETO DAS FONTES	69
	APÊNDICE D - ESQUEMÁTICO COMPLETO DOS GATE-DRIVERS	70
	APÊNDICE E - ESQUEMÁTICO COMPLETO DAS ENTRADAS ANALÓGICAS/DIGITAIS	71
	APÊNDICE F - ESQUEMÁTICO COMPLETO PARA LEITURA DA BEMF	72
	APÊNDICE G - ESQUEMÁTICO COMPLETO DAS CHAVES DE POTÊNCIA	73
	APÊNDICE H - ESQUEMÁTICO COMPLETO DOS TESTES DE MICROCONTROLADOR	74
	APÊNDICE I - ESQUEMÁTICO COM A REPRESENTAÇÃO DA LIGAÇÃO ENTRE OS CIRCUITOS	75
	APÊNDICE J - MÁQUINA DE ESTADOS PARA PROCESSAMENTO NÃO CRÍTICO	76
	APÊNDICE K - CÓDIGO DE PROCESSAMENTO CRÍTICO	79
	ANEXO A - DADOS TÉCNICOS DO PJD45N06	94
	ANEXO B - DADOS TÉCNICOS DO ACS723	95

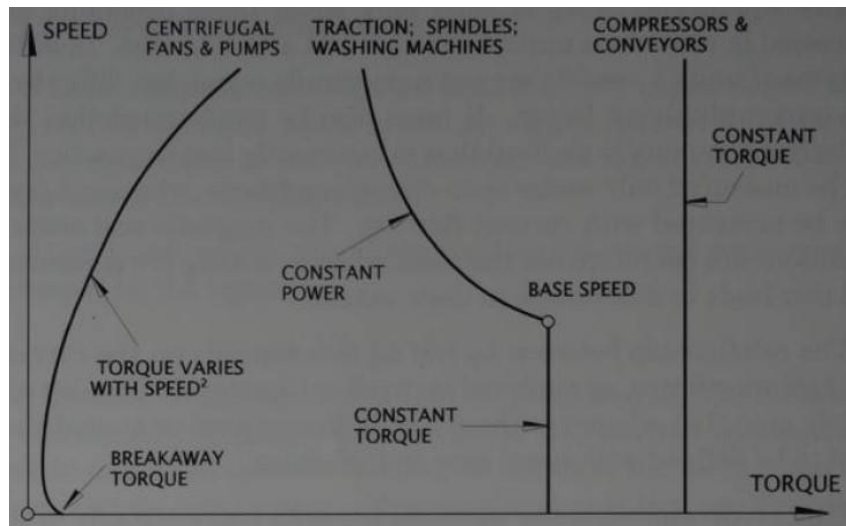
1 INTRODUÇÃO

Motores de corrente contínua sem escovas (do inglês, *BLDC - Brushless Direct Current*) são máquinas elétricas que compartilham muitas características com os demais motores CC, diferindo principalmente por não possuírem comutadores mecânicos, utilizando obrigatoriamente ímãs permanentes.

Embora os motores *BLDC* possuam um custo elevado quando comparado aos motores de corrente contínua comuns, os mesmos requerem menor número de manutenções, tendo assim uma confiabilidade e durabilidade maior (CHAPMAN, 2013). Além disso, por possuírem elevada eficiência, os motores *BLDC* passaram a ser amplamente utilizados na indústria da aviação, militar e em sistemas automotivos (KARTHIKEYAN; SEKARAN, 2011). A exemplo do Brasil, um dos principais consumidores de motores *BLDC* é o setor automotivo, que faz uso deste tipo de motor em refrigeradores (ALBUQUERQUE, 2020).

Sendo o principal componente nos refrigeradores automotivos, o *BLDC* é utilizado na compressão do gás refrigerante. Em compressores, a relação torque versus velocidade torna-se aproximadamente linear (GIESEL, 2007). Essa relação pode ser vista na Figura 1.

Figura 1: Relação torque VS Velocidade para diferentes aplicações do *BLDC*



Fonte: (MILLER, 2010)

Apesar de sua simplicidade construtiva, a complexidade do sistema de acionamento é maior para o *BLDC* do que motores CC comuns. Diversas técnicas podem ser empregadas tanto para o circuito de potência quanto para o controle de posição do rotor, tendo cada uma delas vantagens e desvantagens.

O drive de acionamento pode permanecer em operação por longos períodos de tempo, além de estar sujeito a vibrações, choques mecânicos e altas temperaturas. Assim, o desenvolvimento de um drive para acionamento do *BLDC* encontra diversos desafios, tanto em relação ao

desenvolvimento do *software* embarcado para controle do sistema, quanto na implementação do *hardware*. Além disso, o *hardware*, além de atender a todos os requisitos de operação do sistema, deve possuir um baixo custo.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

Desenvolver um *drive* para acionamento e controle de motores de corrente contínua sem escovas, para aplicação em compressores de geladeiras automotivas.

1.1.2 Objetivos Específicos

- Realizar a caracterização do motor utilizado;
- Caracterizar a carga;
- Selecionar e caracterizar a topologia de acionamento do *BLDC*;
- Implementar o hardware proposto;
- Desenvolver o software embarcado para acionamento e controle de velocidade do motor *BLDC*;
- Comparar os resultados teóricos e os práticos.

1.2 ESCOPO E RESTRIÇÕES

Como definições e restrições técnicas para o desenvolvimento do trabalho, pode-se apontar:

- O circuito deve ser alimentado por uma bateria automotiva de 24 V com capacidade de fornecimento de corrente de no mínimo 10 A;
- Os componentes eletrônicos utilizados devem atender requisitos de custo e disponibilidade comercial;
- Deve-se garantir a reusabilidade tanto do hardware quanto do firmware para aplicações similares.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 MOTORES ELÉTRICOS DE CORRENTE CONTÍNUA

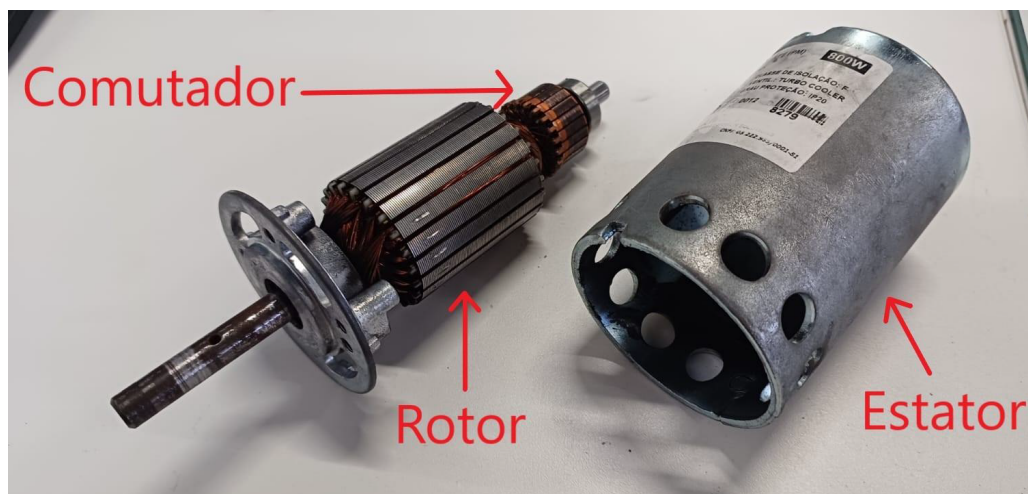
Máquinas elétricas são dispositivos capazes de converter energia elétrica em energia mecânica e vice-versa (CHAPMAN, 2013). Quando tal dispositivo é utilizado para converter energia elétrica em energia mecânica, este é denominado como motor elétrico. Quanto à forma de alimentação, os motores elétricos podem ser divididos em dois grupos: Motores de Corrente Contínua (CC) e Motores de Corrente Alternada (CA).

Ainda, dentro dos motores de corrente contínua, pode-se distingui-los em dois tipos: motores com escovas, também conhecidos como motores CC convencionais, e motores sem escovas, também conhecidos como *BLDC*. O *BLDC*, foco deste trabalho, trata-se de uma derivação construtiva direta dos motores CC convencionais. Nas seções que seguem, será realizado uma revisão acerca do funcionamento e caracterização deste tipo de motor.

Semelhantemente aos motores CA, os motores CC possuem em seu interior tensões e correntes alternadas (CHAPMAN, 2013). Entretanto, do ponto de vista da fonte, no motor CC a corrente flui somente em um sentido. Tal característica é possível somente porque o motor CC possui um mecanismo chamado comutador. O motor CC com escovas possui em seu interior a combinação entre um comutador e um conjunto de escovas, que forma um retificador mecânico (UMANS, 2014).

Os motores CC convencionais são compostos, basicamente, de quatro partes: escovas, comutador, rotor e estator (CHAPMAN, 2013). Um motor CC com o comutador, rotor e o estator pode ser visto na Figura 2.

Figura 2: Partes motor CC



Fonte: Autor (2023)

No caso do *BLDC* entretanto, não há a presença de escovas e a comutação deixa de ser mecânica e passa a ser eletrônica.

2.1.1 Estator do motor CC

No motor de corrente contínua convencional, o estator, ou parte estacionária, é o que fornece suporte físico ao motor, além de conter as peças polares que propiciam um caminho para o fluxo magnético da máquina (CHAPMAN, 2013).

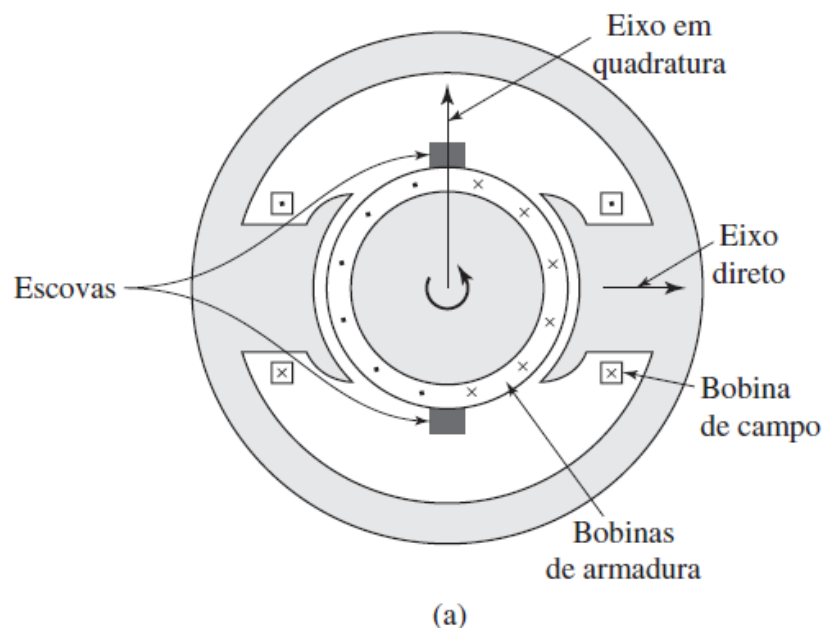
É justamente no estator onde é produzido o campo magnético que interagirá diretamente com o campo magnético do rotor. A produção de campo magnético no estator do motor CC com escovas pode se dar por meio de um conjunto de bobinas, conhecidos como enrolamento de campo, ou através da utilização de ímãs permanentes.

No *BLDC* entretanto, o estator é a parte mais interna do motor, tendo a função de produzir o campo magnético principal, sempre por meio de bobinas, conhecidas como enrolamento de armadura. A relação entre os campos magnéticos do rotor e do estator, será discutida posteriormente.

2.1.2 Rotor do motor CC

A parte rotativa de uma máquina CC é denominada rotor (CHAPMAN, 2013). No motor CC com escovas, o rotor fica localizado no centro da máquina, como pode ser visto na Figura 3. O rotor do motor CC com escovas possui outro conjunto de espiras denominadas de enrolamento de armadura. Estas espiras são responsáveis por gerar a FMM (Força Magneto-Motriz) que interagirá com o campo produzido no estator.

Figura 3: Representação do rotor no interior do motor CC com escovas



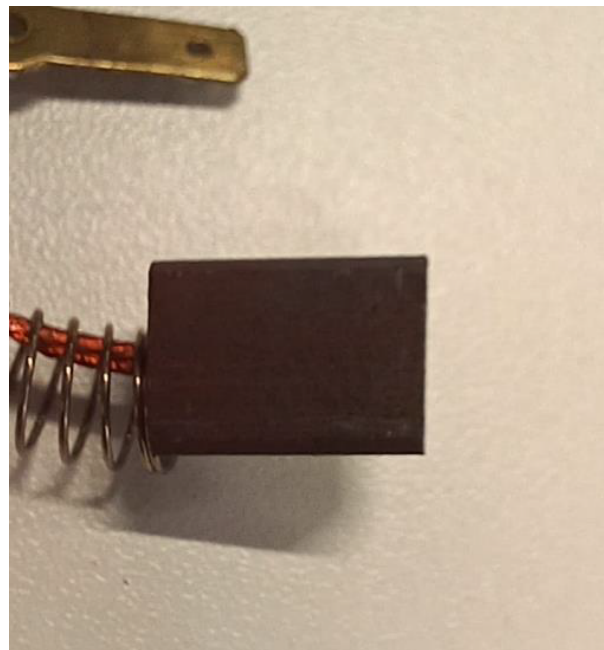
Sendo o enrolamento de armadura responsável pela maior parte do consumo de energia em um motor CC e, estando o rotor em movimento, é necessário que a energia aplicada aos terminais externos da máquina seja conectada às bobinas por meio de um elemento eletromecânico. Tal elemento é conhecido como escova.

No *BLDC*, entretanto, o rotor não possui enrolamentos, tendo seu campo magnético produzido por ímãs permanentes. Além disso, o rotor é posicionado na parte externa do *BLDC*. Desta forma, a energia aplicada aos terminais externos da máquina não precisa ser entregue à parte rotativa.

2.1.3 Escovas

Geralmente fabricadas em carbono, as escovas, conectam os terminais externos do motor CC às faces do comutador do rotor. Uma escova comum pode ser vista na Figura 4. Durante uma revolução completa do rotor ocorrem diversas comutações. No decorrer desse processo, as escovas são desconectadas de um conjunto de espiras e reconectadas ao conjunto seguinte de forma sucessiva.

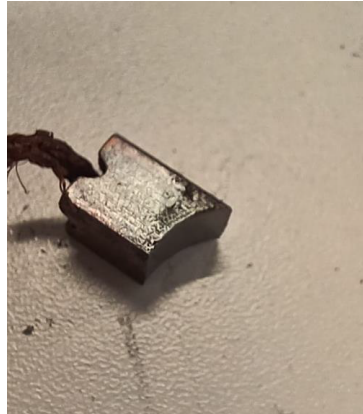
Figura 4: Escovas de um motor CC



Fonte: Autor (2023)

A comutação entre as espiras do enrolamento de armadura faz com que haja picos indutivos de tensão, podendo ocasionar faiscamento, o que gera deterioração das escovas. O resultado disso é o desgaste das escovas tanto por atrito quanto pelo faiscamento, podendo causar até mesmo a deformação das mesmas, como é observado na Figura 5. Esse desgaste é inerente ao funcionamento do motor CC com escovas e faz com que sejam necessárias manutenções periódicas no motor.

Figura 5: Escova danificada pelo uso constante



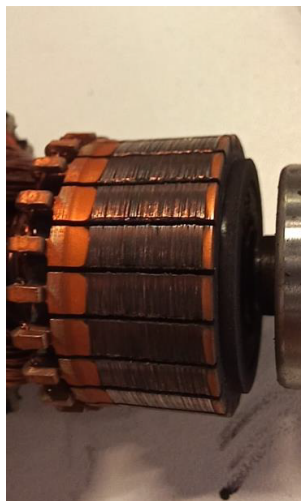
Fonte: Autor (2023)

O desgaste das escovas faz com que a vida útil do motor seja reduzida. Como o *BLDC* não possui escovas para serem desgastadas, além de uma maior eficiência energética, a vida útil total do motor é superior.

2.1.4 Comutador

Sendo a armadura do motor CC convencional composta por diversas espiras, o comutador tem por função conectar corretamente as escovas ao conjunto correto de espiras a cada instante. Essa conexão deve ser tal que a polaridade geral do motor através das espiras seja mantida. Em geral, o comutador é composto por barras de cobre isoladas entre si por um material geralmente à base de mica (CHAPMAN, 2013). Tal característica assegura seu funcionamento ao longo de toda a vida útil do motor. As faces do comutador de um motor CC com escovas são vistas na Figura 6.

Figura 6: Comutador mecânico motor CC



Fonte: Autor (2023)

2.1.5 Relação entre campo e armadura

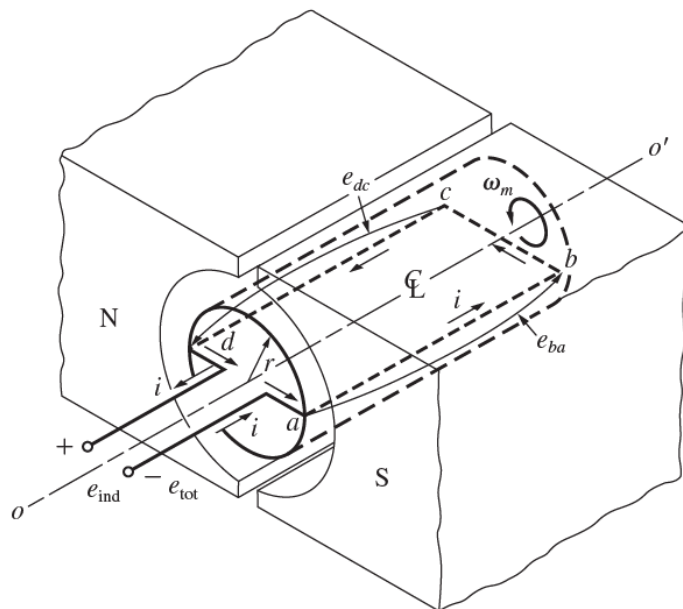
Para compreender a relação entre o campo magnético do rotor e do estator, pode-se realizar algumas aproximações. Considera-se primeiramente o estator constituído de um material ferromagnético duro (ímã permanente) com o rotor posicionado no centro. O rotor é então considerado como contendo uma única espira, conforme Figura 7 e estando em rotação. Assim, segundo CHAPMAN (2013), a tensão induzida e_{ind} em um segmento da espira é dada pela Equação (2.1).

$$e_{ind} = (\mathbf{v} \times \mathbf{B}) \cdot \mathbf{l} \quad (2.1)$$

Onde:

- v - Velocidade do condutor;
- \mathbf{B} - Vetor densidade de fluxo magnético;
- \mathbf{l} - Comprimento do condutor dentro do campo magnético

Figura 7: Rotor de uma espira envolto por estator de ímã permanente

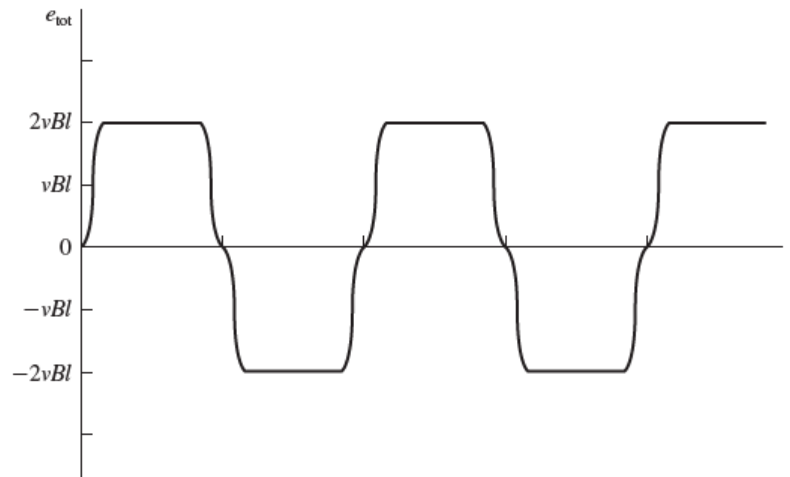


Fonte: (CHAPMAN, 2013)

Sabendo que o rotor possui dois segmentos de espira imersos no campo magnético do estator, e que os campos estão a 90 graus um do outro, a partir da Equação (2.1), a tensão induzida na espira é dada pela Equação (2.2). Conforme dito anteriormente, o motor CC possui em seu interior uma tensão induzida alternada, conforme Figura 8.

$$e_{ind} = 2vBl \quad (2.2)$$

Figura 8: Tensão induzida em um espira em rotação



Fonte: (CHAPMAN, 2013)

Ainda segundo CHAPMAN (2013), considerando a densidade de fluxo magnético \mathbf{B} constante, e a velocidade tangencial φ em função da velocidade angular ω_m , a tensão induzida na espira pode ser reescrita como na Equação (2.3).

$$e_{ind} = \frac{2}{\pi} \varphi \omega_m \quad (2.3)$$

CHAPMAN (2013) pontua também que, ao se aplicar uma tensão à essa espira, haverá o surgimento de uma corrente elétrica. Essa corrente elétrica por sua vez produz um campo magnético e, conseqüentemente, a espira estará sujeita à uma força \mathbf{F} , conforme a Equação (2.4).

$$\mathbf{F} = i(\mathbf{l} \times \mathbf{B}) \quad (2.4)$$

De modo semelhante à tensão induzida na espira, a Equação (2.4) deixa evidente que, nas seções da espira onde o campo magnético produzido é paralelo ao campo do estator, a força resultante é zero. Logo, pode-se considerar a espira como dois segmentos retos de um condutor, de modo que a partir da Equação (2.4), o torque resultante é calculado na Equação (2.5).

$$T_{res} = rF \text{sen}\theta \quad (2.5)$$

Onde:

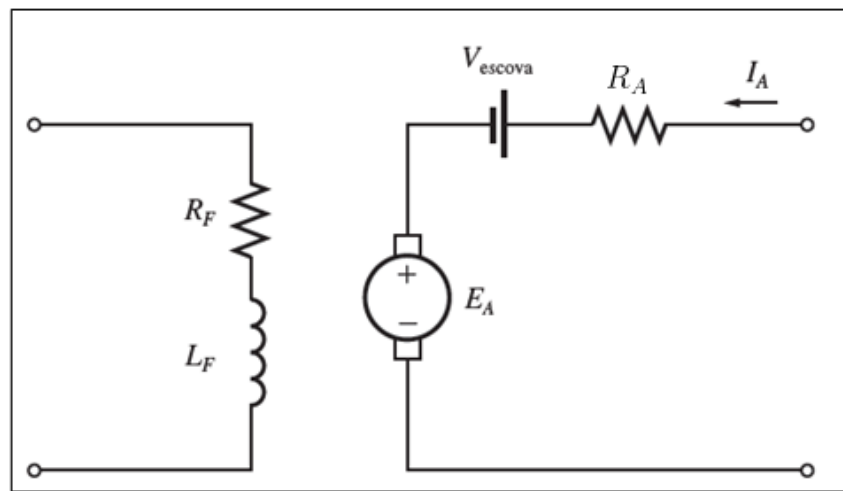
- T_{res} - Torque resultante;
- r - Raio de rotação da espira;

- θ - Ângulo entre r e F

2.1.6 Modelo do motor CC

Visando facilitar o equacionamento do motor CC, é possível realizar algumas aproximações, criando um modelo simplificado. A Figura 9 representa o modelo elétrico equivalente do motor CC. O motor CC é modelado por uma resistência R_A (resistência dos enrolamentos de armadura), uma fonte tensão E_A (tensão induzida na armadura) e por uma resistência R_F (resistência dos enrolamentos do campo).

Figura 9: Modelo equivalente do motor CC



Fonte: (CHAPMAN, 2013)

Ainda, nos motores de ímã permanente, onde não há a presença do enrolamento de campo, o mesmo é suprimido do modelo equivalente, sendo considerado apenas uma constante nas equações. Além disso, como as perdas nas escovas geralmente são pequenas, usualmente estas são suprimidas do modelo equivalente.

Em uma máquina CC real, ao invés de uma única espira na armadura, como foi visto anteriormente, têm-se diversas espiras compostas por um ou mais condutores em paralelo. Desta forma, diversas características construtivas devem ser levadas em consideração. Entretanto, esses valores geralmente são constantes. A partir da Equação (2.3) a tensão induzida na armadura é calculada conforme a Equação (2.6).

$$E_A = \mathbf{K} \cdot \phi \cdot \omega_m \quad (2.6)$$

Onde:

- \mathbf{K} - Constante construtiva do motor;
- ϕ - Fluxo total por polo;

- ω_m - Velocidade angular do rotor;

Considerando o fluxo magnético constante, é possível agrega-lo à constante construtiva K
Ou seja:

$$K' = K \cdot \phi \quad (2.7)$$

Aplicando a Equação (2.7) na Equação (2.6), a tensão na armadura pode ser calculada a partir da Equação (2.8)

$$E_A = K' \cdot \omega_m \quad (2.8)$$

De forma semelhante, o torque induzido no rotor T_i segundo CHAPMAN (2013) pode ser calculado utilizando a Equação (2.9).

$$T_i = K \cdot \phi \cdot I_A \quad (2.9)$$

Da mesma forma, substituindo a Equação (2.7) na Equação (2.8), o torque induzido no eixo de um motor CC é dado pela Equação (2.10).

$$T_i = K' \cdot I_A \quad (2.10)$$

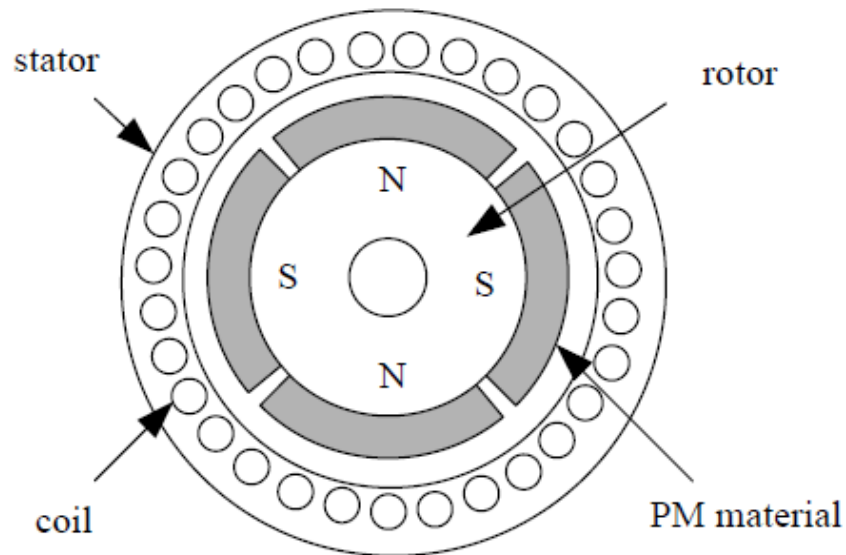
Conhecendo o modelo equivalente e as equações básicas que regem o comportamento do motor CC convencional, é possível discutir o funcionamento do motor *BLDC*.

2.1.7 Motor BLDC

O motor de corrente contínua sem escovas é desenvolvido tendo por base o motor CC com escovas. Entretanto, o motor *BLDC* possui o enrolamento de armadura no estator e, no rotor em substituição ao enrolamento de campo, são utilizados ímãs permanentes (KRISHNAN, 2001). Com isso, os comutadores mecânicos e as escovas, antes utilizados para conectar o enrolamento de armadura, podem ser substituídos por um inversor eletrônico (OGASAWARA; AKAGI, 1991).

Além disso, diferentemente dos demais motores elétricos, o motor *BLDC* é acionado com uma forma de onda específica (discutida posteriormente), o que confere uma alta utilização magnética, permitindo um tamanho reduzido, alto torque e alta eficiência. Na Figura 10 pode ser visto a representação simplificada do *BLDC*.

Figura 10: Representação física do BLDC



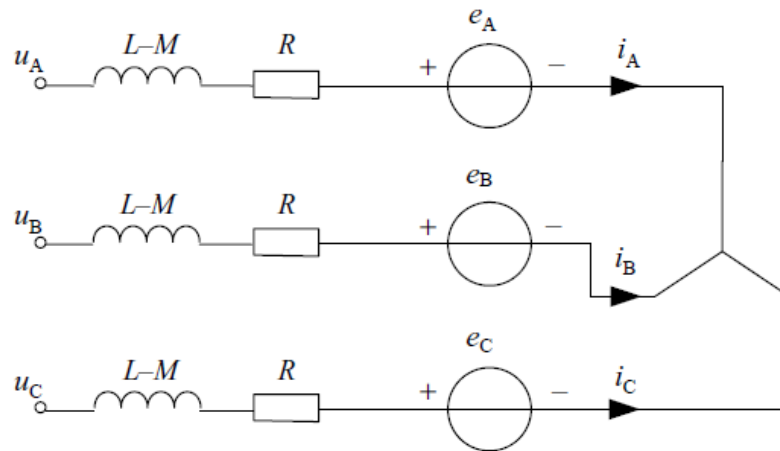
Fonte: (XIA, 2012)

De modo a possibilitar a produção de torque em todas as posições do rotor, o *BLDC* apresenta mais que uma fase, sendo a utilização de três fases a mais comum (HANSELMAN, 2006). Um *BLDC* trifásico possui uma resolução de rotação de 120° elétricos ou 60° mecânicos (HANSELMAN, 2006). Deste modo, a forma de onda da corrente também pode ser chamada de onda quadrada de 120° (MILLER, 2010).

2.1.8 Modelo equivalente do motor CC de três terminais

O modelo equivalente do *BLDC* de três terminais deriva diretamente do modelo do motor CC monofásico com escovas. Realizando as devidas alterações no circuito visto na Figura 9, o modelo equivalente do *BLDC* de três terminais pode ser visto na Figura 11.

Figura 11: Modelo equivalente do motor BLDC



Fonte: (XIA, 2012)

2.2 PARÂMETROS

Construtivamente, as três fases do *BLDC* são fabricadas para que contenham as mesmas características entre si. Entretanto, para uma maior exatidão na determinação dos parâmetros, é importante que todas as fases do motor passem pelos procedimentos de teste e, caso uma diferença superior a 3% entre as fases seja detectada, há a indicação de que o motor esteja com defeito (MILLER, 2010).

2.2.1 Resistência entre dois enrolamentos

Como a resistência dos enrolamentos afeta diretamente a corrente da armadura do *BLDC*, e por conseguinte, a resposta dinâmica do torque eletromagnético, sua determinação se torna relevante para o desenvolvimento do sistema de controle de velocidade do *BLDC* (XIA, 2012). KRISHNAN (2001) sugere que a resistência entre dois terminais da máquina pode ser estimada aplicando uma tensão CC entre eles e, assim, medir a magnitude da corrente. Com os valores de tensão e corrente, a resistência equivalente R_A pode ser determinada relacionando a tensão V_A aplicada aos terminais da máquina e a corrente I_A resultante conforme Equação 2.11.

$$R_A = \frac{V_A}{I_A \cdot 2} \quad (2.11)$$

Na Equação 2.11 a razão é dividida por dois em função de apresentar duas resistências em série para a ligação em Y. UMANS (2014) sugere que o valor de tensão aplicado nos terminais da máquina deve ser tal que a corrente produzida atinja o valor da corrente de regime, fazendo assim, com que a temperatura do condutor seja aproximadamente a temperatura de operação da máquina, fazendo com que o valor obtido seja o mais próximo da resistência de quando o motor

está operando em regime permanente.

2.2.2 Força eletromotriz

Estando diretamente relacionado à produção de torque (MILLER, 2010) e sendo o parâmetro que possibilita a detecção da posição elétrica do rotor (SHAO JIANWEN; NOLAN, 2002), a FEM (Força Eletromotriz) deve ser determinada.

Segundo CHAPMAN (2013), a partir do modelo equivalente visto na Figura 9, a tensão da armadura é dada pela Equação 2.12.

$$V_T = E_A - I_A \cdot R_A \quad (2.12)$$

Deixando os terminais de alimentação da armadura em aberto, faz-se $I_A = 0$. Assim, a Equação 2.8 é reescrita conforme Equação 2.13.

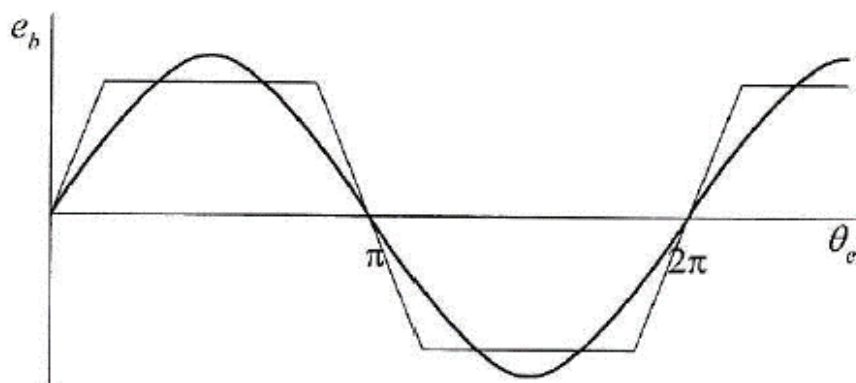
$$\begin{aligned} V_T &= E_A - 0 \cdot R_A \\ V_T &= E_A \end{aligned} \quad (2.13)$$

Nessas condições, com a Equação 2.13, é possível medir a tensão de armadura a partir dos terminais externos da máquina. Assim, utilizando uma máquina primária conectada ao rotor do *BLDC* fazendo com que este gire à uma velocidade conhecida, a partir da Equação 2.8, a constante construtiva pode ser calculada conforme a Equação 2.14.

$$K' = \frac{V_T}{\omega_m} \quad (2.14)$$

Ao realizar este ensaio em um *BLDC*, a tensão V_T apresentará uma forma de onda alternada, semelhante a representada pela Figura 12.

Figura 12: Forma de onda esperada para o ensaio com máquina primária



Fonte: (HANSELMAN, 2006)

Assim, alternativamente à Equação 2.14, a constante K' pode ser determinada com a Equação

ção 2.15 (MILLER, 2010) (KRISHNAN, 2001).

$$K' = \frac{|e_b|}{\omega_m} \quad (2.15)$$

Sendo que:

- $|e_b|$ - Módulo da tensão de pico induzida;
- ω_m - Velocidade angular do eixo;

2.2.3 Indutância

Segundo MILLER (2010) embora as componentes indutivas do *BLDC* estejam relacionadas à produção de torque e ao ripple de corrente devido à frequência de chaveamento, sua determinação aproximada, do ponto de vista do controle, já é suficiente. KRISHNAN (2001) sugere que, com o rotor bloqueado, seja aplicada uma tensão CA (V_{12}), com uma frequência conhecida (f_s), entre dois terminais da máquina. Com a corrente resultante I_{12} , conhecendo a resistência entre as duas fases (R_{12}), a indutância equivalente entre essas duas fases (L_{12}) pode ser determinada utilizando a Equação 2.16.

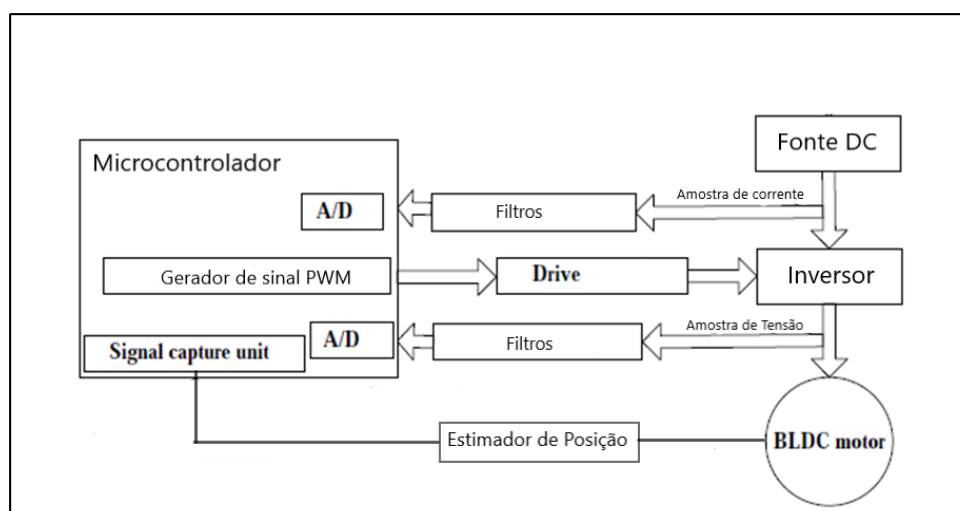
$$L_{12} = \frac{\sqrt{\left(\frac{V_{12}}{I_{12}}\right)^2 - R_{12}^2}}{2 \cdot \pi \cdot f_s} \quad (2.16)$$

2.3 ACIONAMENTO DO MOTOR

Embora os motores *BLDC* sejam máquinas de construção simples, diferentemente dos motores CC comuns, os mesmos não podem ser ligados diretamente à fonte de alimentação, sendo necessário um sistema próprio para controle e acionamento das fases do mesmo (UMANS, 2014).

XIA (2012) divide o sistema de controle em duas partes: *Hardware* e *Software*. Para UMANS (2014), o *Hardware* básico de acionamento de um motor *BLDC* é composto por um sensor de posição do rotor, um controlador e um inversor, como pode ser visto na Figura 12. Aplicando-se algumas técnicas, como sugere KRISHNAN (2001), o sensor de posição do rotor, pode ser substituído por um estimador de posição.

Figura 13: Representação em blocos do *hardware* de acionamento do *BLDC*



Fonte: Adaptado de XIA (2012)

2.3.1 Inversor

XIA (2012) apresenta três métodos principais para acionamento do *BLDC*: *Half-Bridge*, *Full-Bridge* e o acionamento por quatro chaves.

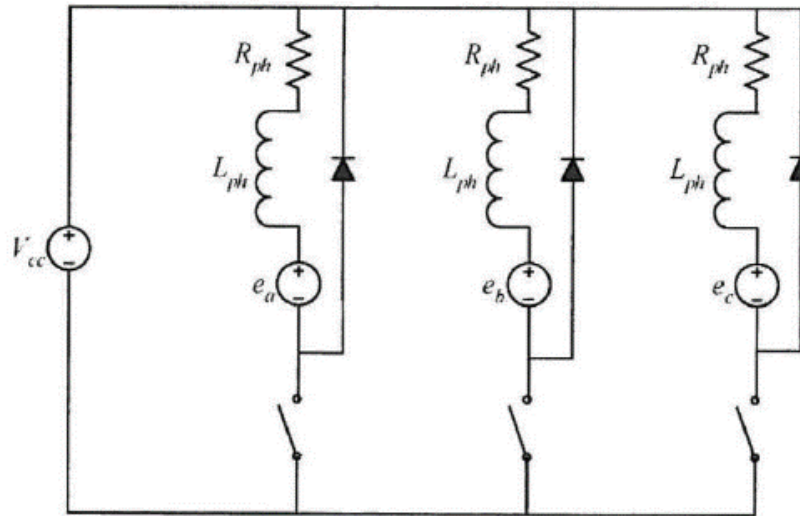
2.3.1.1 Half-bridge

O método de acionamento em *Half-Bridge* é utilizado especialmente em motores com ligação Y (XIA, 2012). No acionamento *Half-Bridge*, o ponto comum do motor pode ser ligado ao terminal positivo ou negativo da fonte, enquanto as chaves de potência são ligadas à cada fase do motor, de modo a alimentar cada uma das fases com o sinal oposto ao ligado no ponto comum. A partir do modelo equivalente do *BLDC* trifásico visto na Figura 11 e utilizando chaves genéricas, o arranjo das chaves de acionamento em *Half-Bridge* pode ser visto na Figura 14.

A vantagem do acionamento em *Half-Bridge* se deve à utilização de somente três chaves de potência, o que reduz o custo do projeto, quando comparado com os demais métodos de acionamento. Entretanto, segundo XIA (2012), o controle de rotação do motor acaba tendo uma redução na resolução, o que acarreta em um grande ripple de torque, além de uma baixa utilização dos enrolamentos.

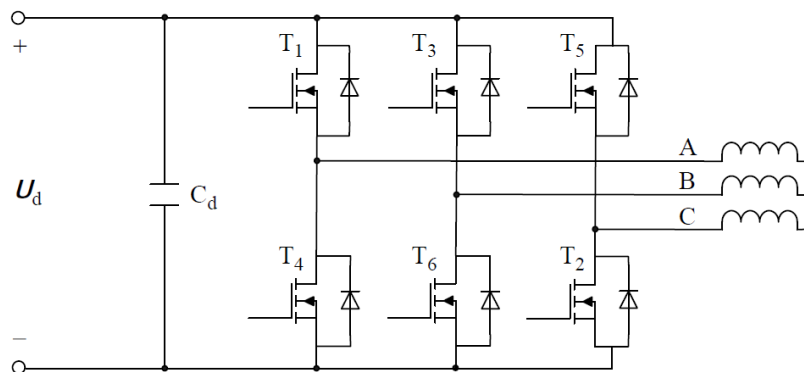
2.3.1.2 Full-bridge

O acionamento em *Full-Bridge* consiste na utilização de seis chaves para o controle das três fases do motor. Nessa configuração duas chaves, uma *high-side* e uma *low-side* são acionadas ao mesmo tempo (OGASAWARA; AKAGI, 1991). A desvantagem deste método se deve tanto

Figura 14: Acionamento em *Half-Bridge*

Fonte: (HANSELMAN, 2006)

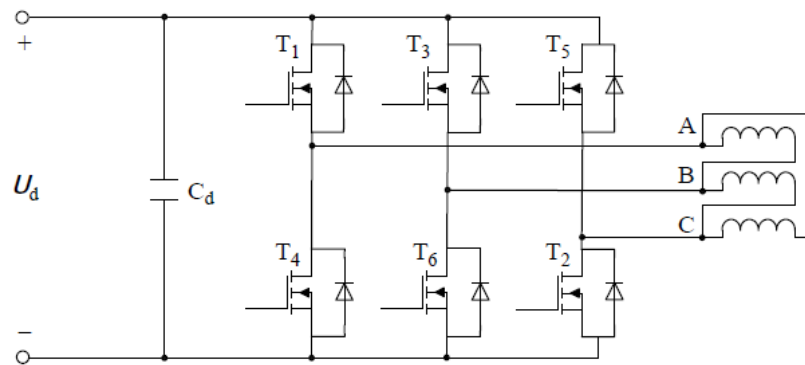
às perdas nas chaves, quanto ao custo, visto que o dobro de chaves, comparado com o método em *half-bridge*, são utilizadas. Apesar disso, tal método oferece flexibilidade no controle e um bom aproveitamento da tensão de alimentação (MILLER, 2010). A Figura 15 representa o conjunto de chaves para o acionamento em Full-Bridge, com o motor ligado em Y.

Figura 15: Acionamento *Full-Bridge* com motor Y

Fonte: (XIA, 2012)

A Figura 16 representa o mesmo arranjo de acionamento, mas com o motor ligado em Delta.

Figura 16: Acionamento *Full-Bridge* com motor em D

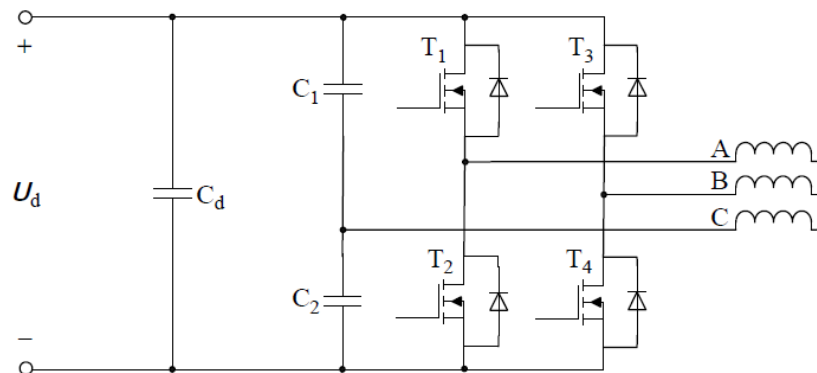


Fonte: (XIA, 2012)

2.3.1.3 Quatro chaves

Também conhecido como acionamento por divisor de tensão, este método de acionamento consiste em controlar diretamente somente duas fases do motor, utilizando quatro chaves para isso, deixando a terceira fase conectada em um divisor de tensão, como é visto na Figura 17.

Figura 17: Acionamento com quatro chaves



Fonte: (XIA, 2012)

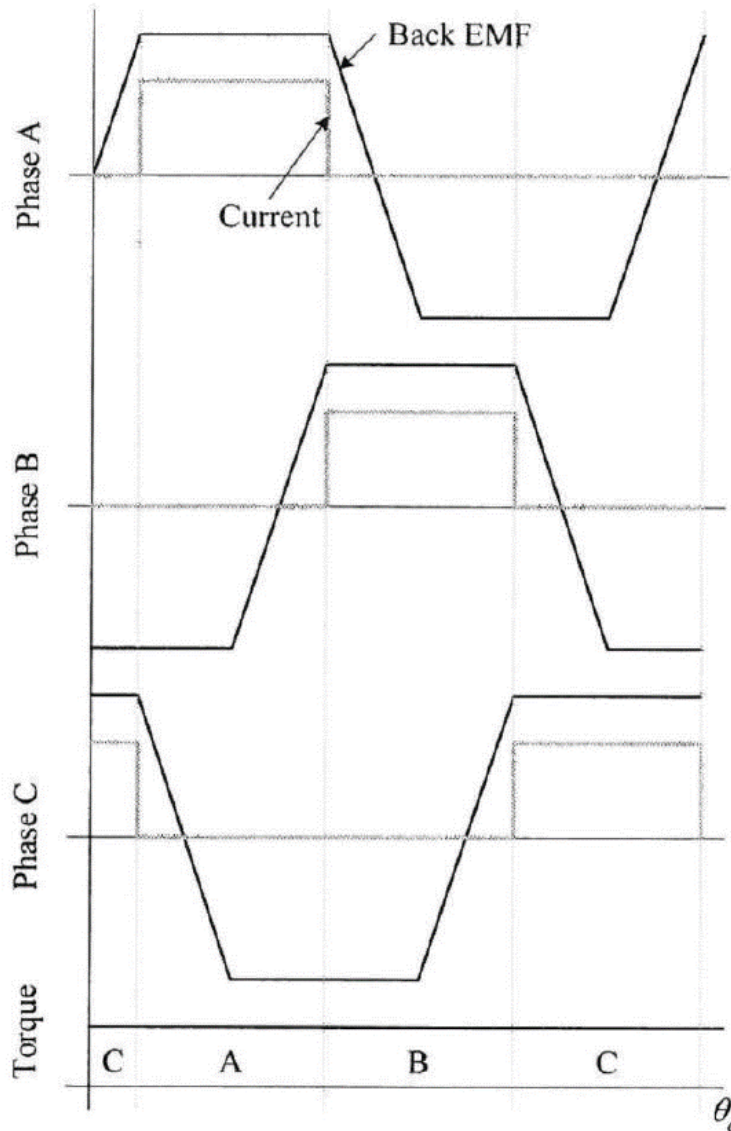
XIA (2012) aponta que embora tal método de acionamento utilize duas chaves a menos, o que acarreta em menor custo, o algoritmo de controle torna-se mais complexo, quando comparado com os demais métodos de acionamento.

2.3.2 Formas de onda

Como dito anteriormente, para o correto funcionamento, o *BLDC* possui uma forma específica de comutação e acionamento das fases. MILLER (2010) denomina esse acionamento como *squarewave control* (controle em onda quadrada). Ainda, segundo os autores, as formas de onda

do acionamento possuem um formato de onda quadrada, com o topo achatado, podendo compor uma onda trapezoidal. A Figura 18 representa a forma de onda da tensão e da corrente em um *BLDC* trifásico.

Figura 18: Forma de acionamento do *BLDC*



Fonte: (HANSELMAN, 2006)

Ao acionar duas fases do motor, é provocada uma rotação no rotor, buscando o alinhamento dos campos do motor. Quando o campo produzido pelo estator encontra-se praticamente alinhado com o campo dos ímãs do rotor, uma das fases é desligada, e outra é acionada, provocando um novo desalinhamento dos campos e, conseqüentemente, a rotação do rotor.

De modo a se obter uma resolução de rotação de 60° , MILLER (2010) propõe a seqüência de acionamento observada na Tabela 1.

O valor “+1” indica que a fase está conectada ao +VCC, através da chave de *high-side*; ao passo que o valor “-1” indica que a fase está conectada ao GND, através da chave de *low-side*.

Tabela 1: Tabela de acionamento para resolução de 60°

Posição do rotor [° elétricos]	Linha			Fase A		Fase B		Fase C	
	A	B	C	Q1	Q2	Q3	Q4	Q5	Q6
330 - 30	0	-1	+1	0	0	0	1	1	0
30 - 90	+1	-1	0	1	0	0	1	0	0
90 - 150	+1	0	-1	1	0	0	0	0	1
150 - 210	0	+1	-1	0	0	1	0	0	1
210 - 270	-1	+1	0	0	1	1	0	0	0
270 - 330	-1	0	+1	0	1	0	0	1	0

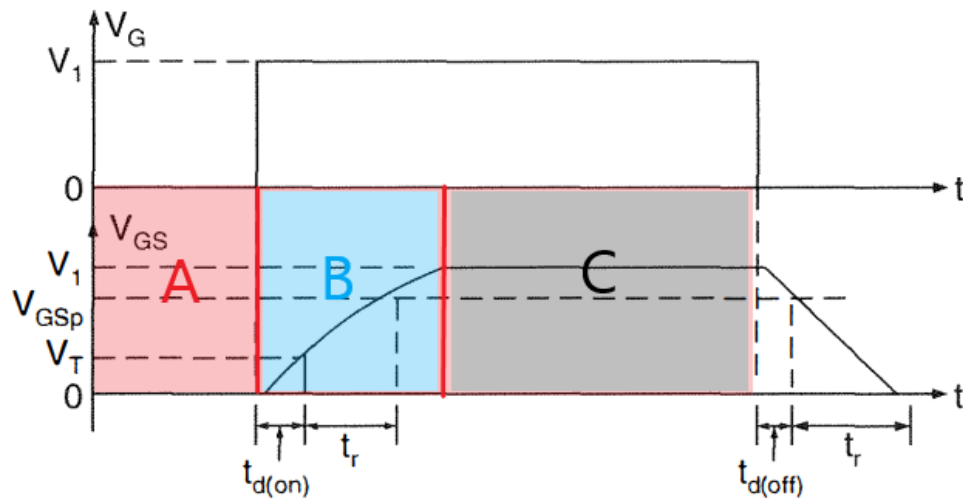
2.3.3 Chaves de potência

Como visto, alternativamente ao comutador mecânico do motor CC convencional, o *BLDC* necessita de dispositivos eletrônicos para realizar a comutação de suas bobinas. Dentre os dispositivos utilizados para tal fim tem-se o transistor bipolar de porta isolada (do inglês insulated gate bipolar transistor – IGBT) e o MOSFET. RASHID (2014) caracteriza tanto o IGBT quanto o MOSFET como dispositivos controlados por tensão, o que resulta na necessidade de baixas correntes de polarização. Além disso, ambos os dispositivos possuem baixas perdas tanto por chaveamento quanto por condução. Entretanto, segundo o autor, embora semelhantes, o MOSFET possui uma velocidade de chaveamento superior ao IGBT, o que torna seu uso mais adequado para operações em alta frequência.

2.3.3.1 Mosfet

Segundo RASHID (2014) é possível operar o MOSFET em três regiões: região de corte, região linear e região de saturação. Para atuar como chave eletrônica, deseja-se que o MOSFET seja operado somente na região de corte, onde o mesmo encontra-se fora de condução; e na região de saturação, onde este opera como uma chave fechada, possuindo baixas perdas. Entretanto, durante a troca de estado entre corte e saturação, devido à suas características internas, o MOSFET passa pela região linear, o que causa o maior aquecimento e perdas na chave. Na Figura 19 é possível observar as curvas características no acionamento do MOSFET, passando da região de corte (A), através da região ativa (B) para a região de saturação (C).

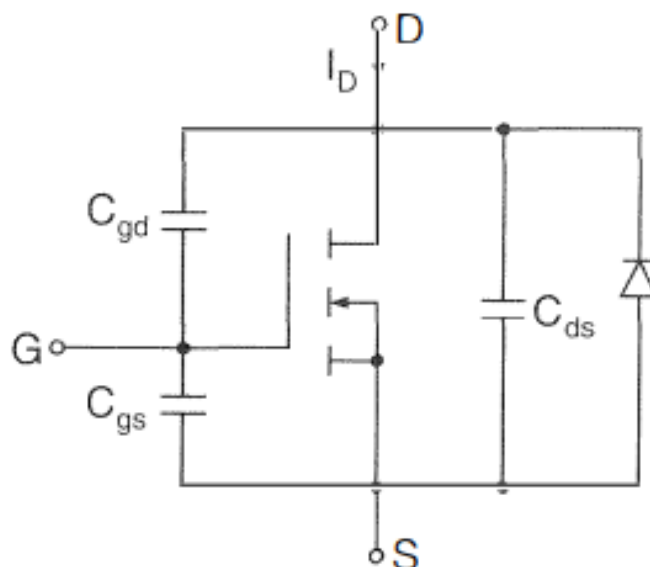
Figura 19: Curva de acionamento do MOSFET



Fonte: Adaptado de RASHID (2014)

Uma característica importante do MOSFET não ideal é a presença de capacitâncias parasitas entre o terminal de comporta (*gate*) e o terminal de dreno (*drain*); e entre o terminal de comporta e o terminal de fonte (*source*). É justamente devido a essas capacitâncias que o MOSFET possui um retardo na entrada em condução e um tempo de atraso no desligamento RASHID (2014). A representação dessas capacitâncias no modelo equivalente do MOSFET podem ser observadas na Figura 20.

Figura 20: Representação das capacitâncias internas do MOSFET



Fonte: (RASHID, 2014)

Segundo RASHID (2014), para garantir altas velocidades de chaveamento, o circuito utili-

zado para acionamento da comporta deve possuir uma baixa resistência série, o que implica em correntes instantâneas elevadas para polarização do MOSFET.

2.3.3.2 Perdas no Mosfet

Para MORITZ (2014), um dos fatores limitantes na condução de corrente elétrica em dispositivos semicondutores é a máxima temperatura de operação. Esta temperatura deve estar dentro dos valores estipulados pelo fabricante do componente. A operação fora dessas especificações pode comprometer o funcionamento do componente. No caso do MOSFET, quando este se encontra na região de corte, não há corrente circulando entre seus terminais, fazendo com que as perdas sejam desprezíveis. Entretanto, nas regiões linear e de saturação, devido à condução de corrente entre seus terminais, ocorrem perdas significativas. As perdas em qualquer dispositivo semicondutor podem ser divididas em perdas por condução e perdas por comutação, sendo que a última é o resultante da entrada e saída de condução (MORITZ, 2014).

Segundo MORITZ (2014), no MOSFET, as perdas por condução P_{con} levam em consideração a resistência equivalente entre o dreno e fonte quando o componente está em condução (R_{DSOn}) e a corrente que circula entre os terminais de potência. A Equação 2.17 sintetiza as perdas por condução no MOSFET.

$$P_{Con} = R_{DSOn} \cdot I_{rms}^2 \quad (2.17)$$

Como para a maioria dos MOSFETs o parâmetro R_{DSOn} é da ordem de $m\Omega$, as perdas por condução são relativamente pequenas quando comparadas com as potências que circulam pelas chaves. Ainda, segundo BARBI (2001), quando uma modulação de pulso é aplicada no acionamento do MOSFET, a Equação 2.17 pode ser reescrita como visto na Equação 2.18, onde $\frac{T_{On}}{T}$ representa a razão de trabalho e I_{don} é igual a corrente de dreno.

$$P_{Con} = \frac{T_{On}}{T} \cdot R_{DSOn} \cdot I_{don}^2 \quad (2.18)$$

As perdas por comutação representam a maior parte das perdas em um MOSFET. As capacitâncias internas do MOSFET resultam em duas características importantes: o tempo para entrar em condução (T_{On}) e o tempo para sair de condução (T_{Off}). Para BARBI (2001), as perdas resultantes da comutação do MOSFET podem ser calculadas utilizando a Equação 2.19.

$$P_{Com} = \frac{f}{2} \cdot (t_{On} + t_{Off}) \cdot i_{don} \cdot V_{dsOff} \quad (2.19)$$

Onde:

- P_{Com} - Perdas por comutação;
- V_{dsOn} - Tensão entre dreno e fonte quando o mosfet está em condução;

Ao final as perdas no MOSFET (P_{MOS}) são dadas pela soma entre as perdas por condução e por chaveamento, resultando na Equação 2.20.

$$P_{MOS} = P_{Con} + P_{Com} \quad (2.20)$$

Com as perdas nas chaves, é possível determinar sua temperatura de operação e, com isso, especificar o radiador de calor, caso necessário, a ser utilizado para garantir que os componentes operem dentro das faixas de temperatura seguras.

2.4 CONTROLE DO MOTOR

Para o correto funcionamento do *BLDC*, os sinais de controle devem estar sincronizados com a posição atual do rotor (BARATIERI, 2011). A detecção de posição do rotor pode ser feita com a utilização de sensores, e tal método é conhecido como *direct position-detection* (detecção direta de posição) (XIA, 2012). Entretanto, a utilização destes sensores encontram diversas complicações, principalmente quando utilizados em compressores de refrigeração (IZUKA K; UZUHASHI, 1985).

Como a amplitude da *back emf* gerada pelo motor é diretamente proporcional à velocidade do rotor, a estimação da posição através da detecção da *back emf* é possível somente acima de uma determinada velocidade (OGASAWARA; AKAGI, 1991). Assim, faz-se necessário um procedimento de partida em malha aberta, para que o rotor acelere até uma velocidade onde a tensão de *back emf* seja suficientemente alta.

2.4.1 Determinação da posição inicial

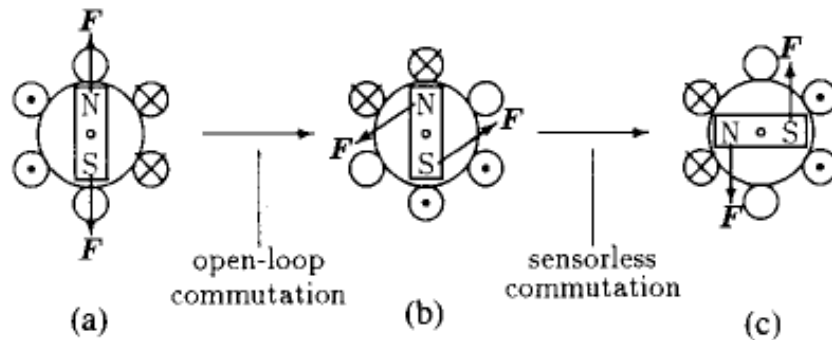
O acionamento das fases do motor *BLDC* se dá de forma sequencial, seguindo um determinado padrão de comutação. Caso essa sequência de acionamentos seja alterada, o rotor pode girar em sentido reverso, o que pode ser indesejado em algumas aplicações. Assim, o primeiro passo para garantir que o motor rotacione na direção correta, é determinar a posição magnética inicial do rotor, a partir da qual a sequência de acionamento seguirá.

2.4.1.1 Posicionamento forçado

Um dos métodos possíveis para a determinação do posicionamento do rotor no momento da partida, sugerido por OGASAWARA; AKAGI (1991), consiste na energização de duas fases do motor por um determinado tempo, fazendo com que o rotor se alinhe na direção da fase energizada. Na Figura 21(a) é mostrado o rotor em uma posição arbitrária qualquer. Quando duas bobinas são energizadas, há a produção de um campo magnético pelas bobinas da armadura, criando uma força magnética, interagindo com o campo magnético do rotor, como visto na Figura 21(b). Assim, o rotor gira de tal forma a fazer com que os campos se alinhem, como

pode ser observado na Figura 21(c).

Figura 21: Processo de alinhamento do rotor



Fonte: (OGASAWARA; AKAGI, 1991)

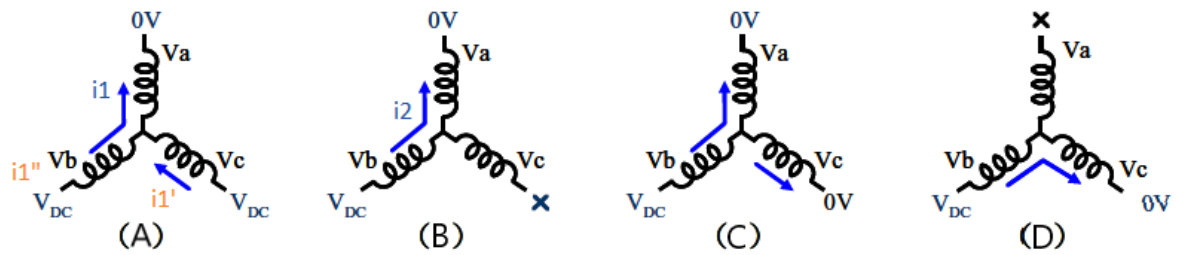
Sendo as bobinas energizadas conhecidas, é possível determinar a posição que o rotor assumiu, após o procedimento, podendo assim iniciar o processo de aceleração do rotor. Este método, entretanto, pode fazer com que o rotor, durante o alinhamento, gire na direção contrária à de rotação normal do sistema, o que pode ser indesejado em algumas aplicações. Além disso, em sistemas onde o torque de partida seja elevado, a aplicação deste método pode ser comprometida.

2.4.1.2 Estimador de posição por corrente

Outro método, sugerido por CHANG YEN-CHUAN; TZOU (2007), consiste na estimação da posição inicial através da leitura de corrente do barramento para cada uma das fases. Quando os enrolamentos do rotor encontram-se alinhados com os polos norte ou sul do estator, a indutância é reduzida devido ao efeito de saturação (CHANG YEN-CHUAN; TZOU, 2007). Assim, com diferentes valores de indutância, é possível determinar a posição magnética do rotor, através da leitura da corrente em diferentes acionamentos.

Para tal método, são utilizados quatro modos de acionamento das fases para a medição de corrente durante a detecção de posição inicial. O primeiro modo, visto na Figura 22(a) consiste na energização simultânea das três fases do motor, sendo duas delas ligadas ao +VCC do barramento através de suas respectivas chaves, e a outra ligada ao GND do barramento. No segundo modo, visto na Figura 22(b) é mantida a configuração da primeira, apenas desconectando uma das fases que antes estava ligada à +VCC, deixando-a em aberto. No terceiro, a fase que foi aberta no segundo modo, é então conectada ao GND, conforme Figura 22(c). No último modo de acionamento, a fase que foi mantida à GND desde o início é então desconectada, conforme Figura 22(d). Para todos os quatro modos de acionamento, a corrente do barramento é medida e salva para a posterior comparação com as demais. A utilização desta sequência resulta numa resolução de 30° elétricos para a determinação da posição.

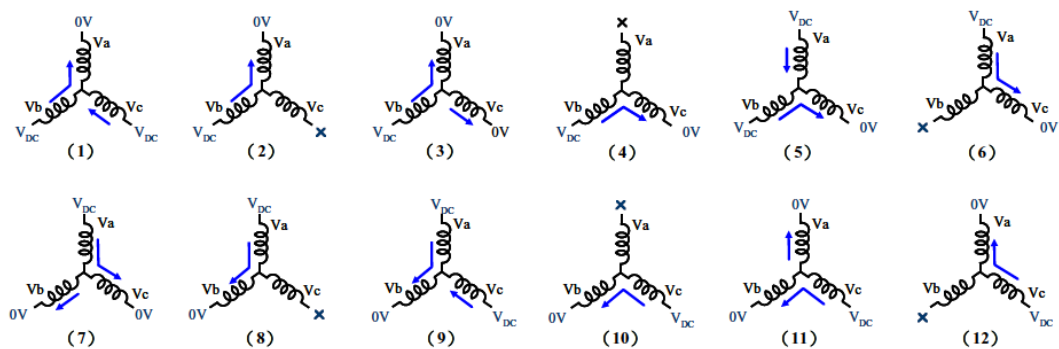
Figura 22: Medição de corrente para detecção da posição inicial



Fonte: Adaptado de (CHANG YEN-CHUAN; TZOU, 2007)

Aplicando os quatro modos de acionamento para as três fases, são obtidos doze acionamentos, como pode ser visto na Figura 23. Nos modos de acionamento onde duas fases encontram-se energizadas ao mesmo tempo (modo um e três), a corrente é calculada pela Equação 2.21. No modo onde uma das fases encontra-se desconectado (modo dois e quatro), a corrente é calculada pela Equação 2.22.

Figura 23: Aplicação das quatro formas de acionamento para as 3 fases do motor



Fonte: Adaptado de (CHANG YEN-CHUAN; TZOU, 2007)

$$i_1(t) = \frac{+V_{cc}}{1,5R} (1 - e^{-\frac{R}{L}t_1}) \quad (2.21)$$

$$i_2(t) = \frac{+V_{cc}}{2R} (1 - e^{-\frac{R}{L}t_2}) \quad (2.22)$$

De modo a linearizar o valor de corrente para os dois casos, o tempo de acionamento das fases para os modos dois e quatro é diferente dos modos um e três. Assim, o tempo dos modos dois e quatro é expresso pela Equação 2.23.

$$t_2 = -\frac{R}{L} \ln\left(\frac{4e^{-\frac{R}{L}t_1} - 1}{3}\right) \quad (2.23)$$

Comparando os valores de corrente obtidos nos doze acionamentos, tem-se que o rotor está

posicionado na direção onde o maior valor de corrente for obtido. Tal método de estimação de posição traz por principais vantagens o fato de não gerar nenhuma rotação durante o processo de estimação e, como todos os valores de corrente são comparados entre si, o método é robusto quanto à variação de parâmetros do motor e, embora nas Figuras 22 e 23 seja representado uma ligação Y , tal método também pode ser utilizado caso o motor esteja ligado em Δ , apenas alterando a resistência equivalente das Equações 2.21 e 2.22.

2.4.1.3 Procedimento de aceleração

Para um controle preciso de velocidade, a detecção da posição do rotor é necessária. Em um sistema *sensorless* (sem uso de sensores), a detecção da posição é realizada através da leitura da *BEMF* nas fases não energizadas do motor. Entretanto, essa leitura é possível somente a partir de uma determinada velocidade (OGASAWARA; AKAGI, 1991) (SHAO JIANWEN; NOLAN, 2002) (IIZUKA K; UZUHASHI, 1985). Assim, até atingir a velocidade mínima de detecção da *BEMF* o motor é acionado em malha aberta (sem realimentação do posicionamento).

Entretanto, o valor de corrente do sistema é detectável desde a comutação do primeiro conjunto de chaves. Assim, segundo MILLER (2010) o valor de corrente pode ser utilizado para determinar a velocidade aproximada do motor, conforme Equação 2.24.

$$\omega_b = \frac{V_{cc} - 2 \cdot R \cdot I_{cc}}{K'} \quad (2.24)$$

Apesar da velocidade poder ser estimada, para a aplicação prática de tal método, é importante que seja considerado um fator de erro, podendo ser determinado a partir de algumas amostras práticas do sistema. Além disso, a partir do momento que a *BEMF* seja detectável, IIZUKA K; UZUHASHI (1985) sugere que um certo número de detecções seja realizado, de modo prevenir que, na troca entre o controle de malha aberta para o controle de malha fechada, o sequenciamento de chaves seja realizado de forma inadequada, podendo haver a perda no controle da máquina.

2.5 CIRCUITO DE CONTROLE E PROTEÇÃO

2.5.1 Corrente

Como visto anteriormente, o controle do motor depende diretamente da leitura de corrente das fases do motor. Para leitura da corrente de cada fase do inversor, pode-se empregar circuitos integrados dedicados, tais como o ACS723LLCTR da fabricante *Allegro Microsystems*. Alternativamente, devido a característica de condução das chaves de potência, MILLER (2010) afirma que a leitura de corrente das fases do motor pode ser realizada através da leitura de corrente do barramento CC. Assim, é possível utilizar um resistor de *shunt* para amostragem da corrente do barramento.

Dentre os dispositivos para proteção contra sobrecorrente, é possível empregar, em série com a entrada, o uso de termistores, como o PTC (*Positive Temperature Coefficient*), que tem sua resistência aumentada proporcionalmente à temperatura. Assim, um aumento anormal nos valores de corrente de entrada, faz com que a resistência do PTC se eleve de modo que, haja a desconexão total ou parcial entre a fonte de tensão e o barramento CC.

Além disso, é possível empregar fusíveis para a proteção do circuito. Fusíveis são componentes utilizados em série com um circuito, de modo a proteger o sistema contra curto-circuito. Em eletrônica embarcada, o modelo mais utilizado é o chamado fusível de vidro, que consiste em um filamento inserido dentro de um tubo de vidro onde, devido às características físicas deste filamento, caso a corrente exceda os valores nominais do mesmo por um determinado tempo, ocorre seu rompimento. Um fusível de vidro, empregado em um circuito eletrônico pode ser visto na Figura 24.

Figura 24: Fusível de vidro em um circuito eletrônico



Fonte: Autor (2023)

Alternativamente aos fusíveis convencionais, é possível aplicar técnicas de desenvolvimento de PCI (Placa de Circuito Impresso) para criar partes intencionalmente frágeis em um circuito eletrônico, atuando de forma semelhante aos fusíveis de vidro. Tal método é conhecido como

PCB fuse (Fusível de PCI) e, embora possua um grau elevado de imprecisão, tem as vantagens de ocupar uma área pequena e possuir baixo custo (DIMITROV BORISLAV; CRUDEN, 2018). O princípio de funcionamento de tal técnica baseia-se na característica de aquecimento do material condutor das trilhas quando percorrida por uma corrente elétrica (BROOKS DOUGLAS; ADAM, 2015).

O projeto da trilha fusível deve ser tal que, para as correntes normais de operação, esta comporte-se como uma trilha comum, entrando em operação, ou seja rompendo-se, somente quando surgir uma corrente de pelo menos três vezes a corrente nominal por um determinado período de tempo (DIMITROV BORISLAV; CRUDEN, 2018). Segundo DIMITROV BORISLAV; CRUDEN (2018) a corrente máxima I_t para uma determinada trilha de cobre é dada pela Equação 2.25.

$$I_t = 0,065 \cdot \Delta T^{0,43} \cdot A^{0,68} \quad (2.25)$$

Onde:

- ΔT - Diferença entre a temperatura ambiente e a temperatura da trilha;
- A - Área total da trilha;

Assumindo a temperatura ambiente como 40 °C e a temperatura de fusão do cobre como 1083 °C (BROOKS, 1998), é obtido o valor de ΔT conforme Equação 2.26.

$$\Delta T = 1083^{\circ}C - 40^{\circ}C = 1043^{\circ}C \quad (2.26)$$

Para determinar a área de uma trilha fusível, pode-se aplicar a Equação 2.26 à Equação 2.25, obtendo a Equação 2.27.

$$A = \frac{I_t}{0,065 \cdot 1043^{0,43}} = \frac{I_t}{1,29} \quad (2.27)$$

Quanto ao formato, DIMITROV BORISLAV; CRUDEN (2018) aponta que a menor dispersão no valor de corrente ocorre com trilhas fusíveis onde há uma concentração de calor em determinados pontos; para tanto, o autor sugere o uso de trilhas em zigue-zague, conforme visto na Figura 25.

Figura 25: Trilha fusível em zigue-zague



Fonte: DIMITROV BORISLAV; CRUDEN (2018)

2.5.2 Sobretensão

Em sistemas automotivos, instantes após a partida do motor, o alternador inicia sua operação, gerando energia para o carregamento da bateria. É precisamente neste instante que ocorre um surto de tensão, conhecido como *load-dump*, que faz com que a tensão no sistema suba além dos níveis normais de operação por um pequeno, embora significativo, período de tempo.

Existem dois métodos principais para a proteção de sistemas eletrônicos automotivos contra surtos de tensão. O primeiro, menos energeticamente eficiente, consiste na dissipação da energia excedente, seja na utilização de componentes ativos (transistores) ou passivos (resistores). Entretanto, a energia na entrada do circuito pode ser suficientemente elevada, de tal modo que ocorra dano aos fusíveis ou demais componentes do circuito por excesso de corrente.

O segundo método baseia-se na desconexão momentânea entre a entrada de tensão e o circuito eletrônico, fazendo com que, ao ser detectado um nível de tensão superior ao desejado, o circuito seja brevemente interrompido, protegendo os componentes sensíveis ao nível de sobretensão.

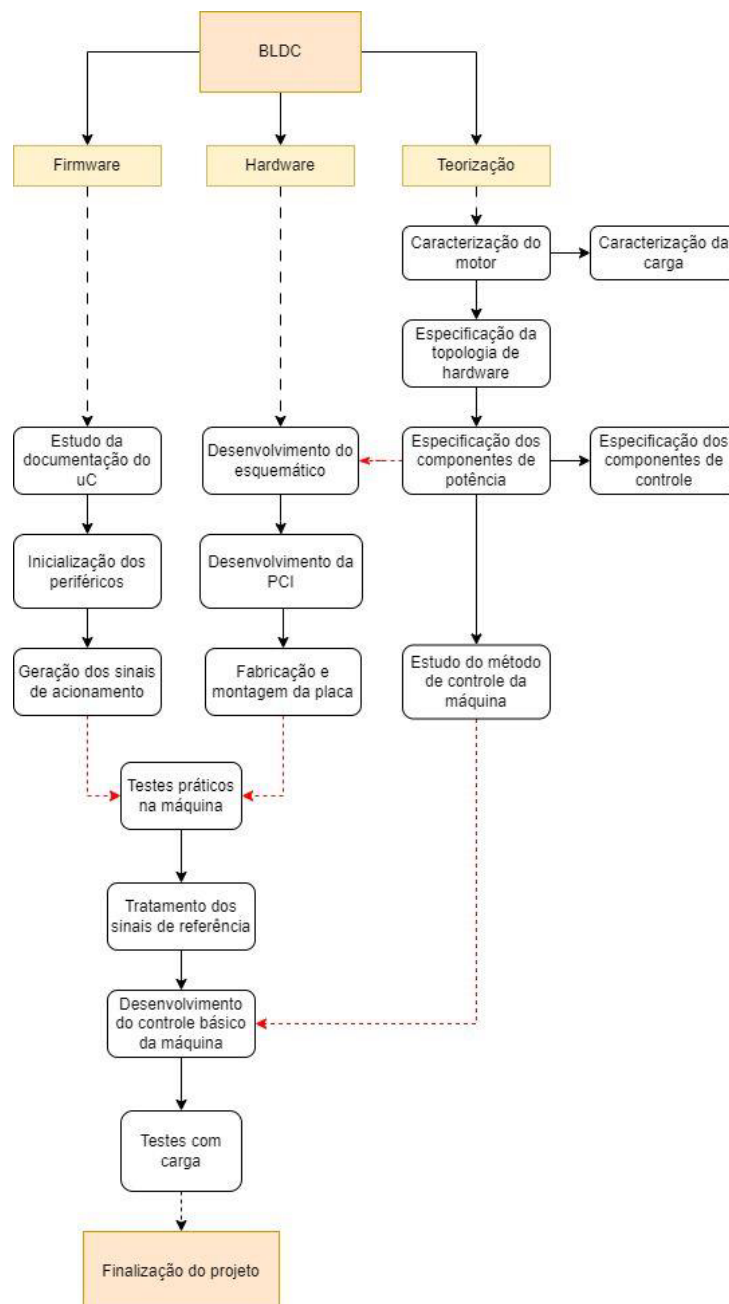
2.5.3 Subtensão

Estando as baterias veiculares sujeitas a diversos ciclos de carga e descarga, os níveis de tensão na alimentação do circuito devem ser constantemente monitorados para que, caso detecte-se um valor de tensão inferior aos valores nominais, o circuito de potência seja desligado, prevenindo que o circuito opere a níveis críticos. Além disso, por operar por longos períodos de tempo ininterruptamente, o sistema pode causar o descarregamento da bateria do veículo, podendo causar falhas operacionais críticas, como a falta de energia suficiente para dar partida no motor.

3 MÉTODO

Este capítulo apresenta o desenvolvimento prático do projeto. As etapas executadas seguiram conforme Figura 26.

Figura 26: Fluxograma de desenvolvimento do projeto



3.1 DESENVOLVIMENTO DO HARDWARE

Conforme consta nos objetivos do projeto, o sistema proposto foi implementado na prática. Com isso, foi desenvolvido uma placa de circuito impresso, utilizando o *software Altium Designer*. O dimensionamento dos componentes e projeto dos circuitos, além das técnicas pertinentes de desenvolvimento de layout são apresentados neste capítulo.

Sabendo que, do ponto de vista da indústria, o custo de um determinado componente é afetado pela quantidade demandada, ou seja, para um mesmo componente, ao comprar uma quantidade significativamente maior, consegue-se um custo inferior por componente, quando comparado com o cenário de uma compra em menor quantidade. Desta forma, por vezes, a utilização de um componente com especificações que vão além das requeridas pelo projeto, mas com um alto consumo para a empresa, pode possuir um custo inferior do que um componente com especificações inferiores, mas com um consumo menor.

3.1.1 Circuito de potência

A etapa de potência do projeto contempla tanto o dimensionamento das chaves de potência em si, quanto do circuito dedicado ao acionamento das mesmas. Como visto anteriormente, pode-se utilizar tanto MOSFETs quanto IGBTs como chaves de potência. Neste projeto, optou-se pela utilização de MOSFETs, atendendo aos requisitos especificados no início do projeto.

3.1.1.1 Chaves de potência

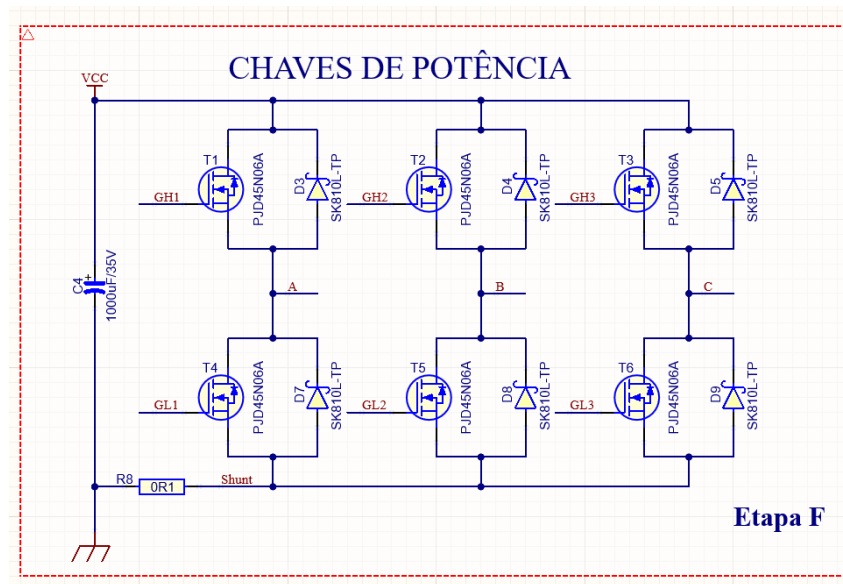
Para a escolha das chaves, deve-se, primeiro, determinar os parâmetros máximos aos quais estas serão submetidas. Como dito anteriormente, o sistema é alimentado por uma bateria de 24 V nominais. Entretanto, por se tratar de um sistema automotivo, a tensão de entrada pode ser ligeiramente superior ao valor nominal, devido ao sistema de carregamento (alternador). Além disso, com a implementação do circuito de proteção contra *load-dump*, o circuito tem sua alimentação limitada à 34 V. Para tanto, de modo a não superdimensionar as chaves, considerou-se uma tensão de operação 10 % maior que a tensão nominal, ou seja 26,4 V, com distúrbios de até 34 V.

Outro parâmetro importante para a escolha das chaves é a corrente. Os testes preliminares com o motor, utilizando um drive genérico, mostrou que a corrente nominal na saída é de 3 A. Além disso, foi observado picos de corrente de até 6 A durante o procedimento de partida do motor.

Com base nesses parâmetros e na relação de custo discutida anteriormente, o MOSFET escolhido para o projeto foi o PJD45N06A do fabricante Panjit. Segundo o fabricante, este componente, ao ser utilizado com algum método de dissipação, a 100 °C, suporta uma corrente de 29 A e uma tensão de 60 V (entre o terminal de dreno e fonte). O circuito implementado

com as seis chaves de potência e o *shunt* para amostragem de corrente pode ser visto na Figura 27.

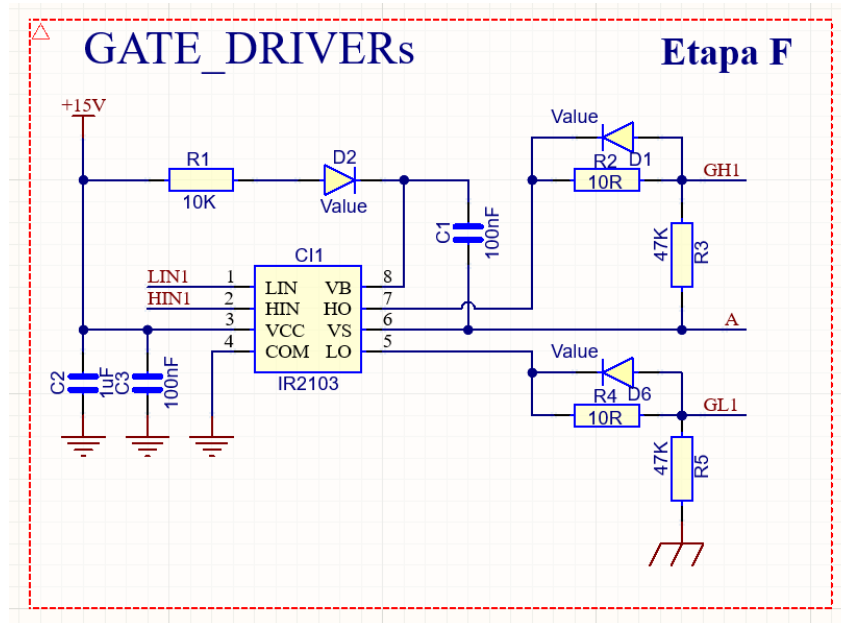
Figura 27: Chaves de potência



Fonte: Autor (2023)

3.1.1.2 Gate drivers

Para o acionamento das chaves foi utilizado um circuito integrado dedicado a esse fim, conhecido como *gate driver*. O componente escolhido foi o IR2103. A implementação dos circuitos periféricos seguiu as recomendações do fabricante e o resultado pode ser observado na Figura 28.

Figura 28: Circuito dos *gate drivers*

Fonte: Autor (2023)

Para garantir o correto valor de V_{GS} , o circuito implementa um elevador de tensão, conhecido como circuito de *bootstrap*. Na Figura 28, C1 e D2 são utilizados para tal função. Entretanto, para o correto funcionamento do circuito de *bootstrap*, é necessário que um sinal PWM seja aplicado à chave de high-side.

Além disso, há a presença de um resistor em série com o terminal de comporta (*Gate*) do MOSFET, utilizado para limitar a corrente de acionamento I_G . Em anti-paralelo com ele é previsto a possibilidade de montagem de um diodo (D1), utilizado para reduzir o tempo de desligamento da chave. Assim, seria possível reduzir as perdas por desligamento da chave.

Como pode ser visto na Figura 28, um único gate driver consegue acionar dois MOSFETs, sendo estes, geralmente, ligados na mesma fase de saída, formando um dos três braços do inversor. Logo, são necessários três circuitos idênticos para o acionamento das seis chaves do inversor.

3.1.2 Circuitos de controle

Os circuitos de controle compreendem, basicamente, as etapas do circuito onde sinais são amostrados ou são gerados, sendo caracterizados pelo baixo consumo energético.

3.1.2.1 Leitura de corrente

Como visto anteriormente, a leitura da corrente do circuito de potência é dada pela leitura da corrente do barramento. Assim, utilizou-se um resistor SMD de $0,01 \Omega$ entre o GND e o

ponto comum das chaves de potência, formando um *shunt*. Com base nisso, a potência nominal dissipada pelo resistor é calculada na Equação 3.1.

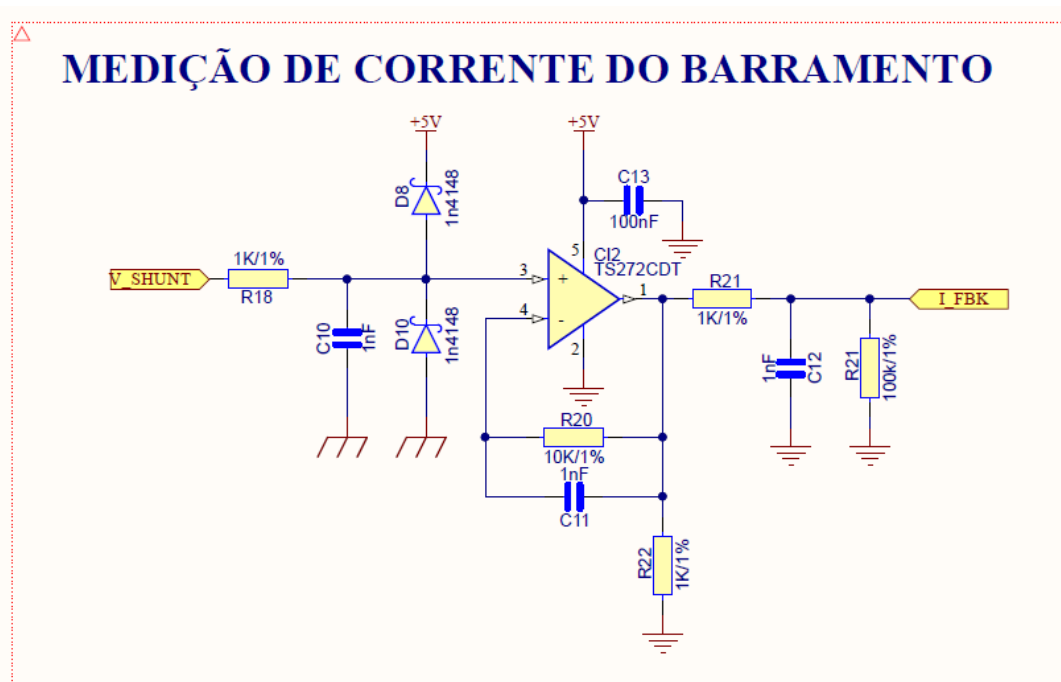
$$\begin{aligned} P_S &= I^2 \cdot R_S \\ P_S &= 3^2 \cdot 0,01 \\ P_S &= 90mW \end{aligned} \tag{3.1}$$

Com uma corrente nominal de 3 A, a tensão esperada no *shunt* é calculada pela Equação 3.2.

$$\begin{aligned} V_S &= I \cdot R_S \\ V_S &= 3 \cdot 0,01 \\ V_S &= 30mV \end{aligned} \tag{3.2}$$

Além de adequar o valor de tensão no resistor de *shunt*, o emprego de um amplificador operacional no circuito de leitura de corrente traz mais dois benefícios: por possuir uma característica de alta impedância de entrada, ele auxilia no desacoplamento entre o circuito de potência e o circuito de controle, o que reduz a propagação de ruído. Além disso, garante a reusabilidade do projeto para sistemas onde os níveis de corrente sejam diferentes dos especificados nesse projeto. Assim, embora no contexto específico deste projeto o emprego de um amplificador operacional traga um acréscimo no custo total, num contexto geral, por poder ser utilizado com outros modelos de motores, é possível manter uma única PCI para diversas aplicações. Isso acarreta em um custo inferior ao de se utilizar diversas PCIs dedicadas à cada projeto. O circuito implementado pode ser visto na Figura 29, onde a etiqueta “VSHUNT” representa a ligação com o resistor de *shunt*.

Figura 29: Circuito de leitura de corrente



Fonte: Autor (2023)

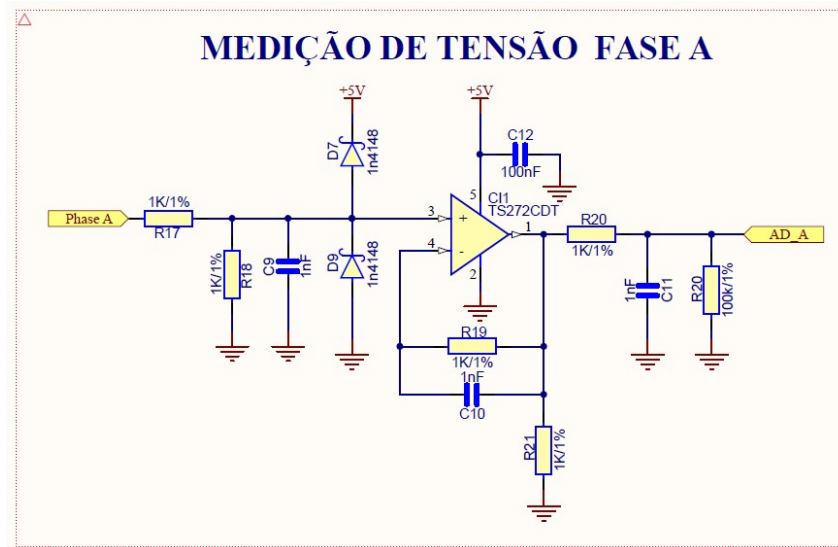
Além disso, para proteger a entrada do circuito, são adicionados dois diodos de *clamp*. Estes diodos, caso o sinal ultrapasse a tensão máxima ou mínima da entrada do AMP-OP, os mesmos entram em condução, protegendo o componente. De modo a adequar os valores lidos para a faixa de tensão do microcontrolador, há um ganho de dez vezes nos dois estágios de amplificação, resultando num ganho total 100.

3.1.2.2 Leitura de BEMF

Para o tratamento da BEMF, optou-se pela leitura do sinal de forma analógica, utilizando um amplificador operacional para cada fase do motor. Tal metodologia apresenta maior flexibilidade no controle, quando comparado com a técnica de leitura apenas do cruzamento por zero do sinal. Desta forma, é possível mensurar o valor atual da BEMF em cada instante, de forma separada, e através de algoritmos implementados pelo *firmware*, é possível estimar um comportamento dinâmico do circuito.

Semelhantemente à leitura de corrente, um filtro passa-baixa foi adicionado na entrada do circuito, apenas sintonizado para uma frequência diferente, o que pode ser visto pela relação entre R28 e C18. Isso se deve à diferente resposta em frequência do circuito durante o procedimento de aceleração. O circuito de leitura da BEMF, de uma das fases, utilizado no projeto, pode ser visto na Figura 30.

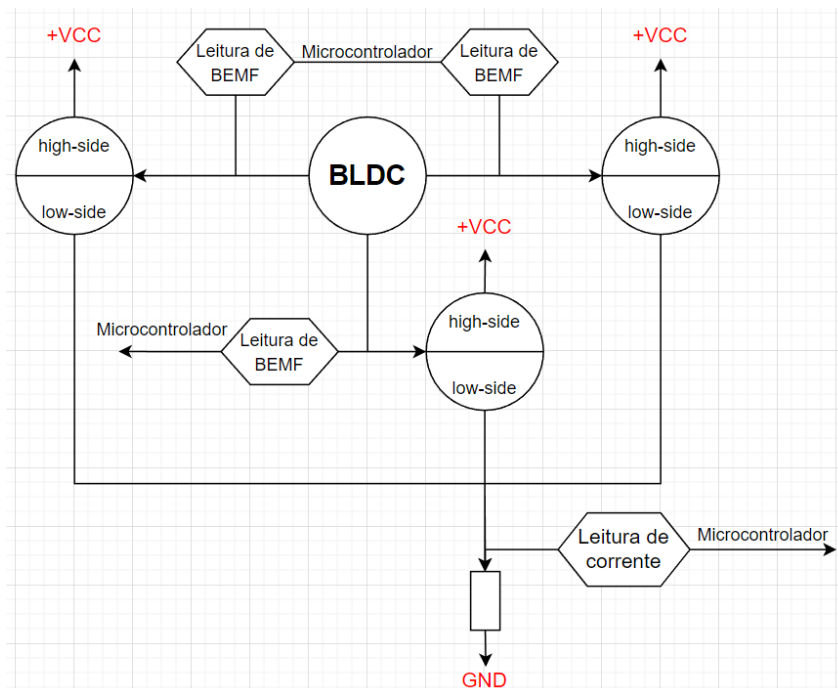
Figura 30: Circuito de leitura da BEMF



Fonte: Autor (2023)

A representação dos circuitos de controle nos três terminais do motor pode ser vista na Figura 31.

Figura 31: Representação simplificada do sistema de controle



Fonte: Autor (2023)

3.1.2.3 Microcontrolador

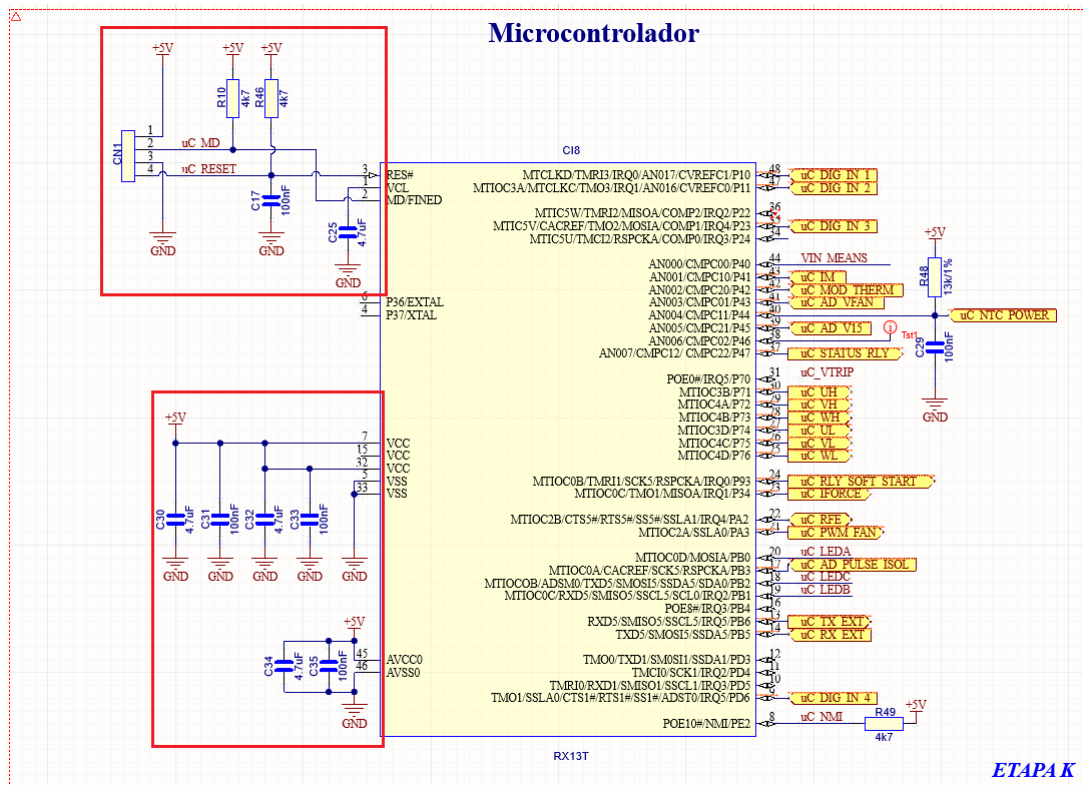
Sendo o principal componente do circuito de controle, o microcontrolador deve possuir, algumas características, tais como:

- 4 canais de ADC (Conversor Analógico/Digital);
- 6 saídas PWM;
- 2 *Timers*

Desta forma, atendendo tanto aos requisitos mínimos listados acima quanto à requisitos comerciais, o microcontrolador escolhido para o projeto foi o RX13Txxx da família Rx da Renesas.

Estando sujeito à diversas formas de ruído, para a correta operação do componente, o fabricante sugere a utilização de alguns componentes, de valor e posição específica, ligados a determinados pinos. A escolha dos componentes periféricos do microcontrolador seguiu as recomendações do fabricante conforme o Application Note da família RX. Os componentes periféricos utilizados no circuito do microcontrolador podem ser observados em destaque na Figura 32.

Figura 32: Circuito do microcontrolador

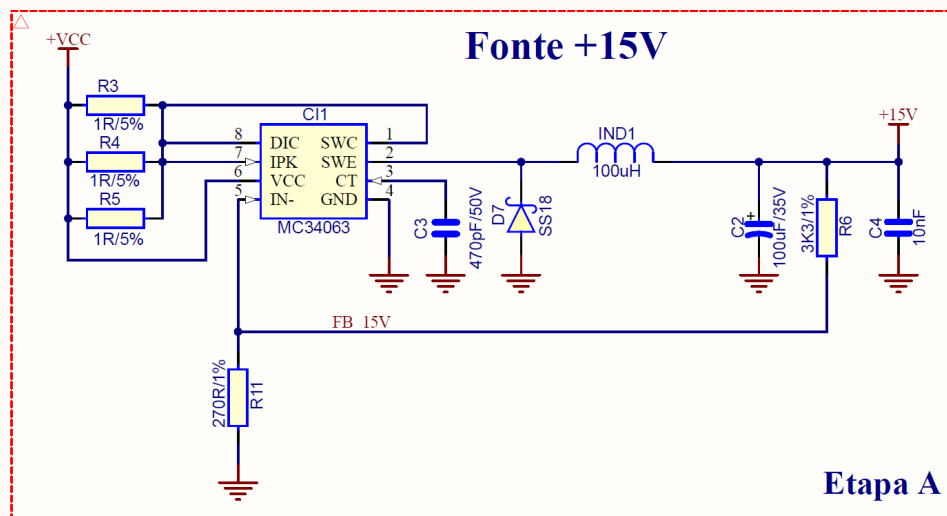


3.1.3 Fontes

Embora o circuito já seja alimentado por uma fonte de tensão (bateria), para o correto funcionamento de todos os circuitos embarcados, é necessário a adequação dos níveis de tensão de entrada para os níveis operacionais de cada circuito. As partes mais críticas, quanto ao nível de tensão, são: o microcontrolador, o circuito de acionamento de *gate* dos MOSFETs e os amplificadores operacionais. Sendo assim, a partir da fonte principal de 24 V, é necessário o desenvolvimento de uma fonte de 15 V e uma fonte de 5 V.

Para a fonte de 15 V, optou-se pela utilização de um conversor *buck*, através de um CI dedicado, alimentado diretamente pela tensão de entrada do circuito. O componente escolhido para tal é o MC34063 da fabricante Texas Instruments e o dimensionamento dos componentes periféricos, além de seu posicionamento ao longo da PCI, seguiu as recomendações do fabricante. O resultado do projeto desta fonte pode ser visto na Figura 33

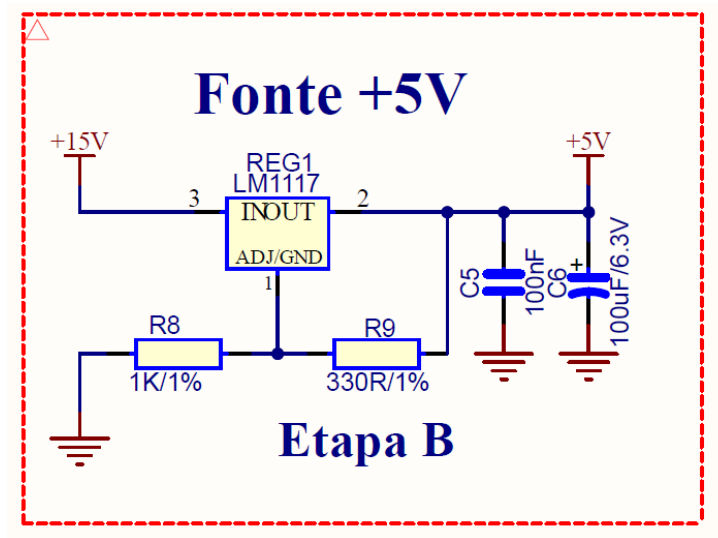
Figura 33: Circuito da fonte de +15 V



Fonte: Autor (2023)

Para a fonte de 5 V, por alimentar circuitos com baixo consumo, utilizou-se um regulador linear LM1117. Este tipo de regulador possui uma baixa eficiência, tendo suas perdas diretamente proporcionais à diferença entre tensão de entrada e tensão de saída. Com isso, a entrada de tensão do regulador foi ligada à saída do regulador de 15 V, ao invés da entrada geral do circuito. O circuito resultante para a fonte de 5 V pode ser observada na Figura 34

Figura 34: Circuito da fonte de +5 V



Fonte: Autor (2023)

3.1.4 Proteções

Visando proteger tanto o circuito de *drive* quanto os demais componentes do veículo, foram implementadas proteções por *hardware* e por *firmware*. As proteções implementadas em *firmware* serão discutidas nas seções seguintes. As proteções por *hardware* são, geralmente, associadas à falhas catastróficas, na qual o tempo de resposta e a criticidade da falha são determinantes para a redução nos danos aos componentes envolvidos.

3.1.4.1 Curto-circuito

Devido as características operacionais e de custo, para a proteção contra curto-circuitos, optou-se pela utilização de trilhas fusíveis. Assim, utilizando a Equação 2.27, a área da trilha fusível para proteger o circuito contra correntes superiores a 11 A é calculada na Equação 3.3.

$$A = \frac{11}{1,29} \approx 8,53\text{mm}^2 \quad (3.3)$$

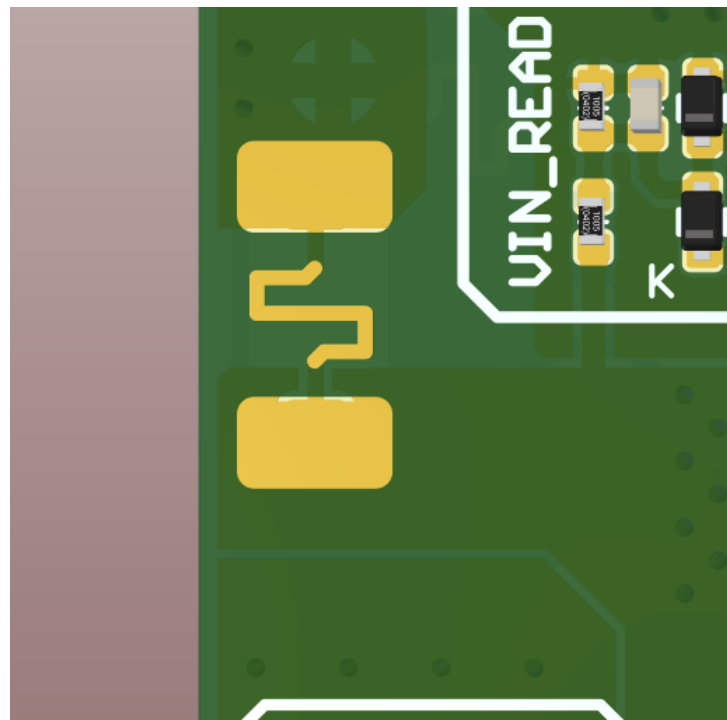
Ainda, sabendo que a camada de cobre depositada sobre o material base da PCI possui uma espessura de aproximadamente 29,57 mm, a largura da trilha fusível L_p é calculada na Equação 3.4.

$$L_p = \frac{A}{h} = \frac{8,53}{29,57} \approx 0,29\text{mm} \quad (3.4)$$

Com uma largura de aproximadamente 0,29 mm, a trilha fusível foi desenhada na PCI com

o formato aproximado ao descrito na seção 2.5.1, como pode ser visto na Figura 35.

Figura 35: Trilha fusível desenhada na PCI

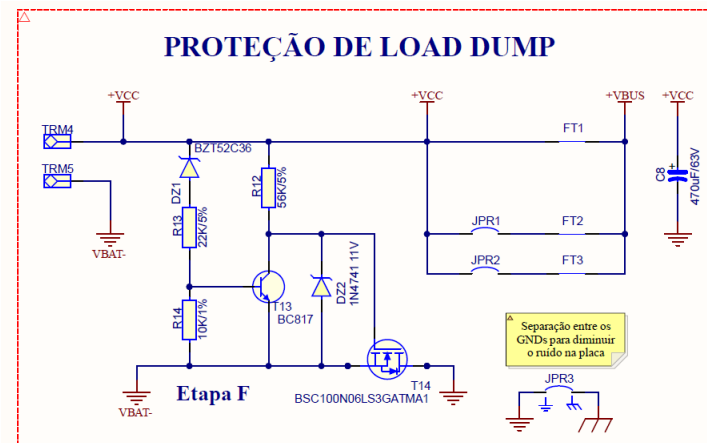


Fonte: Autor (2023)

Além disso, sabendo que a trilha fusível pode atingir elevadas temperaturas, o ponto de concentração tem a máscara de solda aberta, o que garante uma melhor operabilidade com menores danos ao material base da PCI.

3.1.4.2 *Load-Dump*

Para proteção tanto contra sobretensões quanto para picos de *load-dump*, optou-se pela técnica de interrupção da alimentação, conforme descrito na seção 2.5.2. O circuito implementado para tal função pode ser visto na Figura 36.

Figura 36: Circuito de proteção contra sobretensão e *load-dump*

Fonte: Autor (2023)

Neste circuito, DZ1 possui uma tensão reversa de 36 V. Assim, quando tensões superiores à este valor são observadas na entrada de alimentação do sistema, DZ1 entra em condução, fazendo com que T13 entre em saturação e, conseqüentemente fazendo T14 entrar na região de corte, eliminando a conexão entre o GND do circuito e o terminal negativo da bateria.

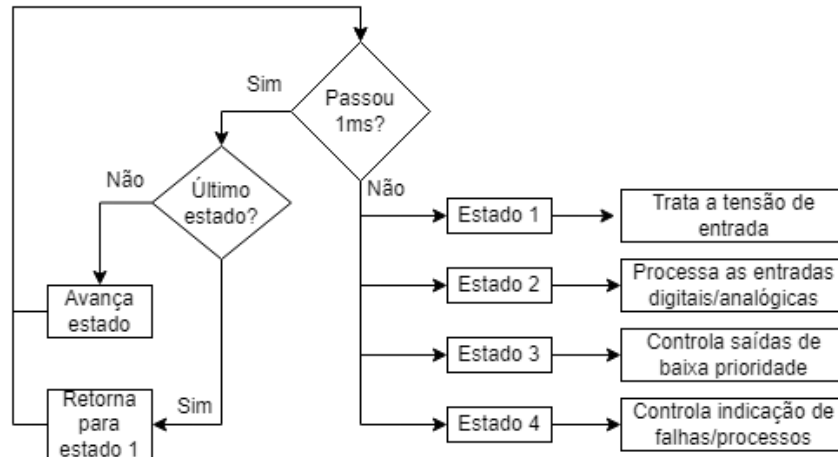
3.2 DESENVOLVIMENTO DO FIRMWARE

O microcontrolador é o componente responsável tanto pela geração dos sinais de controle e acionamento quanto pela leitura e processamento das entradas analógicas e digitais, tais como tensão de entrada, corrente no barramento, tensão de BEMF, entre outros. Assim, pode-se separar o processamento em duas partes: crítica e não crítica.

O processamento não crítico consiste no tratamento dos sinais onde não são requeridas grande capacidade de processamento e nem uma baixa latência. Já o processamento crítico está relacionado ao processamento dos sinais de controle do motor. Sendo assim, o processamento crítico precisa de um baixo tempo de resposta, caso contrário, pode levar ao acionamento indevido das chaves de potência podendo até mesmo levar à perda no controle da máquina.

Para o processamento não crítico, optou-se pela implementação de uma máquina de estados finita, onde os estados são trocados com no mínimo 1 ms de operação. A Figura 37 representa o funcionamento simplificado do processamento não crítico.

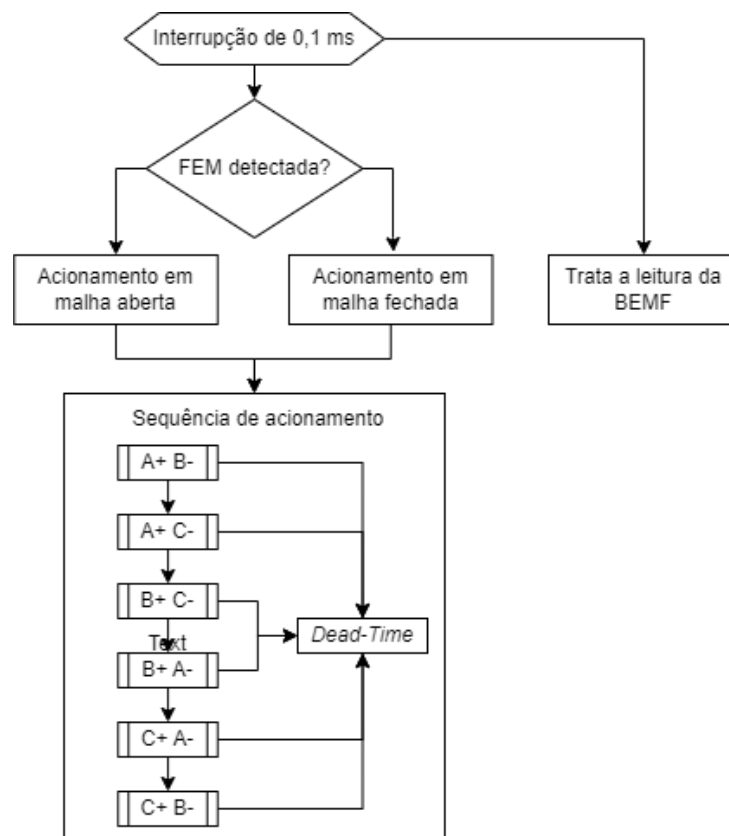
Figura 37: Diagrama simplificado do processamento não crítico



Fonte: Autor (2023)

De modo a garantir o correto controle da máquina, o processamento crítico está atrelado à interrupção de um *timer* de alta prioridade, operando à uma frequência de 10 KHz, o que assegura o tratamento dos sinais a cada 0,1 ms. A Figura 38 representa o diagrama em blocos simplificado do processamento crítico.

Figura 38: Diagrama simplificado do processamento crítico



Fonte: Autor (2023)

Ainda, para reduzir o número de instruções de máquina consumidos e, conseqüentemente o tempo necessário para o processamento dos sinais de controle, foram utilizadas macros com instruções de pré compilação. Instruções de pré compilação consistem em realizar cálculos para o controle em tempo de compilação, ao invés de execução. Assim, embora a compilação do código tenha um tempo maior, o código de máquina gerado é menor e o tempo de processamento por parte da *CPU* do microcontrolador é reduzido, comparado com a realização dos mesmos cálculos em tempo de execução.

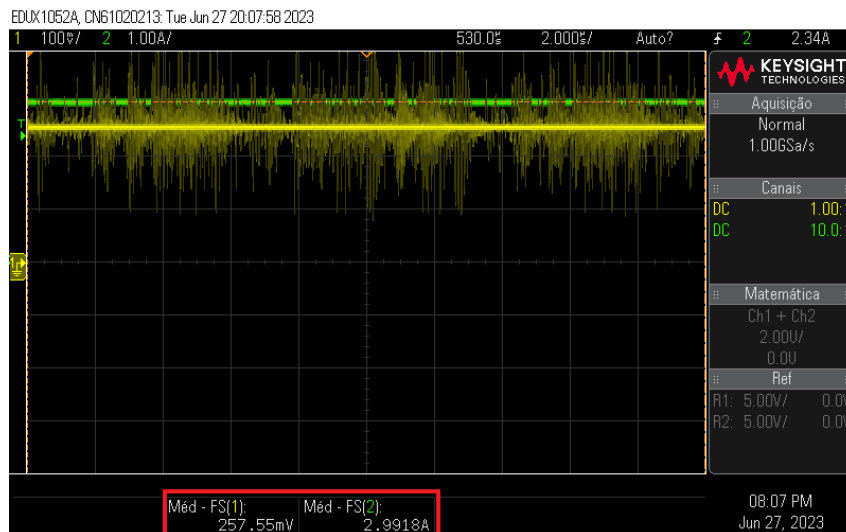
4 RESULTADOS

Para comprovar o desenvolvimento do projeto, seguindo a fundamentação teórica apresentada nos capítulos anteriores, este capítulo apresenta os resultados práticos obtidos com o projeto.

4.1 PARÂMETROS DO MOTOR

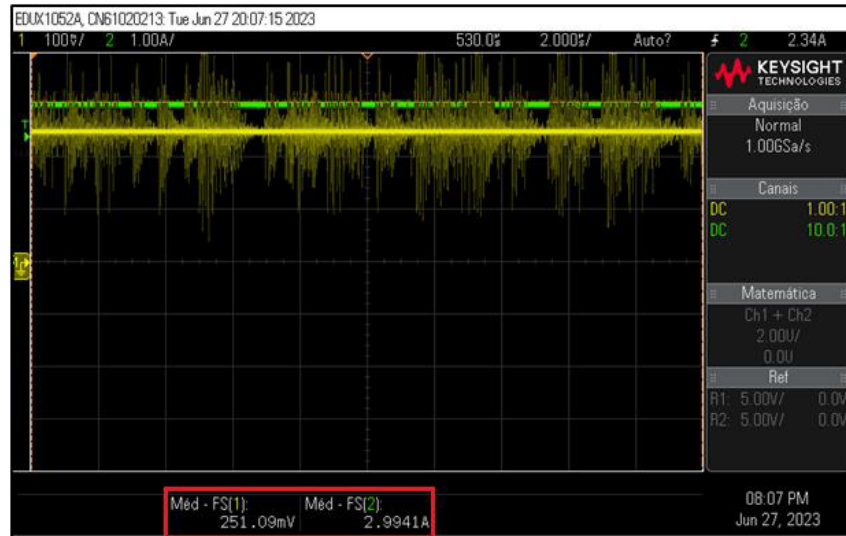
Para determinar a resistência dos enrolamentos de armadura do motor utilizado, foi aplicado o método descrito na seção 2.2.1. Segundo o fabricante do motor, a corrente nominal, em 24 V, é de 3 A. O valor de tensão aplicado às bobinas do motor, de modo a se obter esses 3 A, foi de aproximadamente 268 mV, como pode ser visto nas Figuras 39, 40 e 41.

Figura 39: Resultado obtido do ensaio de resistência entre as fases A e C



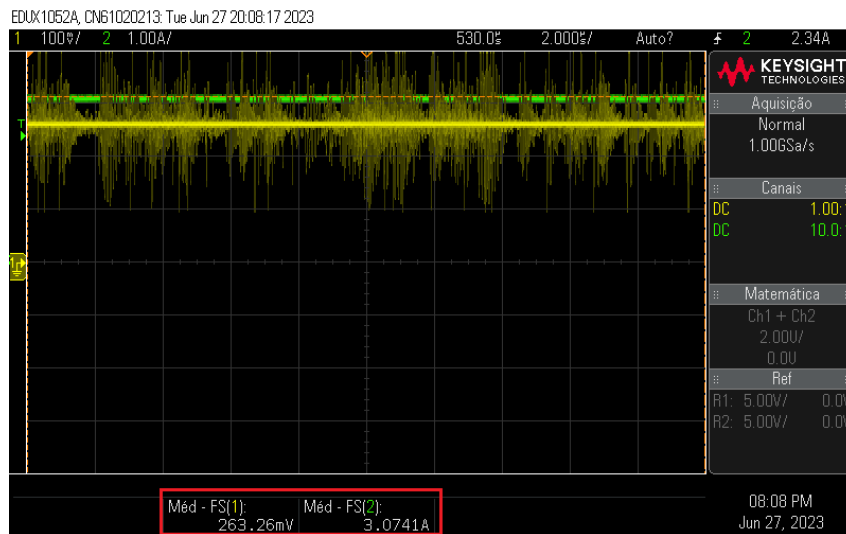
Fonte: Autor (2023)

Figura 40: Resultado obtido do ensaio de resistência entre as fases A e B



Fonte: Autor (2023)

Figura 41: Resultado obtido do ensaio de resistência entre as fases B e C



Fonte: Autor (2023)

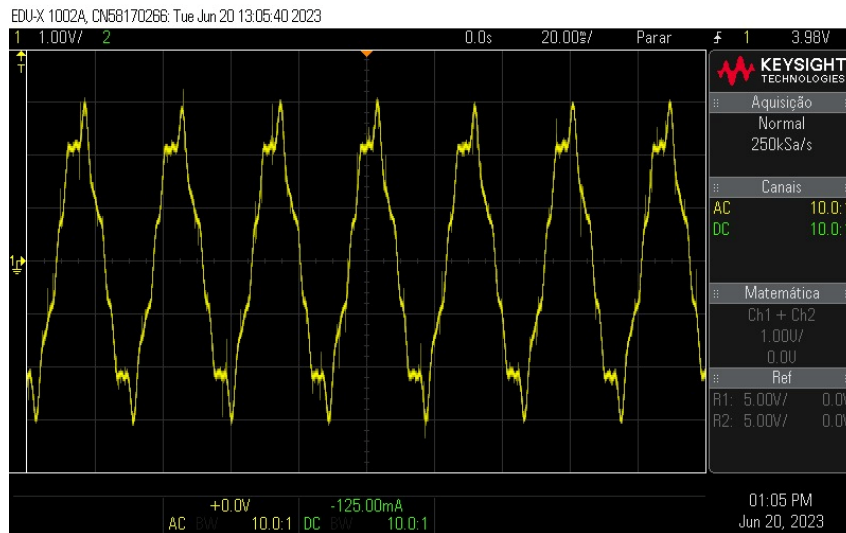
Desta forma, aplicando a Equação 2.11 aos valores obtidos nos ensaios para as três fases do motor, a resistência entre cada par de fase é calculada na Equação 4.1.

$$\begin{aligned}
 R_{12} &= \frac{V}{I} = \frac{0,251[V]}{2,99[A]} = 0,0839\Omega \\
 R_{13} &= \frac{V}{I} = \frac{0,257[V]}{2,99[A]} = 0,0859\Omega \\
 R_{23} &= \frac{V}{I} = \frac{0,263[V]}{3,07[A]} = 0,0857\Omega
 \end{aligned}
 \tag{4.1}$$

Com isso, é possível verificar que a resistência entre as fases apresenta uma diferença inferior a 3 %, o que caracteriza um motor sem avarias nos enrolamentos, conforme descrito na seção 2.2.1.

Para a determinação da FEM induzida pelo motor, foi utilizada uma máquina primária, com o eixo acoplado no *BLDC*. Fazendo com que essa máquina primária rotacionasse o *BLDC* a uma velocidade de $71,21 \frac{rad}{s}$ foi obtida um forma de onda vista na Figura 42.

Figura 42: Forma de onda obtida no ensaio com máquina primária



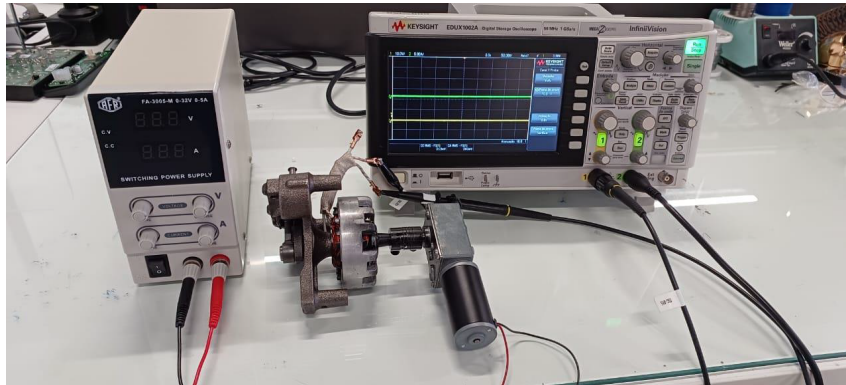
Fonte: Autor (2023)

Aplicando a Equação 2.15 ao valor de tensão obtido neste ensaio, com a velocidade de rotação do eixo, é possível determinar a constante do motor, como visto na Equação 4.2.

$$K' = \frac{V_T}{\omega_m} = \frac{3[V]}{71,21[\frac{Rad}{s}]} = 0,0421[\frac{V}{\frac{Rad}{s}}] \quad (4.2)$$

A bancada de teste utilizada para os ensaios descritos é mostrada na Figura 43.

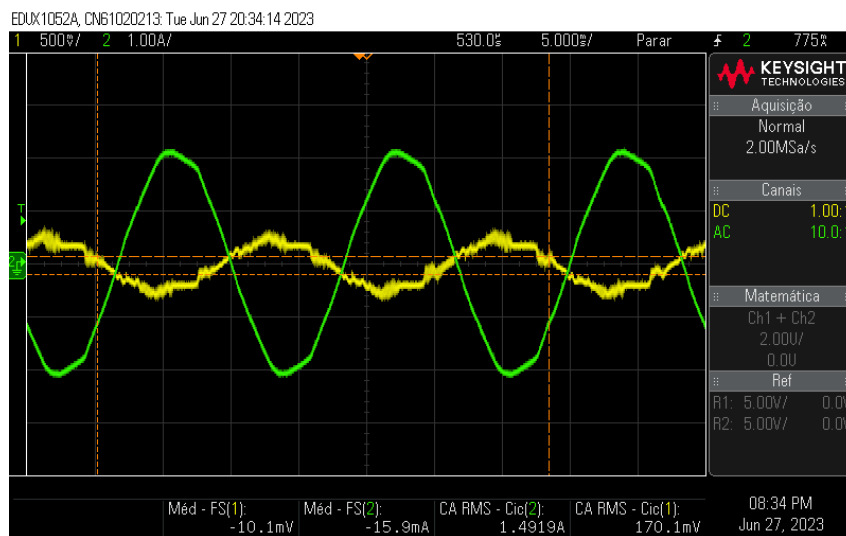
Figura 43: Bancada de testes para determinação dos parâmetros



Fonte: Autor (2023)

Conforme o método descrito na seção 2.2.3 a indutância equivalente entre dois terminais do motor pode ser determinada utilizando uma fonte CA ligada entre duas fases do motor. O valor de tensão aplicado aos terminais, bem como o valor de corrente obtido pode ser visto na Figura 44.

Figura 44: Formas de onda obtidas no ensaio de corrente alternada



Fonte: Autor (2023)

Neste ensaio, com uma frequência de 60 Hz, os valores RMS de tensão e corrente obtidos foram 170,1 mV e 1,492 A, respectivamente. Aplicando esses valores e o valor médio da resistência de fase obtido anteriormente à Equação 2.16, a indutância equivalente entre duas fases é calculada na Equação 4.3.

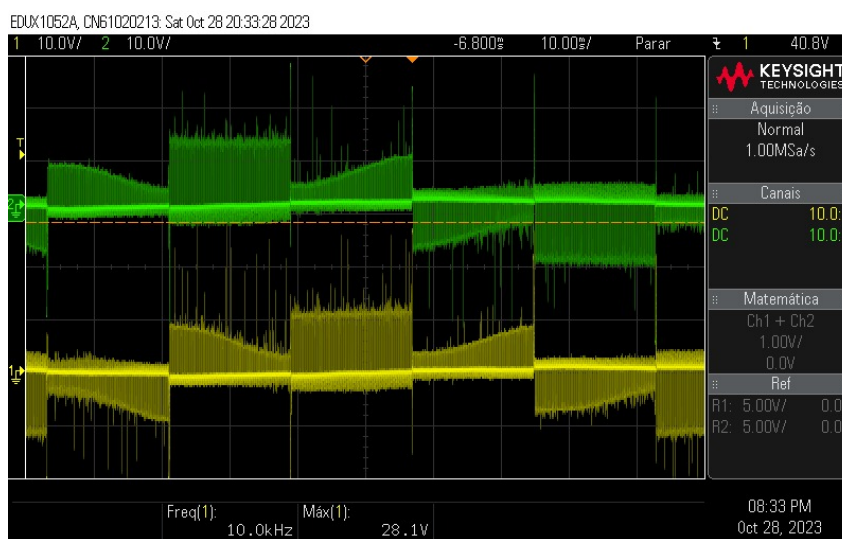
$$L_{ab} = \frac{\sqrt{\left(\frac{0,1701}{1,492}\right)^2 - 0,0851^2}}{2 \cdot \pi \cdot 60} = 201,244\mu H \quad (4.3)$$

4.2 FORMAS DE ONDA OBTIDAS NO ACIONAMENTO

Utilizando um osciloscópio Keysight modelo EDUX1052A foi possível capturar as formas de onda de acionamento do motor, sendo utilizadas para comparar os sinais esperados, descritos nas seções anteriores com os sinais efetivamente aplicados à máquina.

Para o primeiro teste, visando analisar a sequência de acionamento das chaves de potência, utilizou-se uma carga puramente resistiva¹. Com o sistema configurado para gerar uma rotação equivalente à 200 RPM no motor, foi possível observar, entre as fases A e B (em amarelo) e entre as fases C e B (em verde) a forma de onda vista na Figura 45. Destacado na imagem, encontra-se as chaves acionadas em cada momento.

Figura 45: Formas de onda da tensão entre as fases

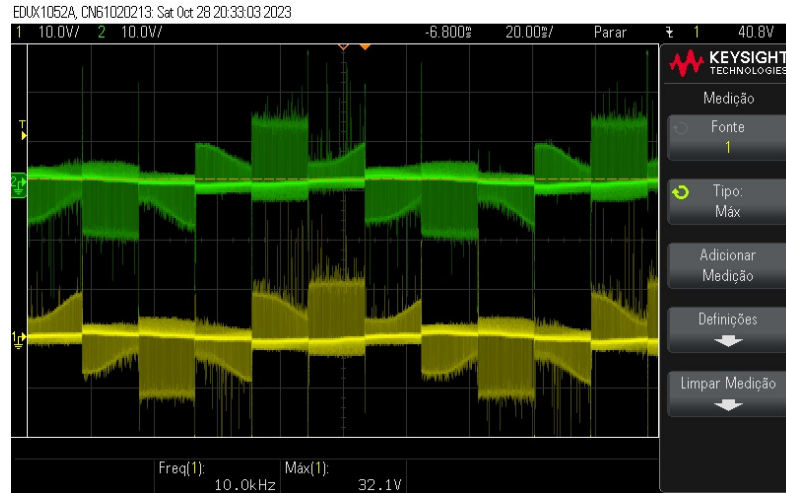


Fonte: Autor (2023)

No segundo teste, para as mesmas condições do teste anterior, agora com o motor sem carga mecânica no eixo conectado às saídas, foi possível observar a forma de onda representada pela Figura 46.

¹ Considera-se àquela em que a parcela indutiva e/ou capacitiva sejam desprezíveis quando comparadas ao valor de resistência.

Figura 46: Formas de onda da tensão entre as fases

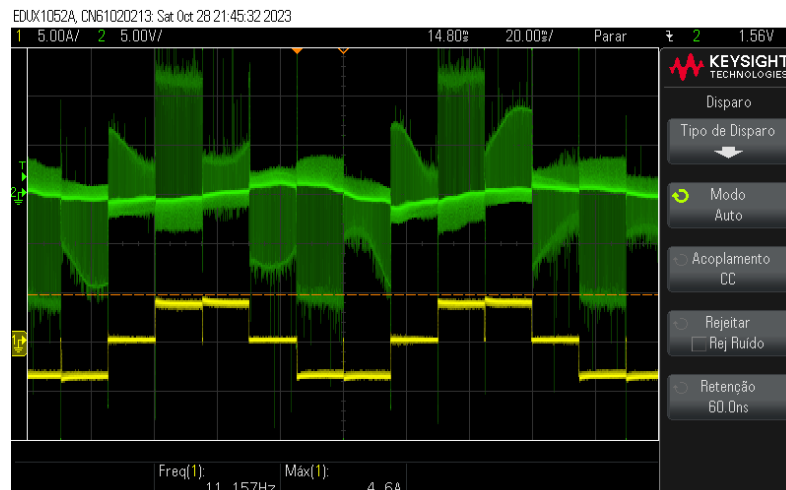


Fonte: Autor (2023)

Nota-se, em destaque, a presença da FEM proveniente da rotação do motor, conforme descrito nas seções anteriores. Ainda, utilizando uma ponta de prova de corrente no segundo canal do osciloscópio, foi possível capturar (em verde) a forma de onda de corrente circulando entre as fases A e B.

Todos os ensaios realizados até esse ponto consideram o sistema em regime permanente. Sendo assim, os pontos de alinhamento e aceleração foram desconsiderados. Para englobar esses dois casos, o circuito de *drive* foi conectado ao sistema de refrigeração completo, submetendo a máquina ao torque nominal de trabalho. Assim, partindo da velocidade de 0 RPM, e acelerando até a velocidade de 200 RPM, foi possível observar as formas de onda de tensão entre as fases A e B e a corrente na fase A, presentes na Figura 47.

Figura 47: Corrente e tensão com o motor submetido ao Torque nominal

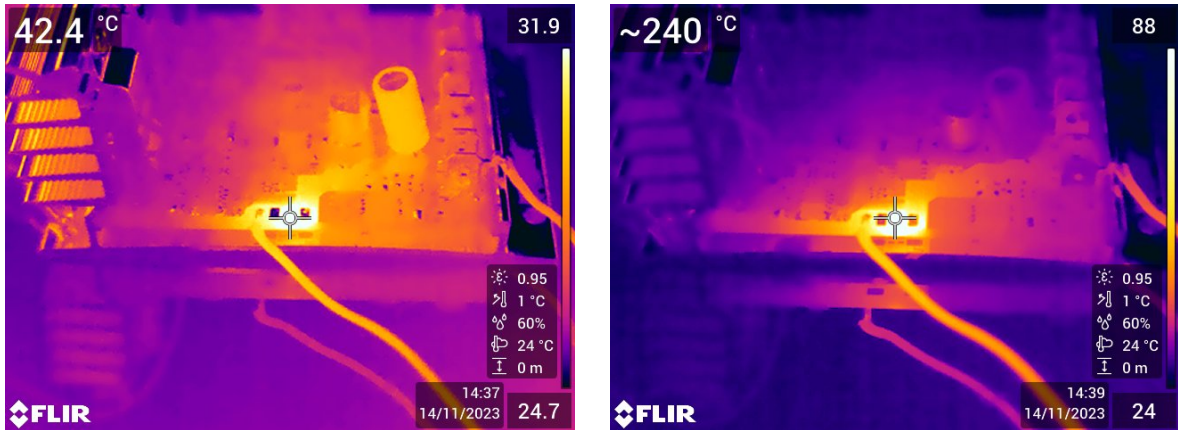


Fonte: Autor (2023)

4.3 PROTEÇÕES

Com o auxílio de uma câmera termográfica FLIR modelo E95 foi possível analisar a elevação de temperatura no *fuse-track*, ocasionada por sobrecorrentes no barramento de potência, como pode ser observado na Figura 48.

Figura 48: Análise termográfica na região do *fuse-track* para diferentes correntes



(a) Corrente de 3 A no barramento.

(b) Corrente de 11A no barramento.

Ao atingir uma temperatura de 300 °C, ocasionado por uma corrente de 12 A, ocorreu o rompimento do *fuse-track*, protegendo a fonte e as chaves de potência de danos maiores. A atuação do sistema implica em pequenos danos ao material base da PCI, como é visto na Figura 49.

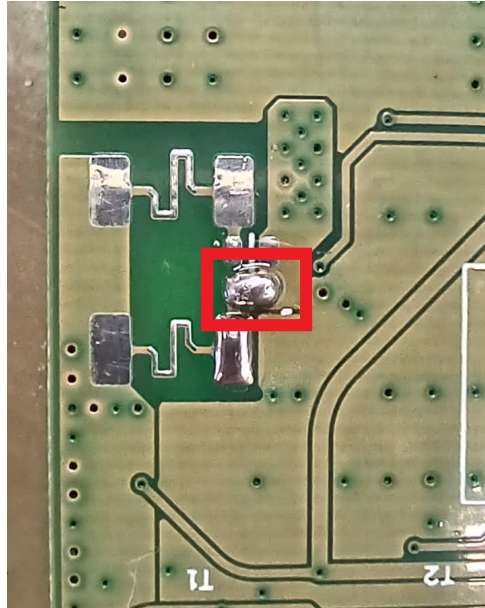
Figura 49: Danos ocasionadas na PCI pela atuação da proteção



Fonte: Autor (2023)

Para reativar os circuitos, como previsto, bastou realizar um curto-circuito em pontos específicos na placa, habilitando os *fuse-tracks* remanescentes, como é visto na Figura 50.

Figura 50: Opção de religamento de *fuse-track sobressalente*



Fonte: Autor (2023)

5 CONSIDERAÇÕES FINAIS

5.1 DISCUSSÃO DOS RESULTADOS

A análise dos resultados práticos demonstra que o sistema desenvolvido cumpriu com os objetivos básicos de operação, além de atingir todos requisitos propostos no início do projeto. Através dos testes práticos, observou-se o correto comportamento do sistema eletrônico, com o sistema operando tanto com o motor livre, quanto com a carga mecânica nominal demandada.

É possível afirmar que, através dos ensaios com máquina primária, de rotor bloqueado e com o ensaio CA, foi possível determinar os parâmetros do motor. Observa-se também, que todos os componentes de potência foram capazes de suportar os esforços requeridos, comprovando que as características de tensão e corrente foram dimensionadas de acordo com a operação. Com isso, observa-se que tanto a topologia de acionamento proposta, quanto o *hardware* e o *firmware* desenvolvidos foram eficazes no controle e acionamento da máquina. Apesar disso, o sistema de dissipação térmica dos MOSFETs SMD mostrou-se com capacidade inferior à necessária, fazendo que a temperatura nas chaves ultrapassasse os limites máximos. Com isso, pode-se concluir que os MOSFETs utilizados podem ser utilizados para potências menores, onde o aquecimento é reduzido. Alternativamente, caso outras técnicas de dissipação sejam utilizadas, como a utilização de dissipadores para componentes SMD estas chaves podem vir a ser utilizadas. Sendo este risco observado desde o início do projeto, os MOSFETs SMD foram substituídos por componentes *PTH* (*Pin Through-hole*) e, além disso, foi necessário utilizar um radiador de calor com convecção forçada para manter a temperatura das chaves dentro dos parâmetros operacionais.

Por fim, a proteção de sobretensão mostrou-se efetiva, tanto com baixas sobretensões por longos períodos, quanto contra sobretensões elevadas de curta duração. Além disso, observou-se que a trilha fusível foi capaz de proteger o sistema contra curto-circuitos em 12 A, sem que houvessem danos consideráveis ao material base da PCI.

5.2 TRABALHOS FUTUROS

Para trabalhos futuros, sugere-se a implementação de métodos de controle mais precisos para garantir o melhor aproveitamento do conjugado eletromagnético da máquina, além de diminuir a vibração ocasionada pelos elementos mecânicos do compressor. Para ampliar as possibilidades de uso do sistema, alterações no *hardware* podem ser empregadas para possibilitar o uso de frenagens regenerativas. Isso permite que o sistema seja utilizado em veículos elétricos, aumentando a autonomia e flexibilidade do sistema. Além disso, após a análise das etapas implementadas, pode-se desenvolver um novo *layout* da PCI, removendo os circuitos que não se mostraram necessários, reduzindo a área total da PCI e, conseqüentemente, reduzindo seu custo. Por fim, análises térmicas mais apuradas podem ser realizadas, de modo a averiguar

o funcionamento tanto do sistema quanto da máquina sob as mais diversas condições de operação. Além disso, a análise do perfil térmico pode permitir o dimensionamento mais apurado do sistema de refrigeração, o que pode levar a substituição das chaves de potência por dispositivos com potência inferior aos utilizados.

REFERÊNCIAS

- ALBUQUERQUE, C. M. **Desenvolvimento de sistema de controle para motor 30 v 3 fases**. Rio Grande do Sul, Brasil, 2020. 71 p.
- BARATIERI, C. L. **Controle de velocidade sensorless de motores brushless dc submetidos a variações periódicas de carga**. 2011. 151 p. Dissertação (Mestrado) — Universidade Federal de Santa Maria, Rio Grande do Sul, Brasil, 2011.
- BARBI, I. **Eletrônica de potência: projetos de fontes chaveadas**. 3^a. ed. Floorianópolis, Brasil: Oxford University Press, 2001.
- BROOKS, D. Fusing current when traces melt without a trace. **Printed Circuit Design**, [S.l.], 1998.
- BROOKS DOUGLAS; ADAM, J. Fusing current in traces. , [S.l.], 2015.
- CHANG YEN-CHUAN; TZOU, Y.-Y. A new sensorless starting method for brushless dc motors without reversing rotation. **IEEE Transactions on Industrial Electronics**, [S.l.], p. 619–624, 2007.
- CHAPMAN, S. J. **Fundamentos de máquinas elétricas**. 5^a. ed. Porto Alegre, Brasil: AMGH Editora, 2013.
- DIMITROV BORISLAV; CRUDEN, A. S.-S. Determining pcb trace capability and fusing, used in switched mode power supplies, based on modeling and simulation with the finite element method. **Acta Electrotechnica et Informatica**, [S.l.], 2018.
- GIESEL, T. **Estudo e projeto de motores de alta velocidade para aplicação em compressores herméticos**. 2007. 113 p. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Santa Catarina, Brasil, 2007.
- HANSELMAN, D. C. **Brushless permanent magnet motor design**. 2^a. ed. Lebanon, Ohio: Magna Physics Publishing, 2006.
- IIZUKA K; UZUHASHI, H. K.-M. E. T. M. K. Microcomputer control for sensorless brushless motor. **IEEE Transactions on Industrial Electronics**, [S.l.], p. 595–601, 1985.
- KARTHIKEYAN; SEKARAN, D. Dc-dc converter csi fed bldc motor for defence applications. **IEEE Transactions on Industrial Electronics**, [S.l.], p. 68–72, 2011.
- KRISHNAN, R. **Electric motor drives: modeling, analysis and control**. Blacksburg, Virginia, U.S.A.: Prentice Hall, 2001.
- MILLER, H. **Design of brushless permanent-magnet machines**. Venice, Florida, U.S.A.: Motor Design Books LLC, 2010.
- MORITZ, R. M. **Estudo de perdas por comutação, condução e cálculo térmico de um inversor trifásico de dois níveis**. Santa Catarina, Brasil, 2014. 121 p.

OGASAWARA; AKAGI, H. An approach to position sensorless drive for brushless dc motors. **IEEE Transactions on Industrial Electronics**, [S.l.], v. 27, p. 928–933, 1991.

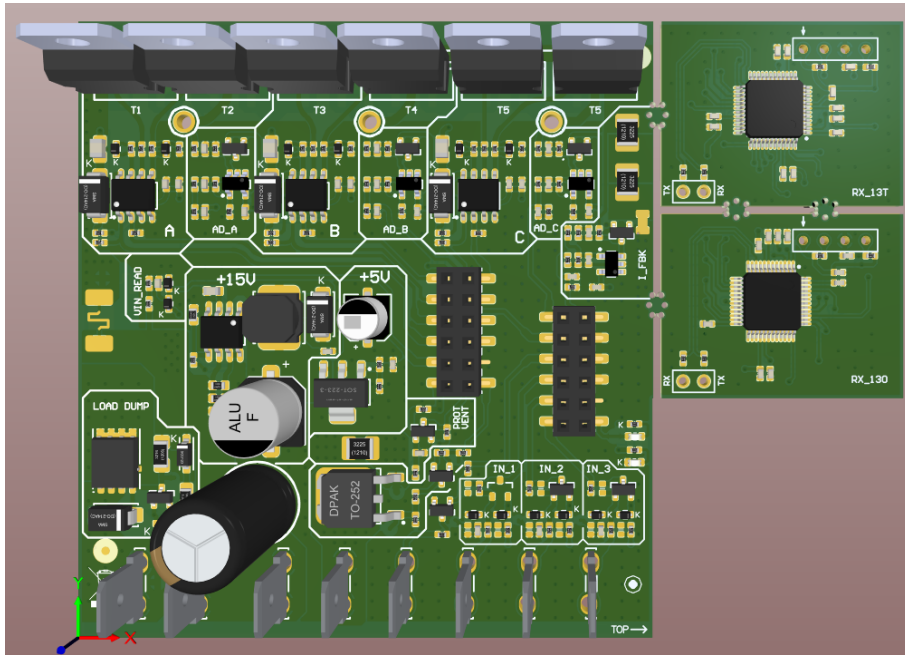
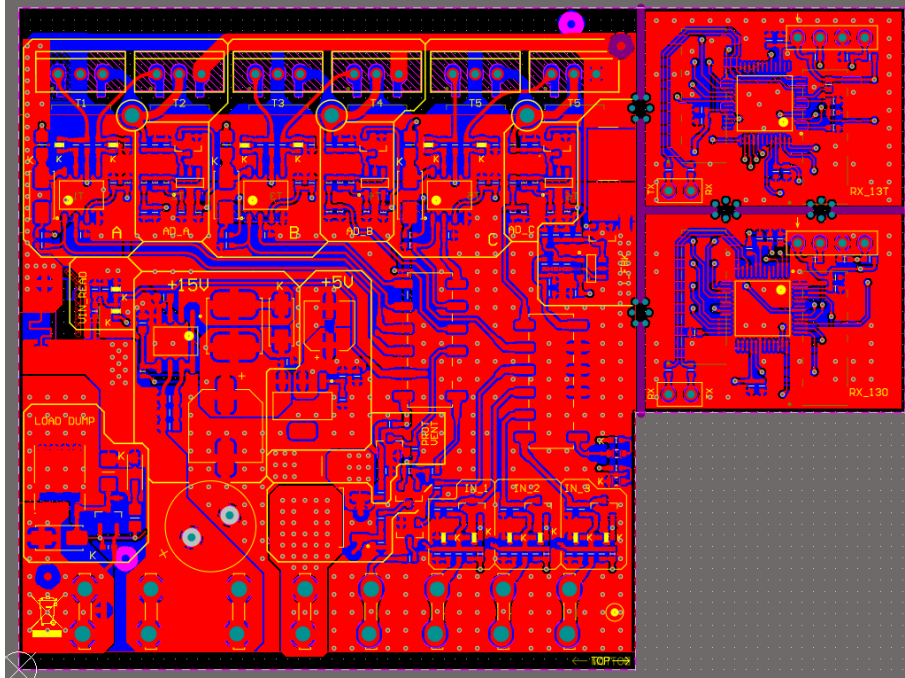
RASHID, M. H. **Eletrônica de potência: dispositivos, circuitos e aplicações**. 4^a. ed. [S.l.: s.n.], 2014.

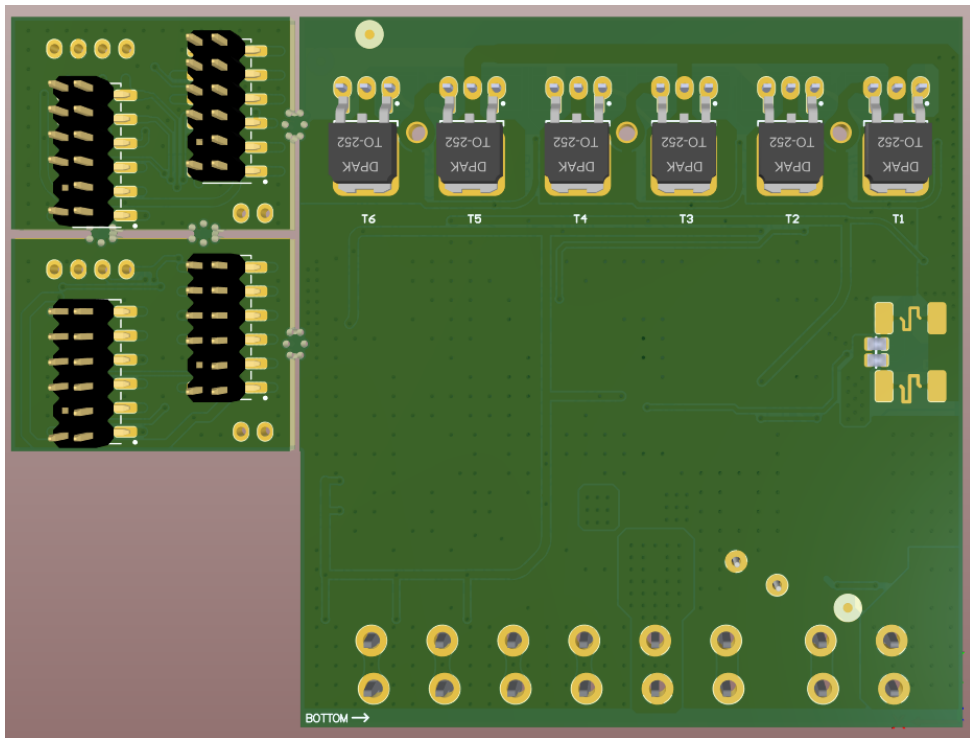
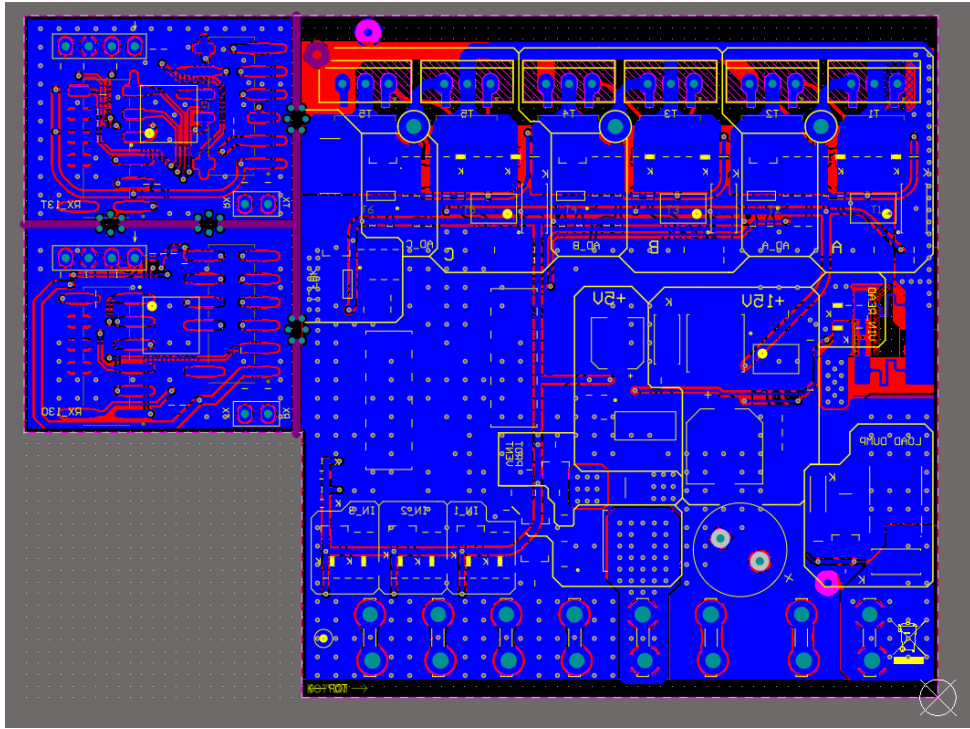
SHAO JIANWEN; NOLAN, D. T.-M. S. D. A novel microcontroller-based sensorless brushless dc (bldc) motor drive for automotive fuel pumps. **IEEE Transactions on Industrial Electronics**, [S.l.], p. 2386–2392, 2002.

UMANS, S. D. **Máquinas elétricas de fitzgerald e kingsley**. 7^a. ed. Porto Alegre, Brasil: AMGH, 2014.

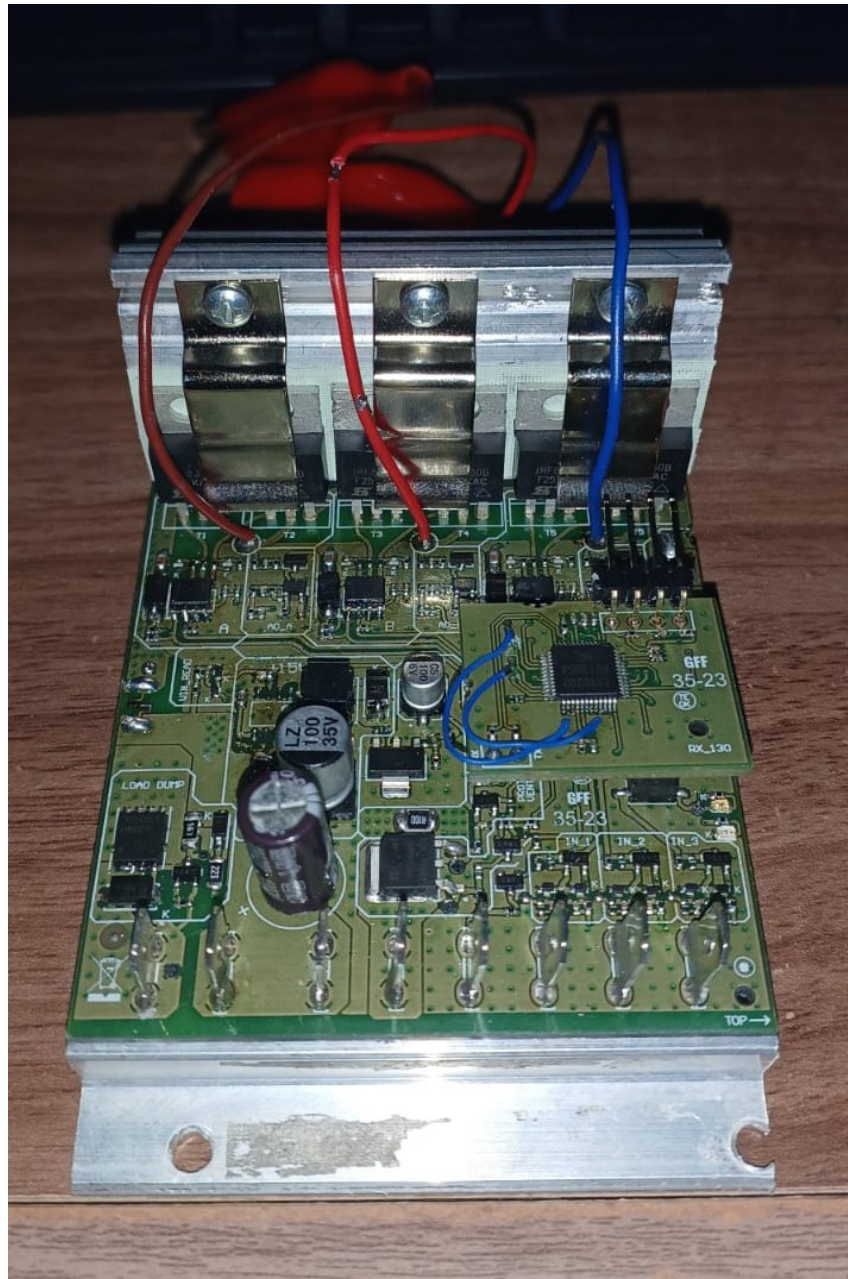
XIA, C. L. **Permanent magnet brushless dc motor drives and controls**. 1^a. ed. Singapura: John Wiley and Sons, 2012.

APÊNDICE A - PROJETO DA PLACA DE CIRCUITO IMPRESSO

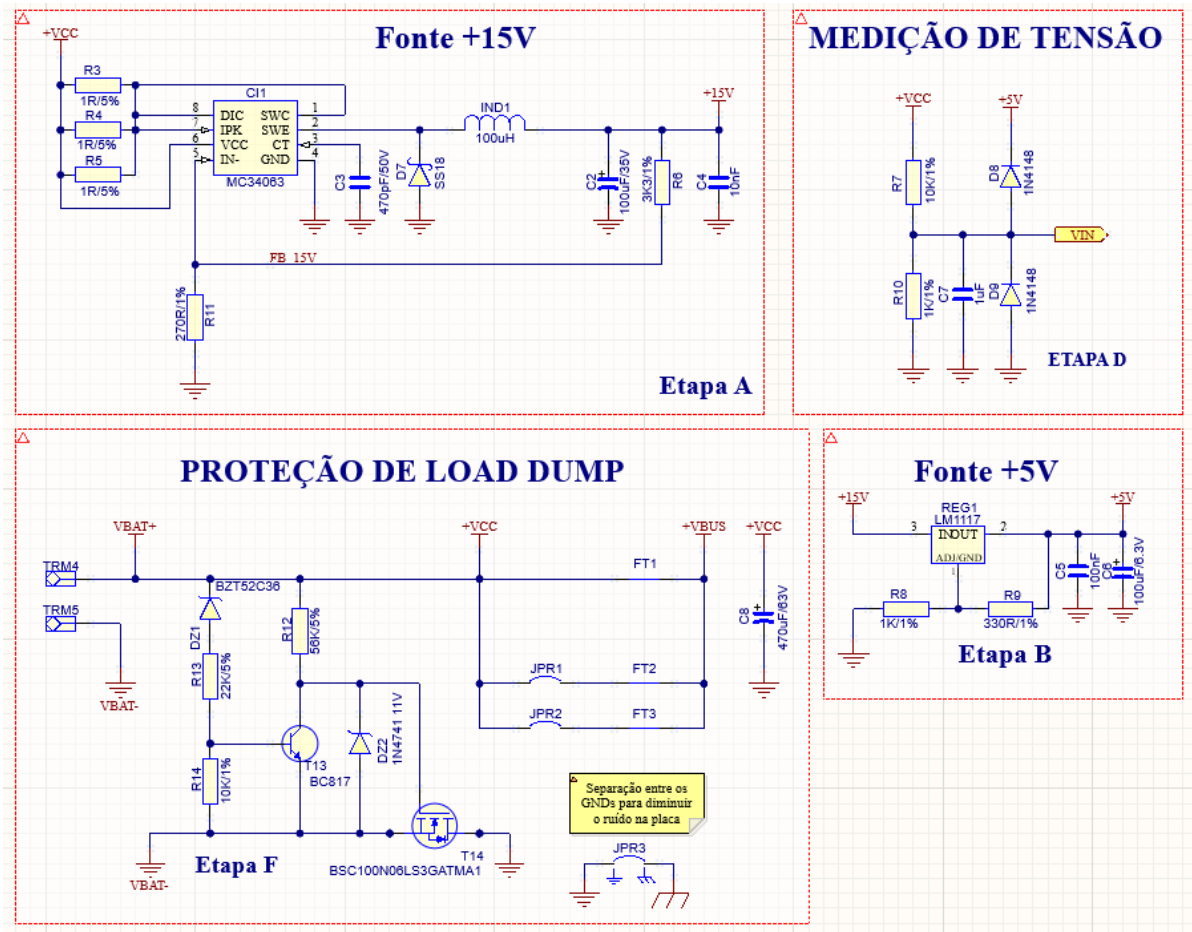




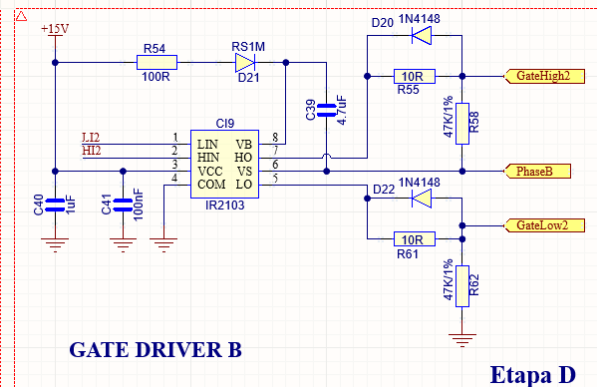
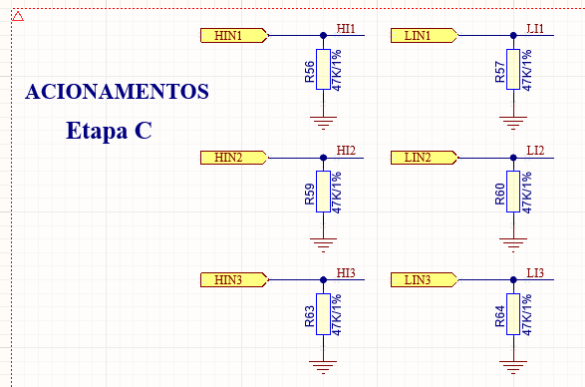
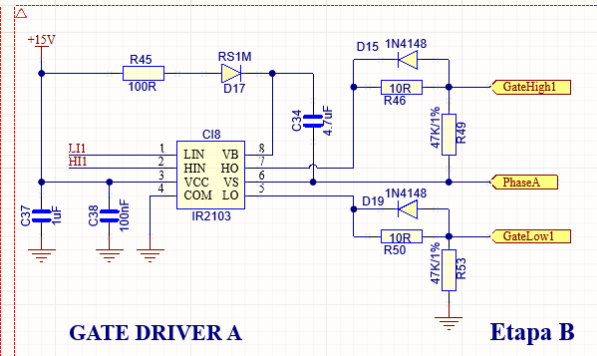
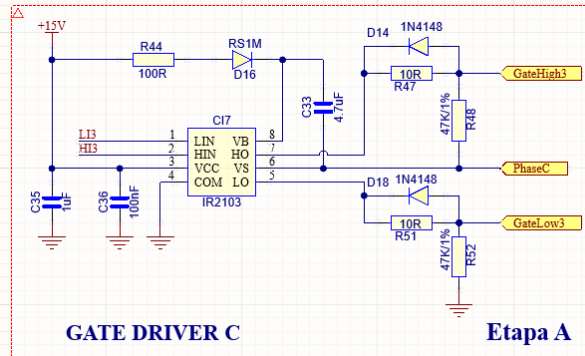
APÊNDICE B - PROJETO MONTADO



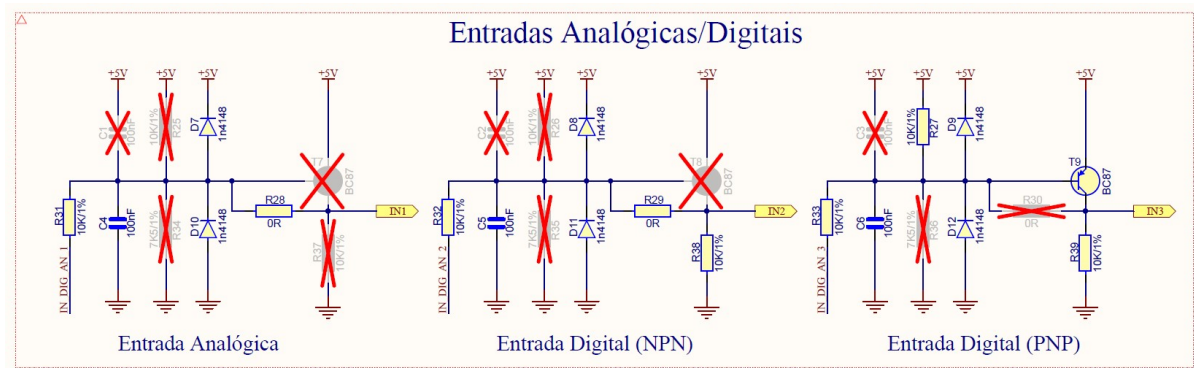
APÊNDICE C - ESQUEMÁTICO COMPLETO DAS FONTES



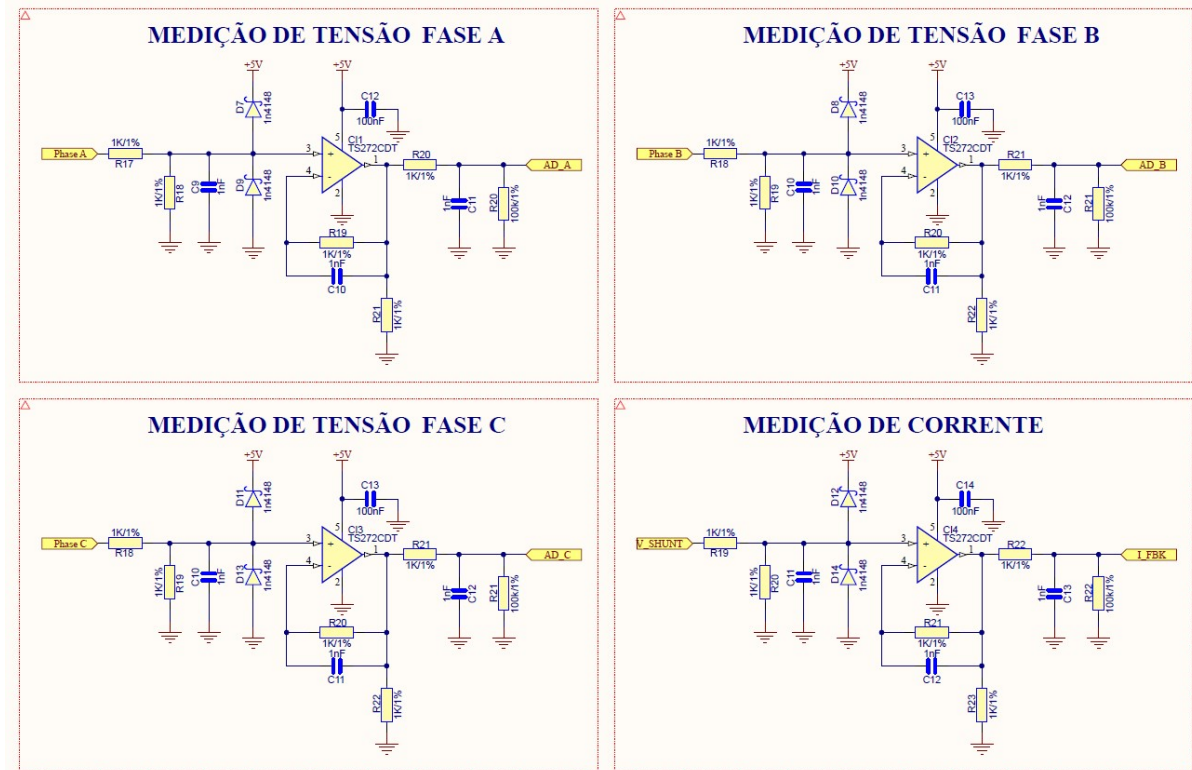
APÊNDICE D - ESQUEMÁTICO COMPLETO DOS GATE-DRIVERS



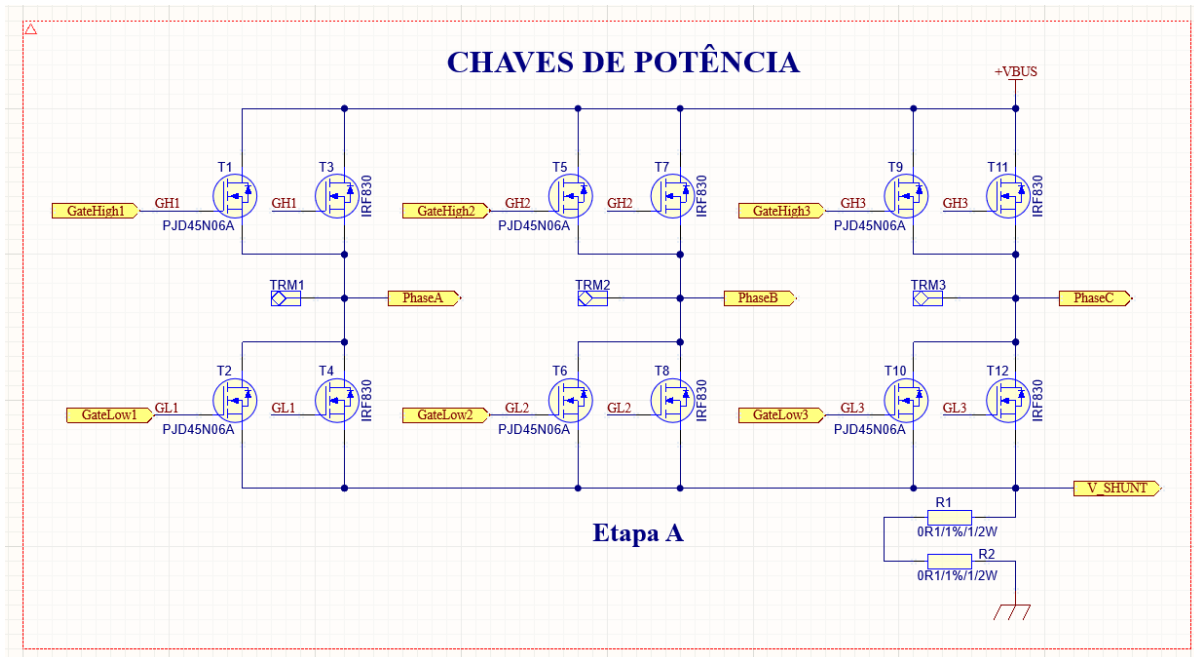
APÊNDICE E - ESQUEMÁTICO COMPLETO DAS ENTRADAS ANALÓGICAS/DIGITAIS



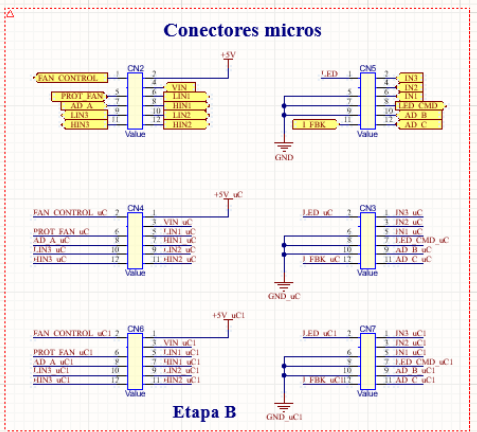
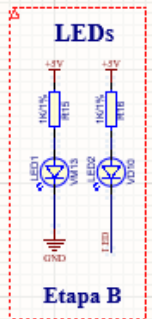
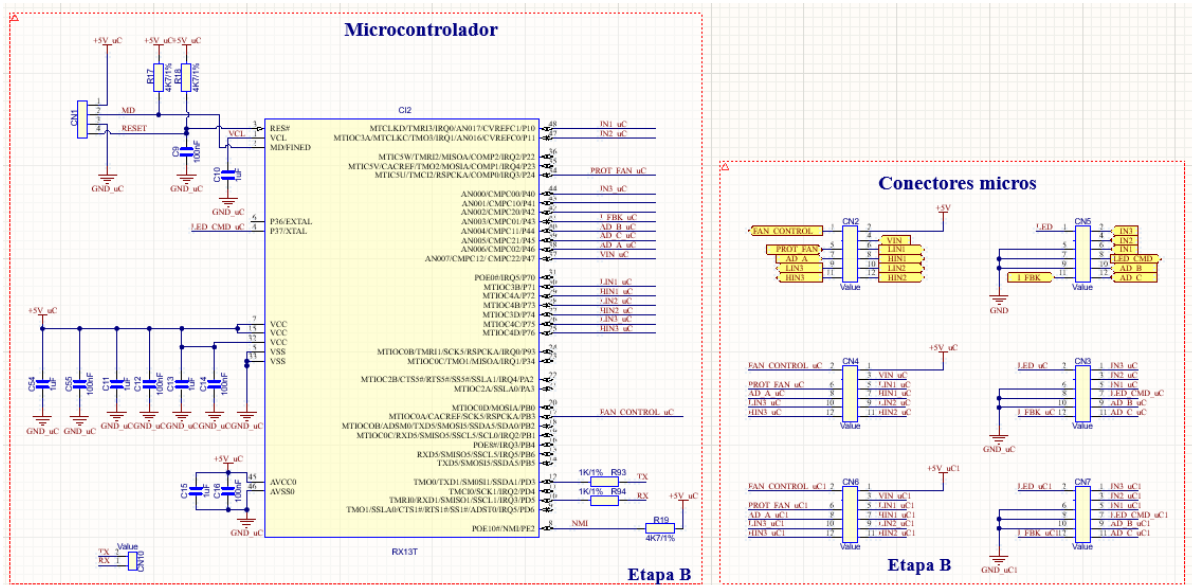
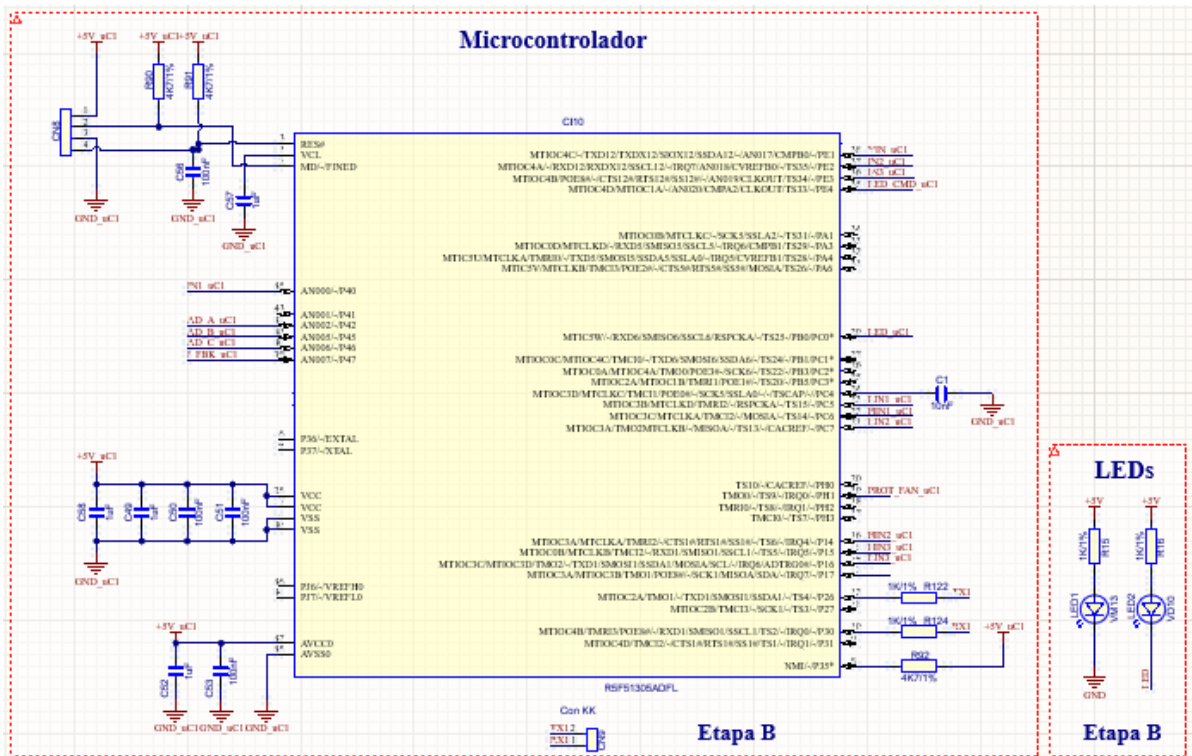
APÊNDICE F - ESQUEMÁTICO COMPLETO PARA LEITURA DA BEMF



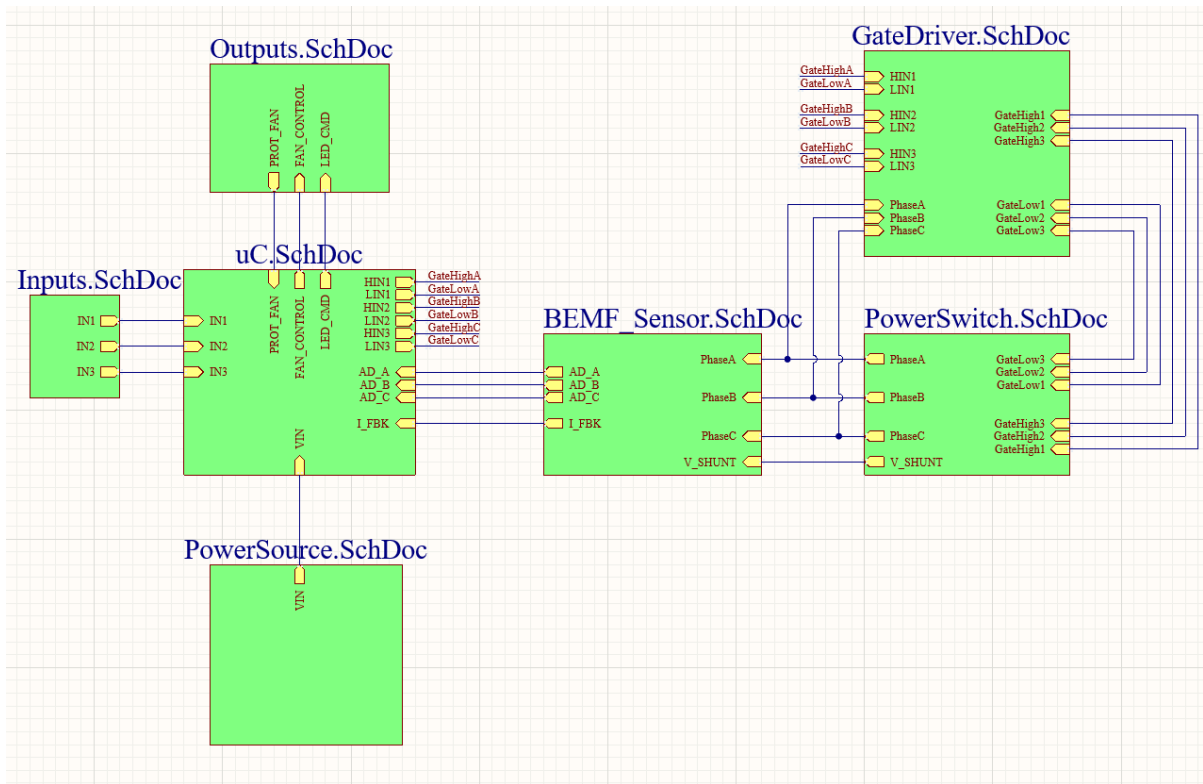
APÊNDICE G - ESQUEMÁTICO COMPLETO DAS CHAVES DE POTÊNCIA



APÊNDICE H - ESQUEMÁTICO COMPLETO DOS TESTES DE MICROCONTROLADOR



APÊNDICE I - ESQUEMÁTICO COM A REPRESENTAÇÃO DA LIGAÇÃO ENTRE OS CIRCUITOS



APÊNDICE J - MÁQUINA DE ESTADOS PARA PROCESSAMENTO NÃO CRÍTICO

```

/**
*****
 * @file      BLDC.c
 * @brief
 * @version   0.1
 * @date
 * @author    Vinicius Velho
*****
 * @section License
 * <b>(C) Copyright 2023 Inova – All rights reserved
*****
 */
/* Includes ----- */
#include "r_smc_entry.h"
#include "Moleware/adc.h"
#include "Moleware/LED.h"
#include "Moleware/Inputs.h"
#include "Moleware/battery.h"
#include "Moleware/FanControl.h"
#include "Moleware/WaveGenerator.h"
#include "Moleware/timing.h"
/* Private typedef ----- */
/* Private macros ----- */
/* Private variables ----- */
static struct
{
    uint8_t stateOfMachine;
    tim_t finitStateMachineTimer;
} bldc;
/* Coupling functions ----- */
/* Private function prototypes ----- */
/* Private functions ----- */
/* Public functions ----- */
void main(void);

void main(void)
{
    // Inicializacao dos perifericos
    R_Config_TMR0_Start();
    R_Config_MTU0_Start();

    // Desabilita os canais de PWM
    MTU0.TIORL.BYTE &= 0x0F;
    MTU0.TIORH.BYTE &= 0x0F;

```

```

// Instancia o temporizador da maquina de estados principal
timing_CreateTimer(&bldc.finitStateMachineTimer , d1_MILI_SECOND);
// Habilita o temporizador da maquina de estados
timing_EnableDisableTimer(&bldc.finitStateMachineTimer , eTIM_ENABLE);

//Inicializa os submodulos
FanControl_Init(battery_GetOperationVoltage());
WaveGenerator_Init();
LED_Init();
LED_SetStatus(eBLINK, eLED_1);
battery_SetOperationVoltage(battery_GetInputVoltage());
WaveGenerator_SetRPM(3000);
WaveGenerator_Cmd(1);

for (;;)
{
    //Atualiza o controle dos modulos a cada 1ms
    if (timing_GetTickStatus(&bldc.finitStateMachineTimer))
    {
        switch (bldc.stateOfMachine)
        {
            case 0:
                // Trata o controle das saidas para os LEDs
                LED_Handler();
                bldc.stateOfMachine++;
                break;

            case 1:
                // Trata os sinais de entrada
                Inputs_Handler();
                bldc.stateOfMachine++;
                break;

            case 2:
                // Trata a protecao da tensao de entrada
                battery_Handler();
                bldc.stateOfMachine++;
                break;

            case 3:
                // Controla a velocidade do ventilador externo
                FanControl_Handler();
                bldc.stateOfMachine++;
                break;

            default:

```

```
    bldc.stateOfMachine = 0;
}

    timing_ResetTick(&bldc.finitStateMachineTimer);
}

}
}
/***** (C) COPYRIGHT Inova *****/
```

APÊNDICE K - CÓDIGO DE PROCESSAMENTO CRÍTICO

```

/**
*****
* @file      WaveGenerator.c
* @brief
* @version   0.1
* @date
* @author    Vinicius Velho
*****
* @section License
* <b>(C) Copyright 2023 Inova – All rights reserved
*****
*/

/* Includes ----- */
#include <stdint.h>
#include "bsp.h"
#include "WaveGenerator.h"
#include "timing.h"
#include "r_smc_entry.h"
#include "stdint.h"
/* Private defines ----- */
#define dCONFIG_DEAD_TIME 10
/// Caso chegue nessa velocidade em open-loop, ha algum problema de rotacao
no motor
#define dVELOCITY_ERROR_RPM 80000
/// Step de aceleracao do motor [RPM]
#define dACCELERATE_RPM_STEP 0
/// Frequencia base para os calculos de velocidade [Hz]
#define dCARRIER_FREQUENCY 10000
/// Resolucao dos acionamentos [grau]
#define dANGLE_RESOLUTION 120
/// Resolucao de uma rotacao [grau]
#define d1_REVOLUTION_ANGLE 360

/// Converte a frequencia [Hz] para tempo [s]
#define dFREQ_TO_TIME(freq) (1 / freq)
/// Converte o valor de rotacao em RPM para HZ
#define dRPM_TO_HZ(RPM) (RPM / 60)

#define dFORCED_ALIGNMENT dTRUE

#define dOPEN_LOOP_SWITCHING_MODE_1 dTRUE
#define dOPEN_LOOP_SWITCHING_MODE_2 dFALSE
#define dOPEN_LOOP_SWITCHING_MODE_3 dFALSE

```



```

#define dOPEN_LOOP_SWITCHING_MODE_4 dFALSE

#define dCLOSED_LOOP_SWITCHING_MODE_1 dTRUE
#define dCLOSED_LOOP_SWITCHING_MODE_2 dFALSE
#define dCLOSED_LOOP_SWITCHING_MODE_3 dFALSE
#define dCLOSED_LOOP_SWITCHING_MODE_4 dFALSE
/* Private typedef
-----*/
typedef struct
{
    uint8_t S1State;
    uint8_t S2State;
    uint8_t S3State;
    uint8_t S4State;
    uint8_t S5State;
    uint8_t S6State;
} outputsState_t;

typedef struct
{
    uint8_t S1Pwm;
    uint8_t S2Pwm;
    uint8_t S3Pwm;
    uint8_t S4Pwm;
    uint8_t S5Pwm;
    uint8_t S6Pwm;
} pwmValues_t;

typedef struct
{
    volatile float operationFrequencyHz;
    uint16_t operationRPM;
    uint16_t phaseTime;
} velocity_t;

typedef struct
{
    uint8_t countPhaseA;
    uint8_t countPhaseB;
    uint8_t countPhaseC;
} zeroCrossing_t;

static struct
{
    velocity_t targetVelocity;
    velocity_t currentVelocity;
}

```

```

zeroCrossing_t zeroCrossing;
trapezoidalChunks_t currentChunk;
motorControlState_t motorControlCurrentState;
switchSelect_t switchSelect;
outputsState_t switchStates;
pwmValues_t switchPwm;
tim_t waveTimer;

uint32_t timeCount;
uint32_t phaseTime;
uint32_t deadTime;

uint8_t isBEMFDetected;
uint8_t BEMFDebounceCount;
} wave;

/* Private macros ----- */
/* Private variables ----- */
/* Coupling functions ----- */
/* Private function prototypes ----- */
void WaveGenerator_CalculateTime( velocity_t *velocityType);
static void WaveGenerator_WaveCalculate( void);
static void WaveGenerator_SwitchesCmd( switchSelect_t switchesSelect);
#if dOPEN_LOOP_SWITCHING_MODE_1 == dTRUE
static void WaveGenerator_SwitchingMode1( void);
#endif

#if dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE
static void WaveGenerator_SwitchingMode2( void);
#endif

#if dOPEN_LOOP_SWITCHING_MODE_3 == dTRUE
static void WaveGenerator_SwitchingMode3( void);
#endif

#if dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
static void WaveGenerator_SwitchingMode4( void);
#endif

static void WaveGenerator_SetPin( outputsState_t *pins);
static void WaveGenerator_SetPWM( pwmValues_t *pwm);

#if dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE || dOPEN_LOOP_SWITCHING_MODE_3 ==
dTRUE || dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
static uint8_t WaveGenerator_WaveformHandler( switchSelect_t switchSelected ,
enabledChunks_t chunkControl);
#endif
/* Private functions ----- */

```

```

/*****
/** @brief
 * @param
 * @retval
*****/
static void WaveGenerator_SetPWM( pwmValues_t *pwm)
{
    if (pwm->S1Pwm == 0)
    {
        dPWM_DISABLE_CH1();
    }
    else
    {
        //dSET_PWM_CH1(pwm->S1Pwm);
        dPWM_ENABLE_CH1();
    }

    if (pwm->S3Pwm == 0)
    {
        dPWM_DISABLE_CH3();
    }
    else
    {
        //dSET_PWM_CH3(pwm->S3Pwm);
        dPWM_ENABLE_CH3();
    }

    if (pwm->S5Pwm == 0)
    {
        dPWM_DISABLE_CH5();
    }
    else
    {
        //dSET_PWM_CH5(pwm->S5Pwm);
        dPWM_ENABLE_CH5();
    }
}

/*****
/** @brief
 * @param
 * @retval
*****/
static void WaveGenerator_SetPin( outputsState_t *pins)

```

```
{
  if (pins->S1State == 0)
  {
    wave.switchPwm.S1Pwm = 0;
  }
  else
  {
    wave.switchPwm.S1Pwm = 50;
  }
  if (pins->S3State == 0)
  {
    wave.switchPwm.S3Pwm = 0;
  }
  else
  {
    wave.switchPwm.S3Pwm = 50;
  }
  if (pins->S5State == 0)
  {
    wave.switchPwm.S5Pwm = 0;
  }
  else
  {
    wave.switchPwm.S5Pwm = 50;
  }
  WaveGenerator_SetPWM(&wave.switchPwm);

  if (pins->S2State == 1)
  {
    dS2_WRITE_HIGH();
  }
  else
  {
    dS2_WRITE_LOW();
  }

  if (pins->S4State == 1)
  {
    dS4_WRITE_HIGH();
  }
  else
  {
    dS4_WRITE_LOW();
  }

  if (pins->S6State == 1)
  {
```

```

    dS6_WRITE_HIGH() ;
}
else
{
    dS6_WRITE_LOW() ;
}
}

/*****
/** @brief
 * @param
 * @retval
*****/

static void WaveGenerator_SwitchesCmd( switchSelect_t switchesSelect)
{
    switch ( switchesSelect)
    {
        case eS1_S4:
            wave.switchStates.S1State = dTRUE;
            wave.switchStates.S2State = dFALSE;
            wave.switchStates.S3State = dFALSE;
            wave.switchStates.S4State = dTRUE;
            wave.switchStates.S5State = dFALSE;
            wave.switchStates.S6State = dFALSE;
            break;

        case eS1_S6:
            wave.switchStates.S1State = dTRUE;
            wave.switchStates.S2State = dFALSE;
            wave.switchStates.S3State = dFALSE;
            wave.switchStates.S4State = dFALSE;
            wave.switchStates.S5State = dFALSE;
            wave.switchStates.S6State = dTRUE;
            break;

        case eS3_S6:
            wave.switchStates.S1State = dFALSE;
            wave.switchStates.S2State = dFALSE;
            wave.switchStates.S3State = dTRUE;
            wave.switchStates.S4State = dFALSE;
            wave.switchStates.S5State = dFALSE;
            wave.switchStates.S6State = dTRUE;
            break;

        case eS3_S2:

```

```

    wave.switchStates.S1State = dFALSE;
    wave.switchStates.S2State = dTRUE;
    wave.switchStates.S3State = dTRUE;
    wave.switchStates.S4State = dFALSE;
    wave.switchStates.S5State = dFALSE;
    wave.switchStates.S6State = dFALSE;
    break;

case eS5_S2:
    wave.switchStates.S1State = dFALSE;
    wave.switchStates.S2State = dTRUE;
    wave.switchStates.S3State = dFALSE;
    wave.switchStates.S4State = dFALSE;
    wave.switchStates.S5State = dTRUE;
    wave.switchStates.S6State = dFALSE;
    break;

case eS5_S4:
    wave.switchStates.S1State = dFALSE;
    wave.switchStates.S2State = dFALSE;
    wave.switchStates.S3State = dFALSE;
    wave.switchStates.S4State = dTRUE;
    wave.switchStates.S5State = dTRUE;
    wave.switchStates.S6State = dFALSE;
    break;

case eDISABLE_ALL_SWITCHES:
default:
    wave.switchStates.S1State = dFALSE;
    wave.switchStates.S2State = dFALSE;
    wave.switchStates.S3State = dFALSE;
    wave.switchStates.S4State = dFALSE;
    wave.switchStates.S5State = dFALSE;
    wave.switchStates.S6State = dFALSE;
}
}

#if dOPEN_LOOP_SWITCHING_MODE_1 == dTRUE
/*****
/** @brief
 * @param
 * @retval
*****/
// O modo de acionamento 1 gera ondas quadradas somente
static void WaveGenerator_SwitchingMode1(void)
{

```

```

// Se o chegou na contagem de tempo minima de fases acionadas
if (wave.timeCount >= wave.currentVelocity.phaseTime)
{
    // Seta todas as chaves para desligado
    WaveGenerator_SwitchesCmd(eDISABLE_ALL_SWITCHES);

    // Se chegou na contagem do Dead-Time
    if (++wave.deadTime >= dCONFIG_DEAD_TIME)
    {
        // Zera o contador de tempo de fase
        wave.timeCount = 0;
        // Zera o contador do Dead-time
        wave.deadTime = 0;
        // Vai para a proxima fase ou volta para a primeira, se ja esta na
        ultima
        if (++wave.switchSelect > eS5_S4)
        {
            // Antes de iniciar uma nova rotacao angular, calcula as variaveis
            de velocidade
            WaveGenerator_WaveCalculate();
            // Volta para o primeiro grupo de acionamento das chaves
            wave.switchSelect = eS1_S4;
        }
    }
}
else
{
    // Seta o estado das chaves conforme necessario
    WaveGenerator_SwitchesCmd(wave.switchSelect);
}

// Escreve nos pinos de cada chave
WaveGenerator_SetPin(&wave.switchStates);
}
#endif

#if dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE
static void WaveGenerator_SwitchingMode2(void)
{
}
#endif

#if dOPEN_LOOP_SWITCHING_MODE_3 == dTRUE
static void WaveGenerator_SwitchingMode3(void)
{
}
#endif
#endif

```

```

#if dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
static void WaveGenerator_SwitchingMode4(void)
{
}
#endif

#if dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE || dOPEN_LOOP_SWITCHING_MODE_3 ==
dTRUE || dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
static uint8_t WaveGenerator_WaveformHandler(switchSelect_t switchSelected ,
enabledChunks_t chunkControl)
{
uint8_t pwm;

switch (wave.currentChunk)
{
case eHALF_SQUARE_UP:
pwm = 45;
break;

case eFULL_SQUARE:
pwm = 90;
break;

case eHALF_SQUARE_DOWN:
pwm = 45;
break;
}

return dFALSE;
}
#endif
/* Public functions ----- */
/* ***** */
/** @brief
 * @param
 * @retval
 * ***** */
void WaveGenerator_ISR(void)
{
wave.timeCount++;

switch (wave.motorControlCurrentState)
{
// Mantem todas chaves desligadas
case eSTOP:
timing_ResetTick(&wave.waveTimer);

```



```

    break;

    // Prepara a posicao inicial do rotor
    case ePOSITION_DETECTION:
#if dFORCED_ALIGNMENT == dTRUE
    // Forca o alinhamento acionando duas bobinas

    // Aciona as fases A+ e B-
    WaveGenerator_SwitchesCmd(eS1_S4);
    WaveGenerator_SetPin(&wave.switchStates);

    // Se passou o tempo determinado para o alinhamento
    if (timing_GetTickStatus(&wave.waveTimer) == dTRUE)
    {
        // Reseta o timer
        timing_ResetTick(&wave.waveTimer);

        // Desliga as chaves
        WaveGenerator_SwitchesCmd(eDISABLE_ALL_SWITCHES);
        WaveGenerator_SetPin(&wave.switchStates);

        // Inicia a aceleracao em open loop
        wave.motorControlCurrentState = eOPEN_LOOP_ACCELERATE;
    }
#else

#endif

    break;

    case eOPEN_LOOP_ACCELERATE:
#if dOPEN_LOOP_SWITCHING_MODE_1 == dTRUE
    WaveGenerator_SwitchingMode1();
#elif dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE
    WaveGenerator_SwitchingMode2();
#elif dOPEN_LOOP_SWITCHING_MODE_3 == dTRUE
    WaveGenerator_SwitchingMode3();
#elif dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
    WaveGenerator_SwitchingMode4();
#endif
    break;

    case eOPEN_LOOP_RUNNING:
#if dOPEN_LOOP_SWITCHING_MODE_1 == dTRUE
    WaveGenerator_SwitchingMode1();
#elif dOPEN_LOOP_SWITCHING_MODE_2 == dTRUE
    WaveGenerator_SwitchingMode2();

```

```

#elif dOPEN_LOOP_SWITCHING_MODE_3 == dTRUE
    WaveGenerator_SwitchingMode3 ();
#elif dOPEN_LOOP_SWITCHING_MODE_4 == dTRUE
    WaveGenerator_SwitchingMode4 ();
#endif
    break;

    case eLOCKED_LOOP_RUNNING:
#if dCLOSED_LOOP_SWITCHING_MODE_1 == dTRUE
    WaveGenerator_SwitchingMode1 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_2 == dTRUE
    WaveGenerator_SwitchingMode2 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_3 == dTRUE
    WaveGenerator_SwitchingMode3 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_4 == dTRUE
    WaveGenerator_SwitchingMode4 ();
#endif
    break;

    case eLOCKED_LOOP_ACCELERATE:
#if dCLOSED_LOOP_SWITCHING_MODE_1 == dTRUE
    WaveGenerator_SwitchingMode1 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_2 == dTRUE
    WaveGenerator_SwitchingMode2 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_3 == dTRUE
    WaveGenerator_SwitchingMode3 ();
#elif dCLOSED_LOOP_SWITCHING_MODE_4 == dTRUE
    WaveGenerator_SwitchingMode4 ();
#endif
    break;
}
}

/*****
/** @brief
 * @param
 * @retval
*****/

/*****
static void WaveGenerator_WaveCalculate(void)
{
    switch (wave.motorControlCurrentState)
    {
    case eOPEN_LOOP_ACCELERATE:
        // Se passou da velocidade limite para entrar em closed loop, indica um
        erro de rotacao

```

```

    if (wave.currentVelocity.operationRPM > dVELOCITY_ERROR_RPM && wave.isBEMFDetected == dFALSE)
    {
        // To-do setar erro de rotacao
        wave.motorControlCurrentState = eSTOP;
    }
    // Se conseguiu detectar a posicao do rotor pelo BEMF
    else if (wave.isBEMFDetected == dTRUE)
    {
        // Acelera agora em closed-loop
        wave.motorControlCurrentState = eLOCKED_LOOP_ACCELERATE;
    }
    // Se nao, continua acelerando em open-loop
    else
    {
        // Incrementa a velocidade de aceleracao
        wave.currentVelocity.operationRPM += dACCELERATE_RPM_STEP;
        WaveGenerator_CalculaTime(&wave.currentVelocity);
    }
    break;

case eLOCKED_LOOP_ACCELERATE:

    break;
}
}

/*****
/** @brief   Inicializa a modulo de controle do motor.
 * @param   Nenhum.
 * @retval  Nenhum.
*****/

/*****
void WaveGenerator_Init(void)
{
    // Inicializa as variaveis do modulo
    wave.motorControlCurrentState = eSTOP;

    // Cria o timer de uso geral do modulo
    timing_CreateTimer(&wave.waveTimer, 10);
    timing_EnableDisableTimer(&wave.waveTimer, eTIM_ENABLE);
}

/*****
/** @brief   Configura a velocidade de rotacao alvo

```

```

* @param   velocityRPM: velocidade (em RPM) que o motor deve atingir
* @retval  Nenhum.

/***** */
void WaveGenerator_SetRPM(uint16_t velocityRPM)
{
    // Salva a velocidade em RPM desejada
    wave.targetVelocity.operationRPM = velocityRPM;
    // Calcula as demais velocidades para referencia
    WaveGenerator_CalculeTime(&wave.targetVelocity);
}

/***** */
/** @brief   Configura o estado da deteçao do BEMF
 * @param   BEMFState: true -> se conseguiu detectar o BEMF e pode
            entrar em closed loop
 *           false -> se perdeu a deteçao do BEMF e deve acionar em
            open loop
 * @retval  Nenhum

/***** */
void WaveGenerator_SetBEMFDetection(uint8_t BEMFState)
{
    wave.isBEMFDetected = BEMFState;
}

/***** */
/** @brief   Determina qual terminal do motor deve ser lido para detectar
            o BEMF, dependendo das chaves acionadas
 * @param   Nenhum.
 * @retval  Fase desligada, a qual pode ser lida

/***** */
motorPhases_t WaveGenerator_GetPhaseToDetect()
{
    motorPhases_t deactivatedPhase;

    switch (wave.switchSelect)
    {
        case eS3_S2:
        case eS1_S4:
            deactivatedPhase = ePHASE_C;
            break;

        case eS5_S4:

```

```

    case eS3_S6:
        deactivatedPhase = ePHASE_A;
        break;

    case eS1_S6:
    case eS5_S2:
        deactivatedPhase = ePHASE_B;
        break;
    }
    return deactivatedPhase;
}

/*****
/** @brief      Comando de start/stop para o acionamento do motor
 * @param      cmd: true  -> habilita o acionamento do motor.
 *             false -> desliga o acionamento do motor
 * @retval     Nenhum.
*****/

/*****
void WaveGenerator_Cmd(uint8_t cmd)
{
    if (cmd)
    {
        // Coloca o controle do motor na detecção de posição (ou alinhamento)
        wave.motorControlCurrentState = ePOSITION_DETECTION;
        // Configura a velocidade mínima de rotação possível
        wave.currentVelocity.operationRPM = 30000;
        // Calcula os tempos de acionamento para cada fase
        WaveGenerator_CalculeTime(&wave.currentVelocity);
    }
    else
    {
        // Coloca o controle do motor em stop
        wave.motorControlCurrentState = eSTOP;
    }
}

/*****
/** @brief      Calcula os tempos de acionamento para cada fase conforme a
 *              velocidade.
 * @param      *velocityType: endereço da variável que armazena todos os
 *              tipos de velocidade
 * @retval     Nenhum.
*****/

/*****

```

```
void WaveGenerator_CalculaTime(velocity_t *velocityType)
{
    // Converte a rotacao de RPM para Hz
    velocityType->operationFrequencyHz = ((float)((float)velocityType->
        operationRPM) / 60) * 1.0000;
    // Calcula quanto tempo cada par de chaves deve ficar acionado para esta
    // velocidade
    velocityType->phaseTime = ((d1_REVOLUTION_ANGLE / dANGLE_RESOLUTION) *
        dCARRIER_FREQUENCY) / velocityType->operationFrequencyHz;
    // Retira o dead time do tempo de acionamento de cada fase
    velocityType->phaseTime -= dCONFIG_DEAD_TIME;
}

/****** (C) COPYRIGHT Inova ***** END OF FILE*/
```

ANEXO A - DADOS TÉCNICOS DO PJD45N06

PJU45N06A / PJD45N06A

Electrical Characteristics ($T_A=25^\circ\text{C}$ unless otherwise noted)

PARAMETER	SYMBOL	TEST CONDITION	MIN.	TYP.	MAX.	UNITS
Static						
Drain-Source Breakdown Voltage	BV_{DSS}	$V_{GS}=0V, I_D=250\mu A$	60	-	-	V
Gate Threshold Voltage	$V_{GS(th)}$	$V_{DS}=V_{GS}, I_D=250\mu A$	1.0	1.7	2.5	V
Drain-Source On-State Resistance	$R_{DS(on)}$	$V_{GS}=10V, I_D=30A$	-	10.5	12	m Ω
		$V_{GS}=4.5V, I_D=15A$	-	12	15	
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS}=60V, V_{GS}=0V$	-	0.01	1.0	μA
Gate-Source Leakage Current	I_{GSS}	$V_{GS}=\pm 20V, V_{DS}=0V$	-	± 10	± 100	nA
Dynamic (Note 5)						
Total Gate Charge	Q_g	$V_{DS}=30V, I_D=10A,$ $V_{GS}=10V$ (Note 2,3)	-	39	-	nC
Gate-Source Charge	Q_{gs}		-	6.1	-	
Gate-Drain Charge	Q_{gd}		-	6.7	-	
Input Capacitance	C_{iss}	$V_{DS}=25V, V_{GS}=0V,$ $f=1.0\text{MHz}$	-	2256	-	pF
Output Capacitance	C_{oss}		-	145	-	
Reverse Transfer Capacitance	C_{rss}		-	93	-	
Turn-On Delay Time	$t_{d(on)}$		-	7.5	-	
Turn-On Rise Time	t_r	$V_{DD}=15V, I_D=10A,$ $V_{GS}=10V, R_G=6\Omega$ (Note 2,3)	-	36	-	ns
Turn-Off Delay Time	$t_{d(off)}$		-	49	-	
Turn-Off Fall Time	t_f		-	12	-	
Drain-Source Diode						
Maximum Continuous Drain-Source Diode Forward Current	I_S	---	-	-	45	A
Diode Forward Voltage	V_{SD}	$I_S=1A, V_{GS}=0V$	-	0.67	1.0	V

ANEXO B - DADOS TÉCNICOS DO ACS723



ACS723

High-Accuracy, Galvanically Isolated Current Sensor IC with Small Footprint SOIC8 Package

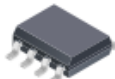
FEATURES AND BENEFITS

- Patented integrated digital temperature compensation circuitry allows for near closed loop accuracy over temperature in an open loop sensor
- UL60950-1 (ed. 2) certified
 - Dielectric Strength Voltage = 2.4 kVrms
 - Basic Isolation Working Voltage = 420 Vpk/297 Vrms
- Industry-leading noise performance with greatly improved bandwidth through proprietary amplifier and filter design techniques
- Pin-selectable bandwidth: 80 kHz for high bandwidth applications or 20 kHz for low-noise performance
- 0.65 mΩ primary conductor resistance for low power loss and high inrush current withstand capability
- Small footprint, low-profile SOIC8 package suitable for space-constrained applications
- Integrated shield virtually eliminates capacitive coupling from current conductor to die, greatly suppressing output noise due to high dv/dt transients
- 4.5 to 5.5 V, single supply operation
- Output voltage proportional to AC or DC current
- Factory-trimmed sensitivity and quiescent output voltage for improved accuracy

Continued on the next page...



PACKAGE: 8-pin SOIC (suffix LC)



Not to scale

DESCRIPTION

The Allegro™ ACS723 current sensor IC is an economical and precise solution for AC or DC current sensing in industrial, commercial, and communications systems. The small package is ideal for space constrained applications while also saving costs due to reduced board area. Typical applications include motor control, load detection and management, switched-mode power supplies, and overcurrent fault protection.

The device consists of a precise, low-offset, linear Hall sensor circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which is sensed by the integrated Hall IC and converted into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic field to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which includes Allegro's patented digital temperature compensation, resulting in extremely accurate performance over temperature. The output of the device has a positive slope when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sensing. The internal resistance of this conductive path is 0.65 mΩ typical, providing low power loss.

The terminals of the conductive path are electrically isolated from the sensor leads (pins 5 through 8). This allows the ACS723 current sensor IC to be used in high-side current sense applications without the use of high-side differential amplifiers or other costly isolation techniques.

Continued on the next page...

