

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

WILLIAN AMERICANO LOPES

**PROJETO E IMPLEMENTAÇÃO DE ROBÔ AUTÔNOMO SEGUIDOR
DE LINHA**

TRABALHO DE CONCLUSÃO DE CURSO

PATO BRANCO

2016

WILLIAN AMERICANO LOPES

**PROJETO E IMPLEMENTAÇÃO DE ROBÔ AUTÔNOMO SEGUIDOR
DE LINHA**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 1, do Curso Superior de Engenharia de Computação do Departamento Acadêmico de Informática - DAINF - da Universidade Tecnológica Federal do Paraná - UTFPR, como requisito parcial para obtenção do título de “Engenheiro de Computação em Computação”.

Orientador: Prof. Dr. Fábio Favarim
Co-orientador: Prof. Dr. Eng. César Rafael
Claire Torrico

PATO BRANCO

2016

LISTA DE FIGURAS

FIGURA 1	– Veículo exploratório Sojourner	10
FIGURA 2	– Robô terrestre L3S	11
FIGURA 3	– Descrição de um sistema de controle	11
FIGURA 4	– Configurações de sistemas de controle: (a) Malha aberta; (b) Malha fechada.	12
FIGURA 5	– Diagrama de blocos de um controlador automático	13
FIGURA 6	– Autômato de Moore	17
FIGURA 7	– Autômato de Mealy	17
FIGURA 8	– Sensores inerciais: (a) Sensor óptico; (b) Giroscópio; (c) Acelerômetro. ..	21
FIGURA 9	– Intersecções no percurso	22
FIGURA 10	– Área de partida-chegada	22
FIGURA 11	– Marcações de sinalização de curvatura	23
FIGURA 12	– Robôs seguidores de linha: (a) Robô <i>Alpha Project</i> (PETRY, 2016); (b) Pololu 3pi modificado (PETRY, 2016); (c) Protótipo de Guadagnin (2014); (d) Cartisx04 de Hirai (2014).	27
FIGURA 14	– <i>Kit</i> de desenvolvimento STM32-F303K8	28
FIGURA 15	– Motor HPCB 3041	29
FIGURA 17	– Ciclo da prototipação	32

LISTA DE SIGLAS

UTFPR	Universidade Tecnológica Federal do Paraná
PID	Proporcional Integral Derivativo
FPU	<i>Floating-Point Unit</i>
RUR	<i>Rossum's Universal Robots</i>
RIA	<i>Robotic Industries Association</i>
NASA	<i>National Aeronautics and Space Administration</i>
LS3	<i>Legged Squad Support Systems</i>
P	Proporcional
PI	Proporcional-Integral
PD	Proporcional-Derivativo
PID	Proporcional-Integral-Derivativo
SED	Sistema a Eventos Discretos
AFD	Autômato Finito Determinístico
NiCad	Níquel-Cádmio
NiMH	Hidreto Metálico de Níquel
CC	Corrente Contínua
CA	Corrente Alternada
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
MDF	<i>Medium-Density Fiberboard</i>
UFRJ	Universidade Federal do Rio de Janeiro
SRAM	<i>Static Random Access Memory</i>
DMPIS	<i>Dhrystone Million Instructions per Second</i>
HPCB	<i>High-Power Carbon Brush</i>
RPM	<i>Revolutions Per Minute</i>
PWM	<i>Pulse Width Modulation</i>
LED	<i>Light Emitter Diode</i>
PCB	<i>Printed Circuit Board</i>
UART	<i>Universal Asynchronous Receiver Transmitter</i>
Li-Po	Lítio-Polímero
EDA	<i>Electronic Design Automation</i>
PCB	<i>Printed Circuit Board</i>
TCC	Trabalho de Conclusão de Curso

SUMÁRIO

1	INTRODUÇÃO	6
1.1	CONSIDERAÇÕES INICIAIS	6
1.2	JUSTIFICATIVA	7
1.3	LIMITAÇÕES DO TRABALHO	8
1.4	OBJETIVOS	8
1.4.1	Objetivo geral	8
1.4.2	Objetivos específicos	8
2	REFERENCIAL TEÓRICO	9
2.1	ROBÓTICA	9
2.1.1	Robótica móvel	10
2.2	SISTEMAS DE CONTROLE PARA ROBÔS MÓVEIS	11
2.2.1	Ações de controle	12
2.2.1.1	Controlador <i>on-off</i>	13
2.2.1.2	Controlador Proporcional	14
2.2.1.3	Controlador Integral	14
2.2.1.4	Controlador Proporcional-Integral (PI)	14
2.2.1.5	Controlador Proporcional-Derivativo (PD)	14
2.2.1.6	Controlador Proporcional-Integral-Derivativo (PID)	15
2.3	SISTEMAS A EVENTOS DISCRETOS	15
2.3.1	Linguagens e Autômatos	16
2.3.1.1	Autômato de Moore	16
2.3.1.2	Autômato de Mealy	17
2.4	CONTROLE HÍBRIDO	18
2.5	ESTRUTURA DE UM ROBÔ MÓVEL	18
2.5.0.1	Sensores não-visuais	20
2.6	REGRAS DA ROBOCORE PARA ROBÔS SEGUIDORES DE LINHA	20
2.6.1	Especificação dos robôs	21
2.6.2	Especificações do Percurso	22
2.7	TRABALHOS RELACIONADOS	23
2.7.1	Robôs seguidores de linha	24
3	MATERIAIS	28
3.1	MICROCONTROLADOR	28
3.2	MOTORES CC	28
3.3	PONTE H	29
3.4	ENCODER MAGNÉTICO	29

3.5	SENSORES DE REFLETÂNCIA	30
3.6	PLACA DE CIRCUITO IMPRESSO	30
3.7	MÓDULO BLUETOOTH	30
3.8	RODAS	30
3.9	BATERIA LIPO	30
3.10	CONVERSOR STEP-UP	31
3.11	ESFERA DESLIZANTE	31
4	METODOLOGIA	32
5	CRONOGRAMA PRELIMINAR	34
	REFERÊNCIAS	35

1 INTRODUÇÃO

Este capítulo está dividido da seguinte forma: A Seção 1.1 apresenta uma visão geral do tema abordado, a Seção 1.2 trata da justificativa desta pesquisa, a Seção 1.3 trata as limitações do projeto e a Seção 1.4 apresenta os objetivos a serem alcançados neste trabalho.

1.1 CONSIDERAÇÕES INICIAIS

A robótica é uma das áreas mais promissoras da engenharia, tendo aplicabilidade em várias áreas: de médicas a aeroespaciais, buscando oferecer produtividade e flexibilidade à sua aplicação. Atualmente é difícil encontrar atividades industriais que não possuam um sistema robótico ou automatizado, seja total ou parcial.

Devido à ampla aplicabilidade e utilidade que os robôs apresentam foram criadas competições, que visam estimular e contribuir com a pesquisa na robótica, tais como:

- A Robogames (2016) que também é conhecida como “Olimpíada dos robôs”, em que são disputadas mais de cinquenta categorias;
- A VEX Worlds (2016), a maior competição de robótica do mundo, que contou com 1075 times e mais de 15.000 participantes em sua última edição (RECORDS, 2016);
- A RoboCup (2016), que em 2016 foi sediada em Leipzig, Alemanha;
- A Robocore (2016a), com eventos realizados no Brasil.

O WinterChallenge, realizado anualmente pela Robocore, em São Paulo, é um dos maiores eventos de robótica da América Latina, contando com a participação de vários países e teve mais de mil competidores e cerca de quinhentos robôs na edição de 2016 (MAUÁ, 2016).

Uma das categorias disputadas é a dos seguidores de linha, na qual os robôs devem seguir, de maneira autônoma, um trajeto que é determinado por uma linha. Nessa categoria, se destacam os competidores do Japão, na competição Robogames, e do México, na Robocore, estes obtendo os três primeiros lugares na competição WinterChallenge na categoria Seguidor de Linha - Pro (ROBOCORE, 2016b). Competindo nesta mesma categoria, a equipe Patobots, da Universidade Tecnológica Federal do Paraná (UTFPR) - Câmpus Pato Branco, conquistou o 5º e 6º lugar, com os robôs *Alpha project* e *Robbie 3.0*, respectivamente.

Com base nesse contexto, este trabalho propõe a construção de um protótipo de um robô seguidor de linha e o seu respectivo controle, visando participar de competições de robótica, como a Robocore.

O desenvolvimento deste trabalho contribuirá com a pesquisa que é feita na Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco, na área de robótica móvel, em que já foram produzidos os trabalhos de Guadagnin (2014) e Petry (2016), os quais servem de base para este trabalho.

1.2 JUSTIFICATIVA

O trabalho de Guadagnin (2014), utilizou-se de controle híbrido, o qual integrou dinâmicas de controle discreto, como detecção de marcas laterais e a faixa central na pista, e contínuo, como o controlador Proporcional Integral Derivativo (PID). Segundo o autor, o robô funcionou de acordo com o esperado para um percurso dentro das normas da Robocore, tendo alguns problemas relacionados à detecção das marcas laterais quando a pista estava com uma inclinação maior que 5° (graus).

O trabalho de Petry (2016) desenvolveu um robô híbrido, realizando um estudo sobre os resultados obtidos por controladores PID e *Fuzzy* (lógica difusa), sendo que o controlador PID apresentou melhor desempenho. Segundo o autor, não foi obtido sucesso considerável com a técnica *Fuzzy*, que necessitava de um processador com poder computacional maior do qual foi utilizado, com memória suficiente para implementar as variáveis de controle. Devido às dificuldades encontradas, a comparação dos métodos foi realizada no robô 3pi, da Pololu, em que o autor conseguiu uma velocidade de 1 m/s em retas.

Comparando os sistemas de controle desenvolvidos verificou-se que o *Fuzzy* necessita de um processador bem mais eficiente comparado ao PID, com memória suficiente para implementar as variáveis de controle, além de maior poder computacional para processamento matemático, possibilitando o sistema de controle compensar o erro em tempo hábil.

Com base nos trabalhos de Guadagnin (2014) e Petry (2016), é proposta a modelagem de um novo *hardware*, utilizando-se de técnicas de controle híbrido, o qual combina dinâmicas discretas (orientadas a eventos) e contínuas (orientadas a tempo) (CASSANDRAS; LAFORTUNE, 2008). Também propõe-se a utilização de um microcontrolador com Unidade de Ponto-Flutuante (FPU), que pode facilitar a implementação de técnicas mais complexas e que exigem maior poder computacional e memória.

1.3 LIMITAÇÕES DO TRABALHO

Este trabalho apresenta as seguintes limitações:

1. Devido à complexidade que é o projeto e a implementação de um robô, este trabalho não se preocupará com o desenvolvimento mecânico do dispositivo, sendo que a estrutura mecânica da mesma será confeccionada sobre uma placa de circuito impresso;
2. Devido à dificuldade em encontrar e adquirir peças de alto desempenho, poderão ser utilizadas peças de menor custo, as quais podem reduzir as capacidades do robô;
3. O projeto do robô seguidor de linha se aterá ao funcionamento em pistas que seguem as normas da Robocore, podendo apresentar restrições de comportamento e até mesmo não funcionar, caso a pista não esteja no padrão estabelecido.

1.4 OBJETIVOS

1.4.1 OBJETIVO GERAL

Projetar e implementar um protótipo de um robô seguidor de linha, com velocidade máxima próxima a 2 m/s, que seja autônomo, através da utilização de controle híbrido, aperfeiçoando as técnicas desenvolvidas por Petry (2016).

1.4.2 OBJETIVOS ESPECÍFICOS

- Projetar e confeccionar a estrutura do protótipo, visando atender as dimensões especificadas pela Robocore;
- Projetar o condicionamento de sinais para os dispositivos a serem utilizados, permitindo uma boa precisão na leitura dos sensores;
- Implementar o controlador PID, de modo a obter um controlador robusto e estável;
- Realizar testes com o protótipo em pistas que sigam as normas da Robocore;
- Implementar um sistema de telemetria, visando obter informações em tempo real do robô;
- Comparar os resultados obtidos com o de Petry (2016).

2 REFERENCIAL TEÓRICO

Neste capítulo são apresentados os principais conceitos teóricos relacionados ao desenvolvimento deste trabalho.

2.1 ROBÓTICA

A palavra “Robô” foi criada pelo escritor tcheco Karel Capek, sendo utilizada em sua peça *Rossum's Universal Robots* (RUR), a qual foi encenada em 1921, em Praga. Na linguagem eslava, *robota* significa atividade forçada ou escrava. Na peça de Capek, os robôs eram pessoas fabricadas artificialmente, ausentes de emoção (POLONSKII, 1996).

A *Robotic Industries Association* (RIA), entidade norte-americana responsável pela indústria de robótica, define um robô industrial como *um manipulador multipropósito reprogramável, controlado automaticamente, programado em três ou mais eixos, os quais podem ser fixos em um lugar ou móveis para aplicações em automação industrial* (ANSI/RIA.R15.06-1999, 2010). Diferentemente do que foi imaginado por Capek, são máquinas que realizam o trabalho.

Secchi (2012) classifica os robôs em três tipos:

- Industriais: São formados por estruturas mecânicas articuladas, as quais se movem pelas ordens de um sistema de controle, normalmente um microcontrolador;
- Médicos: Também conhecidos como de cooperação ou reabilitação, são os utilizados em cirurgias de alta complexidade e precisão, assim como as próteses inteligentes, que visam manter a aparência e funcionalidade do membro de pessoas com deficiência.;
- Móveis: São plataformas mecânicas, que se locomovem através de um certo ambiente e apresentam certa autonomia. São empregados principalmente em tarefas em que se tem risco à vida humana, como na manutenção de reatores nucleares ou exploração de minérios, mas também podem ser aplicados na agricultura e no transporte de cargas.

2.1.1 ROBÓTICA MÓVEL

Mesmo que os robôs industriais apresentem alta precisão e velocidade, estes possuem uma grande desvantagem, que é a falta de mobilidade. Algumas atividades não seriam realizadas sem a utilização destes dispositivos, como a *Mars Pathfinder*, missão exploratória da NASA para o reconhecimento da atmosfera de Marte. O veículo Sojourner (Figura 1) que foi utilizado nesta missão, explorou o território marciano por oitenta e três dias, tirando fotografias e realizando medições do ambiente (NASA, 1997b).



Figura 1 – Veículo exploratório Sojourner
Fonte: (NASA, 1997a)

A robótica móvel lida com o controle de veículos autônomos e semi-autônomos, tendo ênfase em problemas relacionados com o espaço em larga escala, que são regiões com espaços consideravelmente maiores que as observáveis pelo ponto de visão do robô. O espaço em larga escala é de extrema importância para um robô móvel, visto que afeta o seu movimento, compreensão e raciocínio nesta área, sendo estes três subproblemas essenciais para este campo de pesquisa (DUDEK; JENKIN, 2010).

Siegwart et al. (2011) classifica os robôs móveis em duas categorias relacionadas à locomoção:

- Robôs terrestres (*legged robots*): Tem como vantagem a manipulação de objetos e a locomoção em terrenos acidentados, mas tem alta complexidade mecânica e energética. A Figura 2 mostra o *Legged Squad Support Systems* (LS3) da Boston Dynamics, projetado para atuar nos mesmos terrenos acidentados utilizados por *marines* e soldados norte-americanos, ajudando a carregar equipamentos (DYNAMICS, 2016).
- Robôs com rodas (*wheeled robots*): É o tipo de locomoção mais utilizado em robôs móveis e veículos. Normalmente o equilíbrio não é levado em consideração, visto que na maior parte dos projetos as rodas são consideradas em contato com o solo o tempo todo. O robô Sojourner da Figura 1 é um exemplo de robô com rodas.



Figura 2 – Robô terrestre L3S
Fonte: (DYNAMICS, 2016)

2.2 SISTEMAS DE CONTROLE PARA ROBÔS MÓVEIS

Para que o robô seja autônomo, é necessário que este apresente uma resposta desejada para as mais diversas situações. Para tanto se utiliza de um sistema de controle, que consiste em subsistemas e processos, conhecidos como plantas, dos quais se obtém uma saída com desempenho desejado para uma dada entrada (NISE, 2012). Na Figura 3 é mostrado o diagrama de blocos¹ de um sistema de controle em uma forma simplificada, em que aplicando-se uma entrada a este, obtêm-se uma resposta.

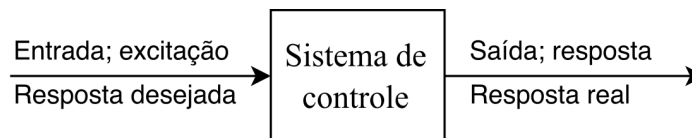


Figura 3 – Descrição de um sistema de controle
Fonte: Adaptado de (NISE, 2012, p.2).

Para Nise (2012) os sistemas de controle podem ser classificados em duas configurações principais:

- Controle por malha aberta (*Open loop control*): É a configuração mais simples, onde o sinal de saída não exerce nenhuma ação de controle no sistema (OGATA, 2010). Desta forma, um sistema de controle de malha aberta não é medido e nem comparado com a entrada, como pode ser visto na Figura 4a, não sendo capaz de compensar distúrbios que possam ser adicionados ao sistema (NISE, 2012).
- Controle por Malha fechada (*Closed loop control*): Nesta configuração, a saída ou a resposta influencia a entrada do sistema (ARAÚJO, 2007). Logo após a Resposta Desejada, na Figura 4b, tem-se um ponto de soma, em que será feita a soma algébrica

¹Diagrama de blocos é uma representação gráfica do sistema que mostra o fluxo de sinais que ocorre entre os componentes deste.

dos sinais associados. Segundo Ogata (2010), o sinal de erro atuante, que é a diferença entre o sinal de entrada e o sinal de realimentação (o qual é calculado logo após o ponto de soma, como pode ser visto na Figura 5), realimenta o controlador que tende a minimizar o erro e deixar a saída com o valor desejado.

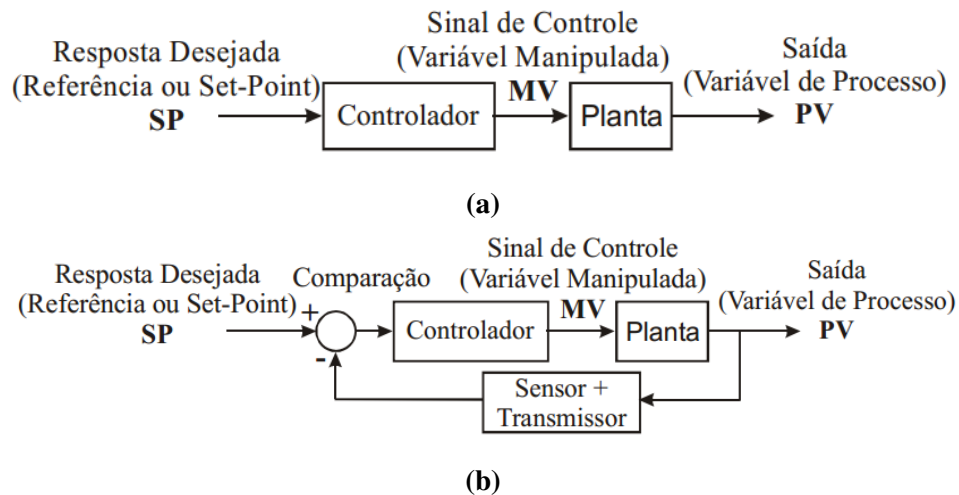


Figura 4 – Configurações de sistemas de controle: (a) Malha aberta; (b) Malha fechada.

(ARAÚJO, 2007)

2.2.1 AÇÕES DE CONTROLE

Conforme Ogata (2010), um controlador automático compara o valor de saída da planta com a entrada do sistema, determinando o desvio e produzindo um sinal de controle que reduzirá este desvio a um valor pequeno. É chamada de ação de controle a maneira pela qual o controlador produz o sinal de controle. Na Figura 5 é mostrado o diagrama de blocos de um sistema de controle, composto por um controlador automático, um atuador, uma planta e um sensor, o qual é o elemento de medição. Os atuadores são dispositivos capazes de alterar o estado do sistema controlado com base em sinais de controle, como os motores elétricos. O sensor é um elemento que converte a variável de saída em uma outra variável pertinente, que possa ser mensurável pelo sistema.

Ogata (2010) classifica os controladores industriais de acordo com as suas ações de controle:

- Controlador de duas posições ou *on-off*;
- Controlador Proporcional (P);
- Controlador Integral;

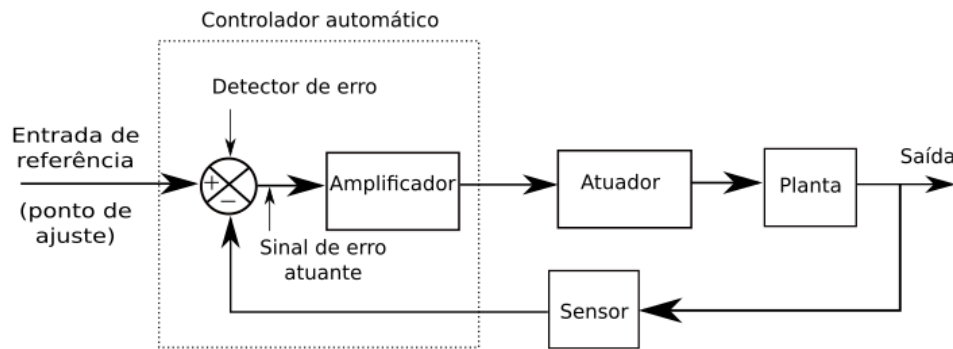


Figura 5 – Diagrama de blocos de um controlador automático

Fonte: Adaptado de (OGATA, 2010, p.18).

- Controlador Proporcional-Integral (PI);
- Controlador Proporcional-Derivativo (PD);
- Controlador Proporcional-Integral-Derivativo (PID)

2.2.1.1 CONTROLADOR *ON-OFF*

Em controladores de duas posições ou *on-off*, o elemento atuante tem somente duas posições, as quais são fixas, geralmente sendo *on* (ligado) e *off* (desligado) (OGATA, 2010). Tem grande aplicabilidade em sistemas de controle industriais e domésticos, devido ao baixo custo e simplicidade de implementação.

Considerando-se o sinal de saída do controlador $u(t)$ e o sinal de erro atuante $e(t)$, o sinal $u(t)$ apresenta um valor máximo ou mínimo, caso o erro atuante seja negativo ou positivo, respectivamente. Assim, tem-se que:

$$u(t) = U_1, \quad \text{para } e(t) > 0, \quad (1)$$

$$u(t) = U_2, \quad \text{para } e(t) < 0, \quad (2)$$

em que U_1 e U_2 são constantes. Em problemas de controle mais complexos, se faz necessária a utilização de outros controladores, devido ao *on-off* apresentar muitas oscilações e *offset*².

s defined as the voltage that must be applied between the two input terminals of the op amp to obtain zero volts at the output. Ideally the output of the op amp should be at zero volts when the inputs are grounded.

²*Offset* é um erro entre o valor desejado e o valor obtido.

2.2.1.2 CONTROLADOR PROPORCIONAL

Na ação de controle proporcional, a relação entre a saída do controlador $u(t)$ e o sinal de erro $e(t)$ é apenas um ganho:

$$u(t) = K_p e(t), \quad (3)$$

em que K_p é o ganho do controlador proporcional. Para Araújo (2007) quanto maior o ganho, menor será o erro $e(t)$, no entanto o sistema pode perder estabilidade devido ao aumento do tempo de acomodação³.

2.2.1.3 CONTROLADOR INTEGRAL

Segundo Ogata (2010), na ação de controle integral, o valor da saída $u(t)$ é alterado a uma taxa de variação proporcional ao sinal de erro atuante $e(t)$:

$$u(t) = K_i \int_0^t e(t) dt, \quad (4)$$

em que K_i é a constante de ganho integral.

2.2.1.4 CONTROLADOR PROPORCIONAL-INTEGRAL (PI)

Para Nise (2012), um controlador Proporcional-Integral pode ser chamado de compensador integral ideal, visto que as ações de controle Proporcional e Integral alimentam o erro e a integral do erro para a planta, respectivamente, desta forma tendendo a eliminar o erro em regime permanente⁴. A sua ação de controle é definida por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt, \quad (5)$$

em que T_i é chamado de tempo integrativo.

2.2.1.5 CONTROLADOR PROPORCIONAL-DERIVATIVO (PD)

Segundo Araújo (2007), as ações proporcional e derivativa deste controlador contribuem com o regime transitório, tendendo a aumentar a estabilidade do sistema e reduzir o tempo de

³Tempo de acomodação (*settling time*) é o tempo para que a curva de resposta alcance valores em uma faixa (2% a 5%) em relação ao valor final, permanecendo nesta faixa interminavelmente.

⁴A resposta temporal de um sistema consiste em duas partes: da resposta transitória e da resposta estacionária ou em regime permanente

acomodação. A ação de controle deste controlador é definida por:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (6)$$

em que T_d é chamado de tempo derivativo (OGATA, 2010).

2.2.1.6 CONTROLADOR PROPORCIONAL-INTEGRAL-DERIVATIVO (PID)

O controlador Proporcional-Integral-Derivativo reúne a ação das três ações de controle, atuando na melhoria do regime permanente e da resposta transitória. A equação do controlador é dada por:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (7)$$

em que K_p é o ganho proporcional, T_i é o tempo integrativo e T_d é o tempo derivativo (OGATA, 2010).

2.3 SISTEMAS A EVENTOS DISCRETOS

Para Cassandras e Lafortune (2008), os sistemas podem ser classificados em duas categorias quanto à natureza do espaço de estados selecionado:

- Estado contínuo: O espaço de estados X é uma continuidade, compreendendo todos os vetores n -dimensionais de números reais ou complexos, eventualmente;
- Estado discreto: O espaço de estados é um conjunto discreto. As variáveis de estado deslocam-se em pontos discretos no tempo, de um valor estado discreto para outro.

Teixeira (2013) diz que a modelagem computacional de um sistema pode ser estruturada em dois fundamentos:

- Estado, que determina o *status* do sistema em determinada situação;
- Transição de estados, o qual caracteriza o crescimento do sistema.

Alguns sistemas são mapeados continuamente e suas estruturas de transição são regidas pelo tempo, enquanto que em outros sistemas seus estados podem não ser contínuos e as transições não dependem do tempo, mas de eventos instantâneos e assíncronos, como o processamento de um dado em um equipamento computacional, em que o conjunto de estados

é discreto e as transições não são feitas por tempo, mas por eventos como *clock* e interrupções (TEIXEIRA, 2013).

Torrico (2003) define um Sistema a Eventos Discretos (SED) como um sistema dinâmico a estado discreto que evolui à proporção assíncrona de eventos. Deste modo, os SEDs tem como características o espaço de estados ser discreto e o mecanismo de transição de estados ser dirigido por eventos.

Alguns exemplos dos Sistemas a Eventos Discretos, conforme Cassandras e Lafortune (2008), são os Sistemas de Filas, os Sistemas de Computador, Sistemas de Comunicação e Sistemas de Manufatura.

2.3.1 LINGUAGENS E AUTÔMATOS

Uma das maneiras formais de se estudar o comportamento lógico de um SED é baseado nas teorias de linguagens e autômatos, que se dá pelo fato de qualquer Sistema a Eventos Discretos ter um conjunto implícito E associado a ele. Cassandras e Lafortune (2008) chama esse conjunto E e a sequência de eventos como “alfabeto” e “palavras” de linguagem, respectivamente. Para demonstrar estes conceitos, o autor fala do exemplo de uma máquina que é ligada uma ou duas vezes ao dia, como um carro ou um computador pessoal, e deseja-se projetar um sistema que realize uma simples tarefa: Quando a máquina estiver ligada, esta primeiramente deve indicar que está ligada e depois reportar este fato (como “chechar óleo”, caso seja um carro). Neste caso, cada um destes sinais define um evento, enquanto que todos os sinais possíveis que a máquina possa emitir são definidos como uma sequência de eventos, ou alfabeto.

Cassandras e Lafortune (2008) define um autômato como um dispositivo apropriado para representar a linguagem de acordo com regras bem definidas. Caso o conjunto de estados do autômato seja finito e caso este seja determinístico, ou seja, que este não possua duas ou mais transições com o mesmo nome saindo de um estado, o autômato é chamado de Autômato Finito Determinístico (AFD). Torrico (2003) diz que um autômato pode ser representado por uma tabela ou por um diagrama de transição de estados, sendo que neste os estados e as transições da máquina podem ser representados por vértices do diagrama e por arcos que conectam estes vértices, respectivamente. Duas variantes dos autômatos são os Autômatos de Mealy e de Moore, as quais serão clarificadas a seguir.

2.3.1.1 AUTÔMATO DE MOORE

O modelo de Moore são autômatos com saídas de estados, em que há uma função de saída a qual especifica uma saída para cada estado (CASSANDRAS; LAFORTUNE, 2008).

As ações são produzidas nos estados e a saída depende somente do estado atual. Um exemplo do Autômato de Moore pode ser visto na Figura 6, em que a saída associada a cada estado é mostrada em negrito (cada saída é acionada pelo autômato quando este entra no estado).

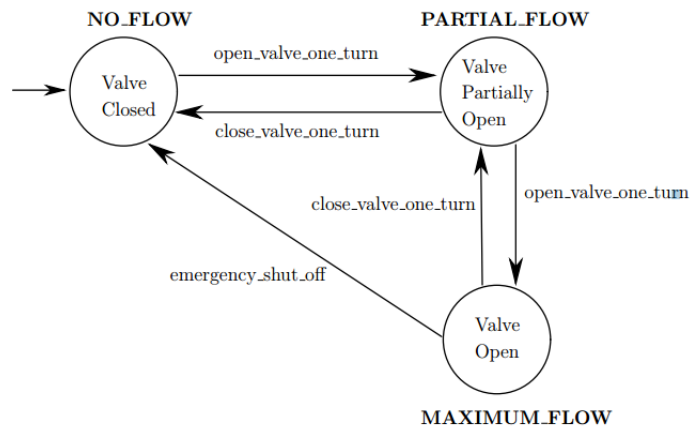


Figura 6 – Autômato de Moore

Fonte: (CASSANDRAS; LAFORTUNE, 2008, p.73)

2.3.1.2 AUTÔMATO DE MEALY

Diferentemente de Moore, o autômato de Mealy depende da entrada e da saída do estado, representando sequencialmente o crescimento dos estados. Conforme mostra a Figura 7, as funções de saída estão nas transições. Este modelo funciona da seguinte maneira para uma transição e_i/e_o do estado x para o estado y : Quando o sistema estiver no estado x , o autômato receberá o evento de entrada e_i e expedirá o evento de saída e_o durante a transição para y (CASSANDRAS; LAFORTUNE, 2008).

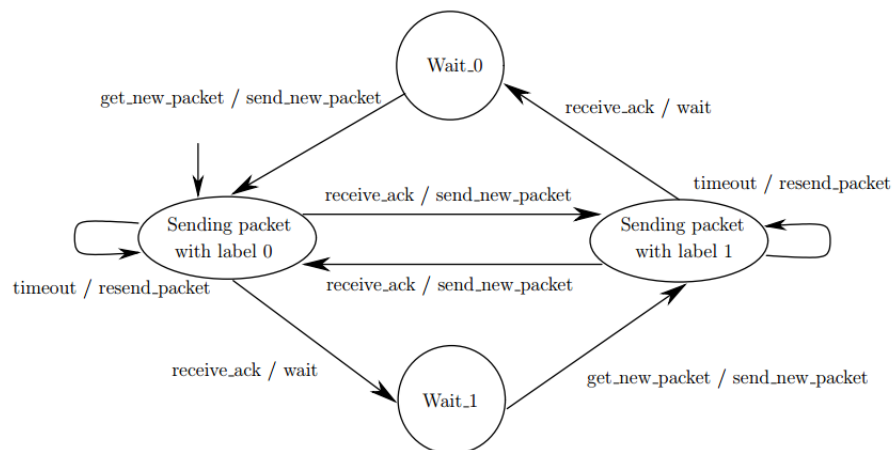


Figura 7 – Autômato de Mealy

Fonte: (CASSANDRAS; LAFORTUNE, 2008, p.73)

2.4 CONTROLE HÍBRIDO

Para ??), a palavra híbrido significa “Que ou o que é composto de elementos distintos ou disparatados”

Para o dicionário Merriam-Webster (2016), a palavra *hybrid* (híbrido para o português) significa “something that is formed by combining two or more things”, ou seja, é combinação de duas ou mais coisas. Aplicando este conceito aos sistemas, tem-se CONTINUAR DAQUI

2.5 ESTRUTURA DE UM ROBÔ MÓVEL

Dudek e Jenkin (2010) classifica os componentes físicos (*hardware*) dos robôs móveis em quatro partes:

- Locomoção: É como o robô se move pelo ambiente;
- Percepção: É como o robô percebe propriedades do ambiente e dele mesmo;
- Raciocínio: É como o robô transforma as suas medidas em ações;
- Comunicação: É como que o robô se comunica com um agente externo.

Com suporte nessa divisão e no restante da literatura, tem-se os seguintes elementos necessários para o projeto de um robô móvel:

Bateria: É um componente essencial do sistema, visto que o robô móvel precisa se locomover pelo espaço e não pode ser dependente de uma fonte externa de energia elétrica. Dudek e Jenkin (2010) diz que baterias com bom desempenho e baixo custo são as baseadas em células de gel (*gell cell*), mas por serem pesadas não são utilizadas com veículos leves. Para estes, são utilizadas tecnologias baseadas em Íons de Lítio (*lithium-ion*), Níquel-Cádmio (NiCad) e as de Hidreto Metálico de Níquel (NiMH), as quais são originalmente projetadas para serem utilizadas em dispositivos sem-fio, como *laptops* e *smartphones*.

Motores: São os responsáveis por fazer com que o veículo “ande”. Muitos robôs utilizam a bateria para acionar os motores de passo ou servomotores, os quais convertem energia elétrica em mecânica (Dudek e Jenkin (2010)). No entanto existem outros dispositivos, capazes de gerar energia mecânica em movimento, como os piezoelétricos e os pneumáticos.

- O motor de passo (*stepper motor*) é um motor elétrico em que o eixo move uma quantidade contralada a cada pulso, assim, controlando estes pulsos obtém-se a saída desejada (DUDEK; JENKIN, 2010).
- O servomotor (*servo motor*) combina um motor elétrico de corrente contínua (CC) ou corrente alternada (CA) com um sensor de orientação de eixo (DUDEK; JENKIN, 2010). O autor diz ainda que controle dos servomotores é mais complexo que os motores de passo, sendo necessário utilizar um controlador de motor mais complexo, os quais variam em complexidade e que geralmente trabalham com parâmetros como posição, velocidade ou movimento de saída.

Controle de um servo motor: Podem ser utilizadas as abordagens de malha aberta e malha fechada. No controle de malha aberta, o valor desejado do eixo é utilizado para projetar um controlador oportuno, no entanto fica suscetível a distúrbios, pois não apresenta realimentação. No controle por malha fechada, a realimentação do sistema é utilizada para ajustar o movimento do dispositivo, sendo que um controlador amplamente utilizado é o PID (DUDEK; JENKIN, 2010).

Dispositivo de comunicação: Devido à necessidade em interagir com um operador humano, como no caso de reportar que uma tarefa foi concluída, Dudek e Jenkin (2010) relata que os robôs utilizam de um meio físico para a comunicação, como por cabo, ou por comunicação sem-fio (*wireless*), como o *bluetooth* (padrão sem-fio de curto alcance), 802.11 (padrão da IEEE IEEE para redes locais sem-fio) ou infravermelho.

Processamento: O processamento é um componente principal dos robôs móveis, tanto que Dudek e Jenkin (2010) discute em que lugar o processamento do veículo deve estar. O autor chega na conclusão que a melhor forma do processamento ser feito é a de separar as tarefas em *on-board* (dentro do veículo) e *off-board* (fora do veículo), responsáveis, respectivamente, pelas tarefas de tempo crítico e as que não são de tempo crítico. Este modelo tem como vantagens a redução no consumo de energia do dispositivo.

Sensores: Secchi (2012) diz que os robôs devem ser capazes de realizar três tarefas essenciais: estimar a sua posição e orientação (*pose*), manter o mapa do ambiente atualizado e identificar os possíveis obstáculos deste. Desta forma, para que o robô conheça o seu ambiente de trabalho e possa se adaptar a este, é necessário coletar informações e adequá-las ao sistema de controle. Assim, o controle de malha fechada, o qual foi abordado anteriormente, necessariamente precisa do sinal de sensores para realizar o *feedback* (realimentação) do sistema.

Dudek e Jenkin (2010) classifica os sensores em duas variantes: visuais e não-visuais. A seguir serão mostrados os principais sensores não-visuais dos robôs móveis. Os sensores

visuais, devido à não terem aplicação neste trabalho, não serão abordados. Para mais informações sobre estes, consultar (DUDEK; JENKIN, 2010), (SIEGWART et al., 2011) e (SECCHI, 2012).

2.5.0.1 SENSORES NÃO-VISUAIS

Sensores inerciais São sensores externos que fazem referência com o mundo externo, ou seja, medem variáveis de posição do robô.

- **Encoder óptico:** Estes dispositivos medem a velocidade e a posição angular em um *driver* de motor ou no eixo de uma roda. Este sensor, segundo Siegwart et al. (2011), é um interruptor de luz, que produz uma quantidade de pulsos para cada revolução do eixo. Os pulsos gerados pela diferença de fase entre os canais A e B é utilizada determinar a direção de rotação, como pode ser visto na Figura 8a.
- **Giroscópios:** São sensores que preservam a orientação em relação a um referencial fixo, provendo a medida absoluta. Podem ser mecânicos ou ópticos. Na Figura 8b pode ser visto um giroscópio de dois eixos.
- **Acelerômetros:** Siegwart et al. (2011) diz que estes sensores são capazes de medir todas as forças externas que estejam agindo sobre ele, inclusive a gravidade. São baseados em sistemas pendulares, em que a primeira integração das acelerações proporciona a velocidade e a segunda, a posição (SECCHI, 2012). A precisão deste dispositivo é de grande importância, visto que pequenos erros podem influenciar na posição estimada, devido à dupla integração das acelerações. O diagrama deste dispositivo pode ser visto na Figura 8c. (MELHORAR PARTE DO ACELEROMETRO)

Sensores de infravermelho: São sensores de proximidade rápidos e baratos, que basicamente consistem na emissão de um pulso infravermelho e detecção do sinal refletido, com a distância aproximada calculada através da força do sinal (DUDEK; JENKIN, 2010).

2.6 REGRAS DA ROBOCORE PARA ROBÔS SEGUIDORES DE LINHA

Na Seção 2.6.1 e Seção 2.6.2 são apresentadas as regras relacionadas à especificação dos robôs e do percurso, respectivamente, para a categoria robô seguidor de linha Pro, em eventos realizados pela Robocore (2016c).

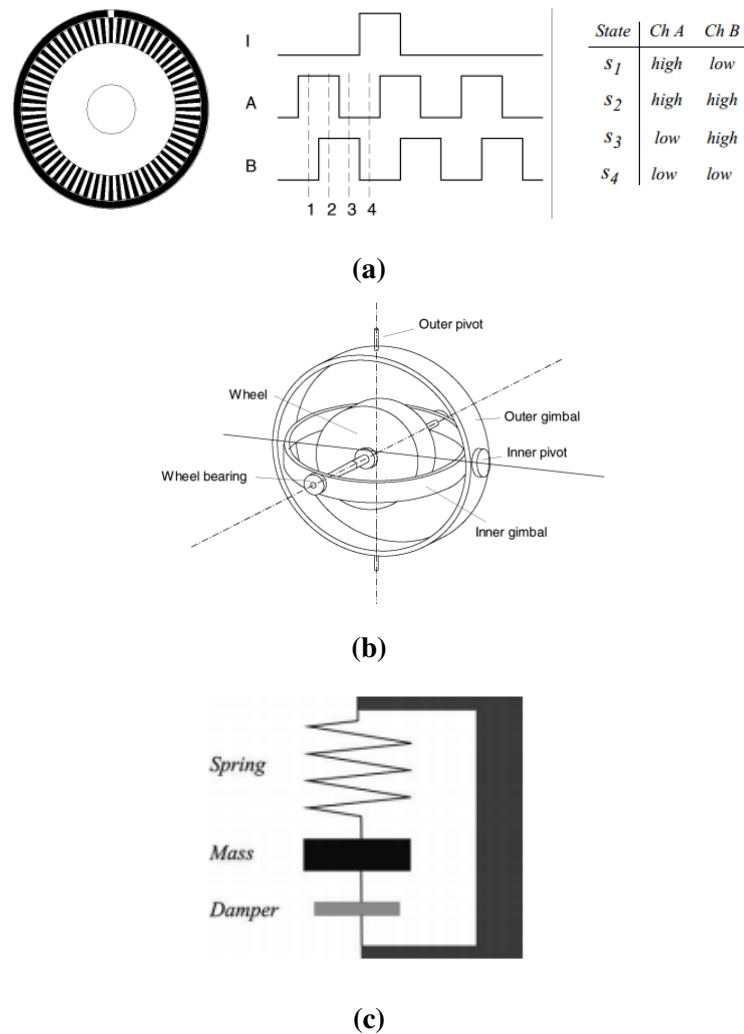


Figura 8 – Sensores inerciais: (a) Sensor óptico; (b) Giroscópio; (c) Acelerômetro.
(ARAÚJO, 2007)

2.6.1 ESPECIFICAÇÃO DOS ROBÔS

Para competir na categoria seguidor de linha, os robôs devem ser totalmente autônomos, não podendo ser controlados externamente por fio ou por rádio, com exceção para quando este for iniciado. Todos os componentes devem ser embarcados. A dimensão máxima permitida é de 250mm de comprimento, 250mm de largura e 200mm de altura. Não é permitido alterar as dimensões do robô durante a partida, assim como alterar o *hardware* ou *software* durante a tomada de tempo. Também não é permitida a utilização de mecanismo de sucção, que vise aumentar a força normal do robô em relação ao solo.

2.6.2 ESPECIFICAÇÕES DO PERCURSO

A pista é feita de uma ou mais placas de MDF revestidas com uma manta de borracha preta, assim, eventualmente serão necessárias emendas para compor a área do percurso. Os robôs, no entanto, devem ser capazes de superar os desníveis decorrentes das emendas, que são de aproximadamente 1mm. Uma linha branca, de $19\pm 1\text{mm}$, indica o percurso. Esta linha pode cruzar sobre ela mesma, tendo, neste caso, um ângulo de intersecção de $90\pm 5^\circ$ (graus), com os 250mm antes e depois do cruzamento sendo retas (conforme pode ser visto na Figura 9). O circuito é totalmente plano, porém podem ocorrer inclinações de até 5° .

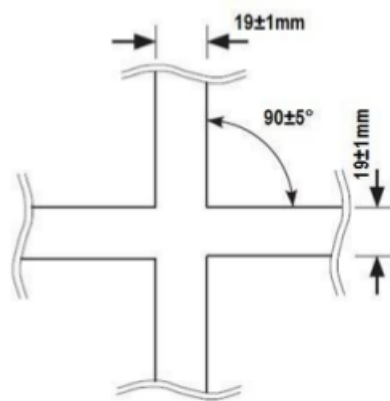


Figura 9 – Intersecções no percurso

Fonte: Disponível em (ROBOCORE, 2016c, p.4).

A área que se estende entre o ponto de partida e o ponto de chegada, considerando 200mm a direita da linha e 200mm a esquerda da linha é denominada “área de partida-chegada”, conforme pode ser visto na Figura 10.

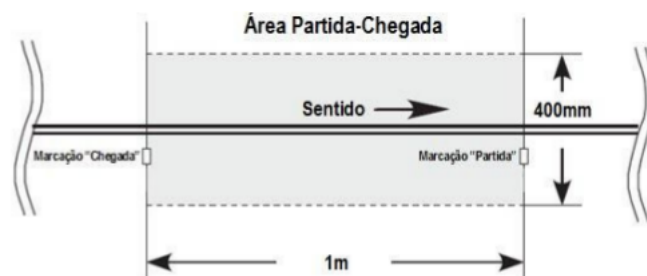


Figura 10 – Área de partida-chegada

Fonte: Disponível em (ROBOCORE, 2016c, p.4).

Quando houver um arco (intersecção entre a faixa branca), o raio deste é de pelo menos 100mm. Quando houver uma alteração na curvatura do percurso, deve haver uma marcação no lado esquerdo da linha, como pode ser visto na Figura 11.

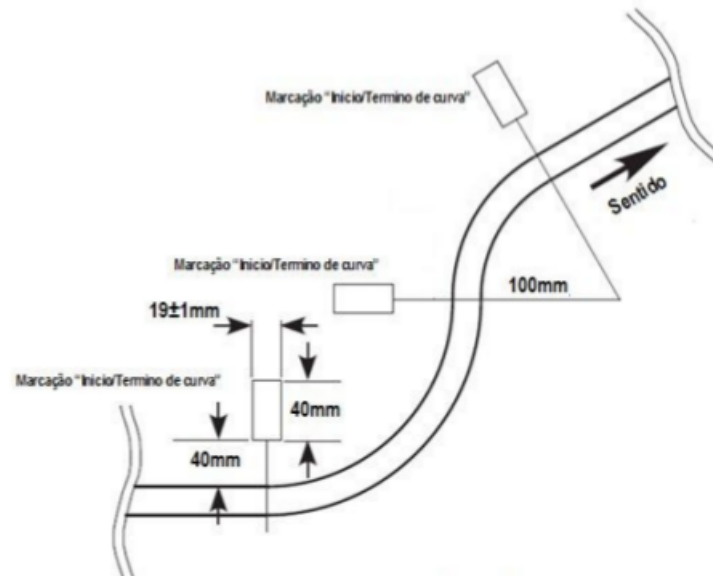


Figura 11 – Marcações de sinalização de curvatura

Fonte: Disponível em (ROBOCORE, 2016c, p.4).

2.7 TRABALHOS RELACIONADOS

Na revisão bibliográfica a seguir são mostradas técnicas desenvolvidas para o sistema de navegação de robôs móveis autônomos.

O objetivo do trabalho de Freitas (2016) foi o desenvolvimento de um sistema robótico autônomo em uma arquitetura híbrida de três camadas, composta pela implementação das camadas Planejador e Executivo. O estudo de caso foi feito pela sistema robótico DORIS, um robô guiado por trilhos, desenvolvido pela COPPE/UFRJ, em parceria com a Petrobras e a Statoil, para a inspeção de plataformas de petróleo. O veículo tem como funcionalidades a detecção de anomalias por vídeo, áudio, vibração, temperatura e por câmera de infravermelho e o mapeamento 3D do ambiente.

Heinen (2002) também propôs uma arquitetura de controle híbrida, integrando as técnicas de controle deliberativo e reativo em uma abordagem de três camadas: Deliberativa, a qual determina um trajeto até o objetivo e evita colisões com obstáculos já conhecidos; Vital, que utiliza de comportamentos reativos para guiar o robô ao seu objetivo e evitar colisões com obstáculos (sejam estes estáticos ou dinâmicos); e Funcional, que faz a integração das camadas vital e deliberativa, fornecendo parâmetros reativos para a camada vital. O sistema de controle proposto foi implementado no simulador SimRob3D, o qual permite a utilização de modelos de ambiente tridimensionais, assim como sensoriais e cinemáticos, permitindo que estes possam ser alterados em tempo real.

A pesquisa de Pessin (2013) se concentra na busca por soluções inteligentes aplicadas em robôs móveis autônomos visando a operação destes em ambientes dinâmicos. Através

da aplicação de aprendizado de máquina, buscou-se uma nova visão sobre a operação destes dispositivos em três desafios da área: navegação, localização e operações com grupos de robôs. No estudo das operações com grupos de robôs, foi utilizada uma aplicação de combate a incêndios, com a avaliação por quatro técnicas: Algoritmos Genéticos, Otimização por Enxame de Partículas, *Hill Climbing* e *Simulated Annealing*. Na investigação sobre a navegação, é apresentado o desenvolvimento de um veículo autônomo de grande porte, funcional para ambientes externos. Relacionado com a localização, é mostrado um método que provê informações de localização para os robôs com base em dados obtidos por redes sem fio.

A seguir serão abordados os trabalhos relacionados aos robôs seguidores de linha.

2.7.1 ROBÔS SEGUIDORES DE LINHA

Nesta Seção serão abordados os trabalhos de (GUADAGNIN, 2014), (PETRY, 2016) e (HIRAI, 2014):

- O trabalho de Guadagnin (2014) foi pioneiro na Universidade Tecnológica Federal do Paraná - câmpus Pato Branco, sendo o primeiro a ser feito sobre robôs seguidores de linha na instituição. Em seu trabalho foi implementado um sistema de controle híbrido, o qual o autor define como sendo um sistema no qual ocorre interação entre variáveis discretas e contínuas. Para o controle contínuo, que era o responsável pela posição do veículo, foi utilizado um controlador P (ação de controle proporcional), devido à planta já apresentar um integrador. O controle discreto, encarregado da detecção de marcas na pista, foi implementado por um autômato de Moore, composto por dez estados e quinze transições. Este controlador definia a velocidade do robô em 100%, para retas, em 75%, para curvas, e 0%, quando parado (em seu estado inicial). Com os controladores contínuos e discretos projetados, estes foram reunidos e implementados no microcontrolador. A detecção da pista foi feita por sensores de refletância.

Com o protótipo implementado, foram feitos testes em uma pista desenvolvida por Guadagnin (2014), a qual foi baseada em pistas da competição Robocore. O percurso consistia de quatro curvas e dois cruzamentos. Primeiramente foram feitos cinco testes, sendo que três destes o robô não reconheceu as marcas laterais devido à inclinação da pista não estar conforme a recomendada. Com a inclinação da pista corrigida, foram feitos oito testes, sendo que destes em somente um as marcas não foram detectadas. Também foi alterada a velocidade em curvas para 82%, sendo que nesta situação o veículo não conseguiu realizar a curva. O robô desenvolvido por Guadagnin (2014) pode ser visto na Figura 12a.

- O trabalho de Petry (2016) teve como base o de Guadagnin (2014), logo nota-se a presença de várias particularidades entre ambos, como o controle híbrido. Foi proposto para este projeto um *hardware* mais eficiente, com um novo sistema de sensoriamento, e a implementação da lógica *fuzzy*, também conhecida como lógica difusa, a qual foi integrada ao controlador discreto.

O sistema de controle discreto foi modelado por Moore, utilizando autômatos determinísticos de estados finitos. Este controlador é muito semelhante ao projetado por Guadagnin (2014), com alteração na velocidade em curvas, sendo utilizado 80%. O controlador *fuzzy* foi projetado com base no valor dos sensores de refletância, comandando a velocidade dos motores de acordo com a base de regras desenvolvida. O controlador contínuo escolhido foi o PID, que assim como o *fuzzy* também utiliza as informações dos sensores de refletância para gerar a ação de controle.

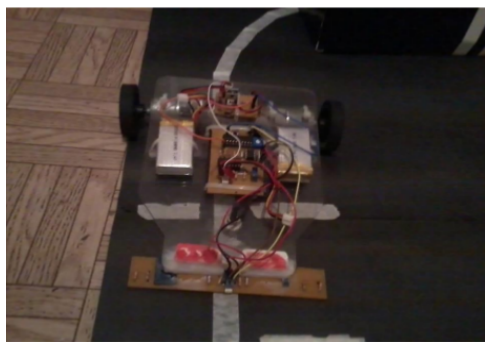
A estrutura do robô de Petry (2016) foi feita utilizando uma placa de fenolite, a partir da qual os componentes são soldados. Para a locomoção foram utilizados dois motores CC de 6V e 1000 RPM, da Pololu, duas rodas de pneus de silicone e um *driver* para motor CC TB6612FNG, da Toshiba. A alimentação foi feita por uma bateria Li-Po de 7,4V de tensão e 32,5A de corrente, com potência de 1300mAh. Foram utilizados os módulos reguladores de tensão XL6009 e MP2259, sendo *step-up* (aumenta a tensão) e *step-down* (diminui a tensão), respectivamente. O microcontrolador utilizado por Petry (2016) foi o MSP430G2553, o qual possui um ADC de oito canais, trabalha com *clock* de 16MHz, memória *flash* de 16KB e 0,5KB de SRAM. Os sensores de refletância utilizados foram os QTR-1A e QTR-8A (o qual é composto por oito unidades de QTR-1A), este empregado no protótipo.

Petry (2016) desenvolveu três protótipos, sendo que na Figura 12b pode ser visto o terceiro protótipo, o qual não funcionou devido à problemas com o circuito. Testes foram feitos com o segundo protótipo e segundo o autor, todas as marcações de curvas, cruzamentos e sinais (início e fim) foram reconhecidos. Também foi utilizado o robô 3pi, da Pololu, o qual pode ser visto na Figura 12c, para fazer a comparação entre os controladores PID e *fuzzy*. Este veículo foi modificado, com a substituição das rodas e dos motores originais pelos mencionados anteriormente. A bateria também foi alterada, sendo utilizadas duas baterias em série, fornecendo 7,4V de tensão de 12A de corrente. Devido à limitações no *hardware*, não foi utilizado o controle de eventos discretos na comparação dos controladores. O controlador PID obteve um melhor desempenho, conseguindo realizar 680 leituras e atuações nos motores por segundo, enquanto que o *fuzzy* conseguiu somente 385. O sistema *fuzzy* não foi capaz de seguir a linha, devido à instabilidade,

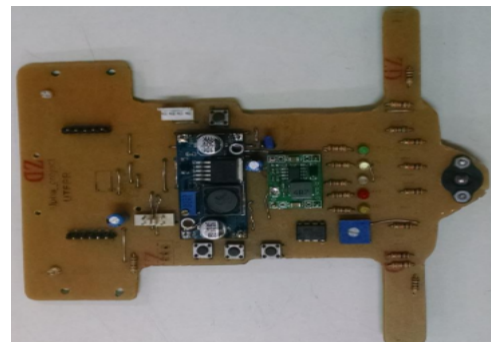
logo foram adicionadas quatro regras, as quais deixaram o robô mais estável mas com frequência de operação muito baixa.

- Hirai (2014) é um dos principais competidores na categoria dos robôs seguidores de linha, tendo sido campeão em várias competições na Ásia, especialmente no Japão, onde em 2015 conquistou a terceira vitória seguida na Robotrace (HIRAI, 2016). O Cartisx04, o qual pode ser visto na Figura ??, é um protótipo desenvolvido com base no Cartis03, sendo esse o seu último robô desenvolvido. Não se tem acesso ao *software* desenvolvido por Hirai (2014), no entanto é disponibilizado o *hardware* utilizado, logo é interessante conhecer como é feito o projeto de competidor de nível mundial.

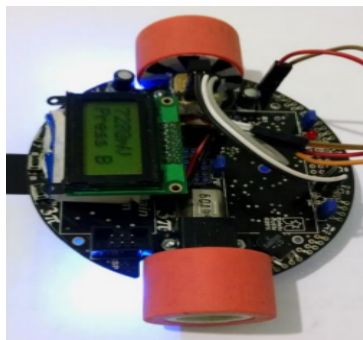
Segundo o autor, o CartisX04 é um robô leve de direção irregular de quatro rodas. São utilizados dois motores Maxon DCX10L, o qual é um motor CC *brushed*, e um motor de direção Maxon RE10. Os sensores utilizados são o giroscópio Invensense ISZ-650 e o sensor de refletância GP2S700, da Sharp. O microcontrolador utilizado foi o STM32F103RE, da STMicroelectronics, o qual é equipado com processador ARM-Cortex M3 de 32 bits, 72MHz de *clock*, 256KB de memória *flash* e 64KB de SRAM. As dimensões do veículo são 175mm de altura, 153mm de comprimento e 7mm de centro de gravidade. O robô pesa 98g, 32g a menos do que o protótipo anterior, o que para o autor propiciou aumento de desempenho.



(a)



(b)



(c)



(d)

Figura 12 – Robôs seguidores de linha: (a) Robô *Alpha Project* (PETRY, 2016); (b) Pololu 3pi modificado (PETRY, 2016); (c) Protótipo de Guadagnin (2014); (d) Cartisx04 de Hirai (2014).

3 MATERIAIS

Neste capítulo serão apresentados os materiais a serem utilizados neste projeto.

3.1 MICROCONTROLADOR

Será utilizado o *Kit* de desenvolvimento NUCLEO-F303K8, da ST Microelectronics, o qual pode ser visto na Figura 14. Esta placa utiliza o microcontrolador STM32F303K8, que possui como processador ARM Cortex-M4. Este processador dispõe de uma FPU, a qual facilita o processamento dos cálculos em ponto flutuante que são necessários para o controle do veículo. As principais informações deste dispositivo estão na Tabela 2.

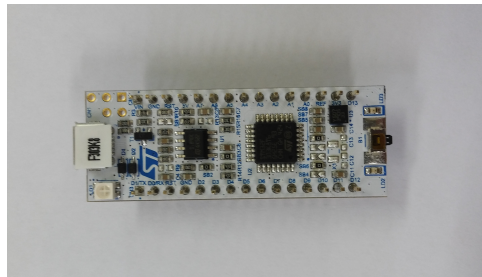


Figura 14 – Kit de desenvolvimento STM32-F303K8

Fonte: Autoria própria

Tabela 2 – Especificações do microcontrolador STM32F303K8

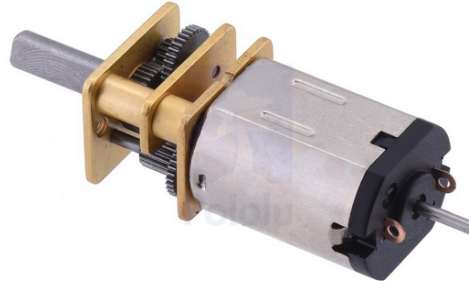
Característica	Descrição
Frequência de operação	72MHz
Desempenho	90 DMPIS ¹
Flash	64KB
SRAM	16KB
Quantidade de temporizadores (<i>timers</i>)	11
Quantidade de canais do ADC	21
Resolução do ADC	12 <i>bits</i>

(STMICROELECTRONICS, 2016)

3.2 MOTORES CC

Será utilizado o motor CC modelo 3041 da Pololu, o qual pode ser visto na Figura 15. Estes motores são classificados pela fabricante como *High-Power Carbon Brush* (HPCB), os

quais são motores escovados *brushed*². Este modelo possui o eixo estendido, possibilitando o acoplamento de um *encoder*, a partir do qual é possível fazer o mapeamento da pista. As principais especificações técnicas deste dispositivo estão na Tabela 3.



**Figura 15 – Motor HPCB 3041
(POLOLU, 2016b)**

Tabela 3 – Especificações do motor 3041 da Pololu

Característica	Descrição
Tensão de alimentação	12V
Corrente de alimentação (sem carga)	100mA
Corrente máxima de alimentação	800mA
Caixa de velocidade	10:1
Rotação máxima	3000 RPM
Torque máximo	0.3kg-cm

(POLOLU, 2016b)

3.3 PONTE H

Será utilizada uma Ponte H para o controle de velocidade dos motores. O modelo que será utilizado é o TB6612FNG, da Toshiba, que é capaz de controlar até dois motores CC com corrente constante de 1,2A (TOSHIBA, 2016). A velocidade do motor é controlada por *Pulse Width Modulation* (PWM).

3.4 ENCODER MAGNÉTICO

Um *encoder* magnético é um transdutor de movimento, que converte movimentos em informações elétricas, sendo possível obter dados como posição e velocidade. Neste trabalho

²Um motor escovado realiza a troca de fase do rotor através de escovas.

será utilizado o modelo 3081 da Pololu, o qual realiza 12 contagens por revolução do eixo e é compatível com o motor 3041 (POLOLU, 2016a).

3.5 SENSORES DE REFLETÂNCIA

O sensor de refletância é um dispositivo eletrônico que consiste de um *Light Emitter Diode* LED e um fototransistor, medindo assim a refletância de uma superfície. Este circuito será utilizado para detectar a linha do percurso. O modelo que será utilizado nesse trabalho é o QRE1113P, da Fairchild Semiconductor (SEMICONDUCTOR, 2016).

3.6 PLACA DE CIRCUITO IMPRESSO

O *chassi* do robô, ou seja, a estrutura deste, será confeccionada em uma placa de circuito impresso (PCB) de fenolite.

3.7 MÓDULO BLUETOOTH

Será utilizado o módulo *bluetooth* HC-05 para a telemetria. Este módulo possuiu a configuração mestre-escravo e comunicação *Universal Asynchronous Receiver Transmitter* (UART).

3.8 RODAS

Serão utilizadas rodas de poliuretano ou silicone.

3.9 BATERIA LIPO

Será utilizada uma bateria do tipo Lítio-Polímero (Li-Po) de duas células, 7,4V, 1300mAh e 32,5A de corrente máxima de descarga, pois possui alta capacidade de corrente e densidade de carga.

3.10 CONVERSOR STEP-UP

O conversor *Step-up* que será utilizado é o XL6009, que é um módulo elevador de tensão. Este circuito possui eficiência de 94%, corrente e tensão de saída máxima de 3A e 35V, respectivamente (XLSEMI, 2016).

3.11 ESFERA DESLIZANTE

Será utilizada uma esfera deslizante para sustentar a parte frontal do robô e manter os sensores de refletância em sua correta posição de funcionamento.

4 METODOLOGIA

Neste capítulo são descritas as etapas do trabalho proposto e como as quais serão executadas. A metodologia utilizada será baseada no método de prototipação, de Engenharia de *Software*, descrito em (PRESSMAN, 2010), em que o sistema é melhorado continuamente, como pode ser visto na Figura 17. A primeira etapa é a de coleta e análise de requisitos, em que serão definidos os componentes e materiais necessários para o desenvolvimento do projeto; em seguida, no “projeto rápido”, é feito o projeto de forma a fazer com que o protótipo seja funcional; em seguida serão feitos os testes e os refinamentos necessários para obter um produto final que melhor atenda aos requisitos esperados. Com base nesta abordagem, são definidas as seguintes etapas:

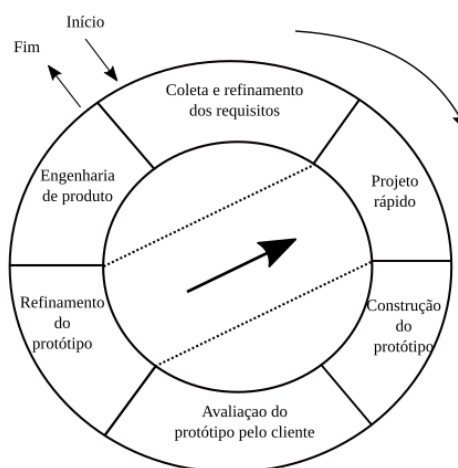


Figura 17 – Ciclo da prototipação
Fonte: Adaptado de (PRESSMAN, 2010, p.36).

1. Definição dos componentes: Segundo Pressman (2010), a primeira fase da engenharia de *hardware* compreende a análise de requisitos de *hardware* e o planejamento de desenvolvimento, o qual estabelece a finalidade dos dispositivos. Com base nisso, serão escolhidos os componentes que façam com que os objetivos de projeto possam ser executados;
2. Condicionamento de sinais: São utilizados sensores para que grandezas físicas pertinentes possam ser aferidas pelo sistema, tais como velocidade e tensão elétrica. No entanto, para que o dispositivo de aquisição de sinais possa medir o valor dos sensores eficaz e

corretamente, se faz necessário fazer o condicionamento de sinais destes dispositivos. Este processo será feito através das descrições contidas nos *datasheets* (folha de especificação) dos componentes e por simulações em *softwares* de circuitos elétricos.

3. Projeto do protótipo: Esta parte consiste em projetar o *chassi* do robô (a sua estrutura física) com todos os componentes necessários para o seu funcionamento. Será utilizado um *software* EDA para a criação da placa de circuito impresso (PCB). Nesta etapa também serão feitos os códigos no microcontrolador, que é o dispositivo de processamento do sistema.
4. Projeto do controlador: Para o projeto do controlador PID, será utilizada a modelagem experimental para obter um modelo matemático, visto que é mais simples do que a fenomenológica, conforme diz Guadagnin (2014). Para este controlador serão utilizados *softwares* matemáticos para a simulação do modelo obtido. Para o projeto do controlador de Sistemas a Eventos Discretos, será utilizado o modelo de Moore, pois é de fácil implementação e é oportuno ao que se deseja implementar. Será utilizado uma ferramenta computacional de controle supervisão para esta modelagem.
5. Sistema de telemetria: O sistema de telemetria tem o intuito de informar um operador externo sobre as condições de operação do robô. Neste trabalho, serão transmitidas informações como a velocidade do robô e o traçado da pista. A comunicação será feita por um dispositivo *bluetooth*.
6. Integração do sistema e implementação do protótipo: Com as etapas anteriores finalizadas, é necessário integrá-las. A integração será feita com os sensores e atuadores sendo soldados ou acoplados na PCB e os controladores e a aquisição de sinais implementados no microcontrolador.
7. Testes de desempenho: Serão realizados testes de desempenho de modo a verificar o que pode ou o que precisa ser melhorado no protótipo. Podem ser feitos ajustes tanto de *hardware* (como a substituição de componentes) quanto de *software* (alterações no código do microcontrolador). Pode ser necessário retornar a etapas anteriores para realizar estas correções.
8. Implementação do projeto final: Com os testes de desempenho e as alterações necessárias feitas, é implementado o projeto final de forma a validar este trabalho. Na versão final do veículo espera-se atingir os objetivos que foram propostos na Seção 1.4.

5 CRONOGRAMA PRELIMINAR

O Quadro 4 apresenta um cronograma preliminar do desenvolvimento do Trabalho de Conclusão de Curso (TCC). Na sua elaboração foi considerada a continuação do desenvolvimento no semestre seguinte, junto à disciplina de TCC 2.

Atividades	Meses											
	08/16	09/16	10/16	11/16	12/16	01/17	02/17	03/17	04/17	05/17	06/17	07/17
Elaboração e entrega da proposta												
Revisão bibliográfica												
Condicionamento de sinais dos sensores												
Projeto do protótipo												
Projeto do controlador												
Desenvolvimento do sistema de telemetria												
Elaboração e apresentação do TCC 1												
Confecção do protótipo												
Implementação do controlador												
Integração do sistema												
Testes finais de desempenho												
Elaboração e apresentação do TCC 2												

Quadro 4 – Cronograma das atividades previstas

REFERÊNCIAS

- ANSI/RIA.R15.06-1999. **Robot Terms and Definitions**. 2010. Disponível em: <<http://www.robotics.org/product-catalog-detail.cfm/productid/2953>>. Acesso em: 15 ago. 2016.
- ARAÚJO, F. M. U. de. **Sistemas de Controle**. Natal: DCA-UFRN, 2007. Apostila.
- CASSANDRAS, C. G.; LAFORTUNE, S. **Introduction to Discrete Event Systems**. 2. ed. New York: Springer US, 2008.
- DUDEK, G.; JENKIN, M. **Computational principles of mobile robotics**. 2. ed. [S.l.]: Cambridge University Press, 2010.
- DYNAMICS, B. **LS3 - Legged Squad Support System**. 2016. Disponível em: <http://www.bostondynamics.com/robot_ls3.html>. Acesso em: 25 set. 2016.
- FREITAS, R. S. de. **Arquitetura híbrida e controle de missão de robôs autônomos**. Dissertação (Mestrado) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2016.
- GUADAGNIN, A. J. **Controle Híbrido de um Robô Seguidor de Linha**. 2014. Trabalho de Conclusão de Curso. Universidade Tecnológica Federal do Paraná.
- HEINEN, F. J. **Sistema de Controle híbrido para Robôs Móveis Autônomos**. Dissertação (Mestrado) — Universidade do Vale do Rio dos Sinos, São Leopoldo, 2002.
- HIRAI, A. **Cartisx04**. 2014. Disponível em: <<http://anikinonikki.cocolog-nifty.com/blog/2014/11/cartisx04.html>>. Acesso em: 13 out. 2016.
- HIRAI, A. **All Japan Micro Mouse 2015 Robotrace Three Consecutive Victories**. 2016. Disponível em: <<http://anikinonikki.cocolog-nifty.com/blog/2015/11/2015-8a43.html>>. Acesso em: 08 nov. 2016.
- MAUÁ, I. **Instituto Mauá de Tecnologia sedia o evento de Robótica Winter Challenge 2016**. 2016. Disponível em: <<http://maua.br/imprensa/press-releases/instituto-maua-tecnologia-sedia-evento-robotica-winter-challenge-2016>>. Acesso em: 23 ago. 2016.
- MERRIAM-WEBSTER. **Online dictionary**. 2016. Disponível em: <<http://www.merriam-webster.com/dictionary/hybrid>>. Acesso em: 08 nov. 2016.
- NASA. 1997. Disponível em: <<https://mars.jpl.nasa.gov/MPF/rover/sim2.jpg>>. Acesso em: 24 set. 2016.
- NASA. **Mars Pathfinder/ Sojourner Rover**. 1997. Disponível em: <<http://www.jpl.nasa.gov/missions/mars-pathfinder-sojourner-rover>>. Acesso em: 24 set. 2016.
- NISE, N. S. **Engenharia de Sistemas de Controle**. 6. ed. Rio de Janeiro: LTC, 2012.

- OGATA, K. **Engenharia de Controle Moderno**. 5. ed. São Paulo: Pearson Prentice Hall, 2010.
- PESSIN, G. **Estratégias inteligentes aplicadas em robôs móveis autônomos e em coordenação de grupos de robôs**. Dissertação (Mestrado) — Universidade de São Paulo, São Carlos, 2013.
- PETRY, M. L. **Controle Híbrido de um robô autônomo seguidor de linha**. 2016. Trabalho de Conclusão de Curso. Universidade Tecnológica Federal do Paraná.
- POLOLU. **Pololu magnetic encoder datasheet**. 2016. Disponível em: <<https://www.pololu.com/product/3081>>. Acesso em: 01 set. 2016.
- POLOLU. **Pololu micromotor datasheet**. 2016. Disponível em: <<https://www.pololu.com/product/3048>>. Acesso em: 01 set. 2016.
- POLONSKII, M. **Introdução à robótica e mecatrônica**. 2. ed. Caxias do Sul: EDUCS, 1996.
- PRESSMAN, R. S. **Engenharia de Software**. São Paulo: Pearson, 2010.
- RECORDS, G. W. **Robotics tournament VEX Worlds is named largest in the world after 1,075 teams take part**. 2016. Disponível em: <<http://www.guinnessworldrecords.com/news/2016/4/robotics-tournament-vex-worlds-is-named-largest-in-the-world-after-1-075-teams-ta-426576>>. Acesso em: 29 ago. 2016.
- ROBOCORE. 2016. Disponível em: <<https://www.robocore.net/eventos>>. Acesso em: 22 ago. 2016.
- ROBOCORE. **Classificação WinterChallenge**. 2016. Disponível em: <https://www.robocore.net/modules.php?name=GR_Eventos&evento=24&tab=2>. Acesso em: 23 ago. 2016.
- ROBOCORE. **Regras Seguidor de Linha**. 2016. Disponível em: <https://www.robocore.net/upload/attachments/robocore_regras_seguidor_de_linha_108.pdf>. Acesso em: 28 ago. 2016.
- ROBOCUP. 2016. Disponível em: <<http://www.robocup.org/about-robocup/objective>>. Acesso em: 22 ago. 2016.
- ROBOGAMES. 2016. Disponível em: <<http://robogames.net/index.php>>. Acesso em: 29 ago. 2016.
- SECCHI, H. A. **Uma introdução aos robôs móveis**. Serra: NERA-IFES, 2012. Traduzido do original *Una Introducción a los Robots Móviles*.
- SEMICONDUCTOR, F. **Minature Reflective Object Sensor**. 2016. Disponível em: <<http://cdn.sparkfun.com/datasheets/Sensors/Proximity/QRE1113.pdf>>. Acesso em: 01 set. 2016.
- SIEGWART, R.; NOURBAKHSI, I. R.; SCARAMUZZA, D. **Introduction to autonomous mobile robots**. Cambridge: The MIT Press, 2011.
- STMICROELECTRONICS. **STM32F303x6/x8 datasheet**. 2016. Disponível em: <<http://www.mouser.com/ds/2/389/DM00092070-524505.pdf>>. Acesso em: 01 set. 2016.

TEIXEIRA, M. **Explorando o uso de Distinguidores e de Autômatos Finitos Estendidos na teoria do Controle Supervisório de Sistemas a Eventos Discretos**. Tese (Doutorado) — Universidade Federal de Santa Catarina, Florianópolis, 2013.

TORRICO, C. R. C. **Controle Supervisório Hierárquico de Sistemas a Eventos Discretos: Uma Abordagem Baseada na Agregação de Estados**. Tese (Doutorado) — Universidade Federal de Santa Catarina, Florianópolis, 2003.

TOSHIBA. **Driver IC for Dual DC motor**. 2016. Disponível em: <<https://www.pololu.com/file/0J86/TB6612FNG.pdf>>. Acesso em: 01 set. 2016.

WORLDS, V. 2016. Disponível em: <<http://www.roboticseducation.org/competition-teams/vex-robotics-competition/>>. Acesso em: 29 ago. 2016.

XLSEMI. **400KHz 60V 4A Switching Current Boost / Buck-Boost / Inverting DC/DC Converter**. 2016. Disponível em: <<https://www.pollin.de/shop/downloads/D351434D.PDF>>. Acesso em: 01 set. 2016.