



MILTON FELIPE SOUZA SANTOS

**ANÁLISE E PROPOSTA DE ARQUITETURAS DE
HARDWARE PARA VEÍCULOS AUTÔNOMOS**

CAMPINAS

2013



**UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO**

MILTON FELIPE SOUZA SANTOS

ANÁLISE E PROPOSTA DE ARQUITETURAS DE HARDWARE PARA VEÍCULOS AUTÔNOMOS

Orientador: Prof. Dr. José Raimundo de Oliveira

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas para obtenção do título de Mestre em Engenharia Elétrica, na área de Automação.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO
DEFENDIDA PELO ALUNO MILTON FELIPE SOUZA SANTOS
E ORIENTADA PELO PROF. DR. JOSÉ RAIMUNDO DE OLIVEIRA

CAMPINAS

2013

iii

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

Santos, Milton Felipe Souza, 1982-
Sa59a Análise e proposta de arquiteturas de hardware para veículos autônomos /
Milton Felipe Souza Santos. – Campinas, SP : [s.n.], 2013.

Orientador: José Raimundo de Oliveira.
Dissertação (mestrado) – Universidade Estadual de Campinas, Faculdade de
Engenharia Elétrica e de Computação.

1. Robótica. 2. Navegação autônoma. 3. Sistemas embutidos de computador.
4. Hardware - Arquitetura. 5. Detectores. I. Oliveira, José Raimundo de, 1950-. II.
Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de
Computação. III. Título.

Informações para Biblioteca Digital

Título em inglês: Analysis and proposal of hardware architectures for autonomous vehicles

Palavras-chave em inglês:

Robotics

Autonomous navigation

Embedded computer systems

Hardware - Architecture

Sensors

Área de concentração: Automação

Titulação: Mestre em Engenharia Elétrica

Banca examinadora:

José Raimundo de Oliveira [Orientador]

Fabricao Nicolato

Marconi Kolm Madrid

Data de defesa: 17-05-2013

Programa de Pós-Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE MESTRADO

Candidato: Milton Felipe Souza Santos

Data da Defesa: 17 de maio de 2013

Título da Tese: "Análise e Proposta de Arquiteturas de Hardware para Veículos Autônomos"

Prof. Dr. José Raimundo de Oliveira (Presidente):



Prof. Dr. Fabricio Nicolato:



Prof. Dr. Marconi Kolm Madrid:



*À minha esposa Ana Raquel, por todo amor, carinho e
dedicação.*

Agradecimentos

À Deus por ter permitido que este trabalho fosse feito.

A minha querida esposa Ana Raquel pelo incentivo e paciência nas noites inteiras dedicadas a pesquisa e escrita deste trabalho.

Aos meus pais Milton e Nadirlene pelo incentivo aos estudos que originou este trabalho.

Ao meu orientador Prof. Dr. José Raimundo de Oliveira pela paciência e grande experiência na pesquisa e redação do conteúdo aqui finalizado.

Aos amigos Jorge Polar, Leonardo Silvério, André Távora, Eduardo Silva, Miguel Vig Filho, Moisés Danziger e muitos outros pelo incentivo a conclusão deste infindável trabalho.

*“This is an exercise in fictional science, or science fiction, if you like that better. Not for
amusement: science fiction in the service of science.”*
(Valentino Braitenberg)

Resumo

Este trabalho analisa as possibilidades de arquiteturas de hardware buscando encontrar uma plataforma de desenvolvimento escalável e possível de se adaptar a diferentes estruturas mecânicas de veículos móveis. Esta plataforma deve ainda ter sensores suficientes para permitir comportamentos autônomos ao veículo. Para isto, de forma a entender melhor o conceito de autonomia, primeiramente foram analisadas as capacidades mentais humanas. Das capacidades estudadas foram escolhidas as capacidades de sensação, percepção, orientação e cognição como capacidades relacionadas ao hardware quando implantadas em veículos móveis artificialmente. Em seguida cada uma destas quatro capacidades mentais foi analisada a fundo buscando métodos e sistemas para solucionar estes problemas. Foram analisadas também as possibilidades de topologias em rede de forma a conectar os módulos individuais e propostos critérios de escolha dos módulos e topologias do sistema. Após todas estas análises exaustivas, onde somente as partes mais importantes foram incluídas neste trabalho, foi escolhida a topologia de barramento serial com arbitração em hardware para permitir modularidade, escalabilidade e confiabilidade. Como barramento elétrico foi escolhido o barramento CAN (Controller Area Network) que juntamente com diversos módulos especificados neste trabalho resultam na plataforma que foi chamada “Sistema Autônomo Completo”. Com base neste sistema, foram propostos como resultados duas simplificações: uma baseada em veículos que operam no solo (bidimensional) e outra para veículos que operam em ambientes tridimensionais como o ar ou a água. Buscando um enfoque educacional foi proposta uma sequência de implantação do sistema autônomo completo e alguns estudos de casos estão apresentados. Com este trabalho foi possível estruturar e criar uma sequência de desenvolvimento de um veículo móvel robótico em fases que podem ser facilmente seguidas por escolas ou empresas.

Palavras-chave: robótica, robôs móveis, navegação autônoma, sistemas embarcados, sistemas embutidos de computador, arquitetura de hardware, hardware – arquitetura, sensores, detectores, navegação, orientação, guiagem, redundância de sensores, odometria, robótica de exteriores, robótica cognitiva, robôs móveis educacionais, robôs móveis de baixo-custo, robôs autônomos.

Abstract

This work analyzes hardware architectures of embedded systems, searching for a development platform of mobile robots. This platform must be scalable and easily adaptable to several types of mechanical designs of mobile vehicles. And it must have enough sensors in order to reach autonomous performance.

For better understanding concepts of autonomy, the human brain capacities were studied. From the studied capacities, sensation, perception, representation and orientation were chosen. These four capacities were chosen as related with hardware implementations if artificially designed for mobile vehicles. Network topologies for connecting modules of independent methods for the artificial brain capabilities mentioned were also analyzed.

For choosing the best proposal, some criteria were defined for the modules and system topologies. After this analysis, where only the most important parts were included, a topology was chosen. The chosen topology is the serial bus with hardware arbitration. The chosen electrical bus was the CANbus, which together with the other modules specified in this work resulted in the platform called “Full Autonomous Vehicle”.

Based on this system platform, simplifications were proposed: one focused on vehicles with two-dimensional movements, and other focused on vehicles with three-dimensional movements. Searching for an educational point-of-view, an implementation sequence was proposed for the full autonomous vehicle and some cases were studied. With this work was possible to organize and create a development sequence of a robotic mobile vehicle divided by phases. These phases can be easily followed by schools and companies.

Keywords: *robotics, mobile robots, autonomous navigation, embedded systems, computer embedded systems, hardware architecture, sensors, navigation, orientation, guidance, sensors redundancy, odometry, exterior robotics, cognitive robotics, educational mobile robots, low-cost mobile robots, autonomous robots, hardware architecture.*

Lista de figuras

Figura 1 – Diagrama em blocos de referência de um sistema de navegação autônoma proveniente de uma compilação de diagramas de livros e websites.....	3
Figura 2 – Diagrama representativo dos níveis de abstração do estudo de autonomia	4
Figura 3 – Escolha das perguntas para desenvolvimento deste trabalho.....	5
Figura 4 – Controlador inteligente hierárquico aninhado.....	6
Figura 5 – Modelo lógico do controlador inteligente hierárquico.....	7
Figura 6 – Tipos de erros comuns. (a) Bias; (b) Fator de Escala; (c) Não-linearidade; (d) Assimetria; (e) Zona Morta; (f) Quantização.	24
Figura 7 – Sistema robótico considerando erros possíveis.....	27
Figura 8 – Fusão direta de dados.....	30
Figura 9 – Representação dos dados por vetores com subsequente fusão dos vetores.	30
Figura 10 – Processamento individual de cada sensor com subsequente combinação de resultados.....	31
Figura 11 - Sensores utilizados em guiagem táctil. (a) chave de contato; (b) chave de fim de curso; (c) detector de distância por ultrassom.	34
Figura 12 – <i>Encoder</i> óptico incremental com sinais em quadratura; (a) desenho esquemático de um disco óptico incremental; (b) foto de um disco óptico incremental utilizado no robô Asuro.	35
Figura 13 – Pulsos obtidos nas saídas dos fotorreceptores onde cada pulso determina a distância percorrida e a fase de um sinal em relação ao outro indica o sentido de giro da roda.	35
Figura 14 – Robô Asuro.	37
Figura 15 – Método de localização por trilateração.	41
Figura 16 – Representação dos pulsos para o cálculo das distâncias no método de triangulação.	42
Figura 17 – Representação de um sistema de trilateração GPS.	44

Figura 18 – Localização por triangulação. Um sensor rotativo instalado no veículo mede os ângulos λ_1 , λ_2 e λ_3 entre o eixo longitudinal do veículo e as três fontes de sinais de referência S1, S2 e S3.....	45
Figura 19 – Esquema representando as formas de onda recebidas pelo método de triangulação.	45
Figura 20 – Esquema mostrando os ângulos utilizados nos cálculos do método de triangulação.	46
Figura 21 – Diagrama do processo de posicionamento por marcas.	48
Figura 22 – Localização com o uso de visão global aplicado ao algoritmo de localização de Monte Carlo.....	49
Figura 23 – Diagrama em blocos das funções processadas em um veículo autônomo usando visão global.....	50
Figura 24 – Localização através de marcas naturais utilizando vídeo.	53
Figura 25 – Robô Experimental SAVAR – Robô que utiliza somente uma câmera para se localizar.	54
Figura 26 – Modelo de perspectiva de uma câmera.	54
Figura 27 – Microsoft Kinect. (a) produto. b) componentes internos de vídeo. (c) diagrama esquemático dos componentes internos do Kinect.	56
Figura 28 – Sequencia de rastreamento esquelético do Kinect.	56
Figura 29 – Plataforma de desenvolvimento Eddie do consórcio Microsoft-Parallax. Conjunto composto de uma base robótica com motores e sensores, um <i>laptop</i> e um kinect.	57
Figura 30 – Projeto utilizando navegação por mapas.....	58
Figura 31 – Diagrama em blocos do processamento de um sistema de navegação por mapas.....	59
Figura 32 – Algoritmo básico para construção de mapas.....	60
Figura 33 – Topologias de sistemas modulares.....	66
Figura 34 – Arquitetura básica da aviônica do foguete Sonda-L.	81
Figura 35 – Barramento de Alimentação Misto.	82

Figura 36 – Sistema autônomo completo com identificação das funcionalidades e atividades mentais humanas equivalentes.....	85
Figura 37 – Diagrama de exemplo do circuito eletrônico do módulo de sensores de proximidade (SP).....	87
Figura 38 – Fluxograma exemplo do software recomendado para o módulo SP.....	88
Figura 39 – Diagrama de exemplo do circuito eletrônico do módulo de cálculo odométrico (CO).....	89
Figura 40 – Fluxograma exemplo do software recomendado para o módulo CO.....	90
Figura 41 – Diagrama de exemplo do circuito eletrônico do módulo de unidade de medição inercial (UMI).....	91
Figura 42 – Fluxograma exemplo do software recomendado para o módulo UMI.	92
Figura 43 – Diagrama de exemplo do circuito eletrônico do módulo de triangulação/trilateração (TT).....	93
Figura 44 – Fluxograma exemplo do software recomendado para o módulo TT.	94
Figura 45 – Diagrama de exemplo do circuito eletrônico do módulo de dados GPS (MDGPS).....	95
Figura 46 – Fluxograma exemplo do software recomendado para o módulo MDGPS.	96
Figura 47 – Diagrama de exemplo do circuito eletrônico do módulo de registro de mapas (RM).	97
Figura 48 – Fluxograma exemplo do software recomendado para o módulo RM.	98
Figura 49 – Diagrama de exemplo do circuito eletrônico do módulo de localização por marcas (LM).	99
Figura 50 – Fluxograma exemplo do software recomendado para o módulo LM.	100
Figura 51 – Diagrama de exemplo do circuito eletrônico do módulo de visão computacional (VC).....	101
Figura 52 – Fluxograma exemplo do software recomendado para o módulo VC.....	102
Figura 53 – Diagrama de exemplo do circuito eletrônico do módulo de atuação e controle (MAC).	103
Figura 54 – Fluxograma exemplo do software recomendado para o módulo MAC.	104

Figura 55 – Diagrama de exemplo do circuito eletrônico do módulo de alimentação (MA).	105
Figura 56 – Fluxograma exemplo do software recomendado para o módulo MA.	106
Figura 57 – Diagrama de exemplo do circuito eletrônico do computador de bordo (CB).	107
Figura 58 – Exemplo de diagrama de estados do computador de bordo (CB).	108
Figura 59 – Fluxogramas exemplo para os estados de INICIALIZAÇÃO (a) e DEPURAÇÃO (b) do módulo CB.	108
Figura 60 – Fluxogramas exemplo para os estados de PRÉ-OPERAÇÃO (a), OPERAÇÃO (b) e EMERGÊNCIA (c) do módulo CB.	109
Figura 61 – Diagrama de exemplo do circuito eletrônico do módulo de comunicação (MC).	110
Figura 62 – Fluxograma exemplo do software recomendado para o módulo MC.	111
Figura 63 – Diagrama de exemplo do circuito eletrônico do módulo de emergência (ME).	112
Figura 64 – Fluxograma exemplo do software recomendado para o módulo ME.	113
Figura 65 – Diagrama de exemplo do circuito eletrônico do registrador de bordo (RB)...	114
Figura 66 – Fluxograma exemplo do software recomendado para o módulo RB.	115
Figura 67 – Diagrama da montagem do cabo de depuração para o projeto do foguete Sonda- L.	116
Figura 68 - Fases da implementação do desenvolvimento de um projeto de um veículo autônomo.	122
Figura 69 – Arquitetura geral mínima 2D.	124
Figura 70 – Configuração Ackerman. Em uma configuração de rodas Ackerman todos os eixos estendidos para todas as rodas se encontram em um mesmo ponto.	124
Figura 71 – Diagrama da arquitetura proposta para veículo com configuração Ackerman de rodas para movimentação indoor.	125
Figura 72 – Configuração mecânica com esteira.	126
Figura 73 – Diagrama da arquitetura proposta para veículo com configuração mecânica em esteira tipo “tanque de guerra” para terrenos off-road.	126

Figura 74 – Arquitetura geral mínima 3D.	128
Figura 75 – Configuração mecânica de um dirigível de hélio para ambientes indoor.	128
Figura 76 – Diagrama da arquitetura proposta para um dirigível de hélio autônomo se movendo no espaço livre.	129
Figura 77 – Propostas de configurações mecânicas de aeronaves não tripuladas existentes no mercado.	130
Figura 78 – Diagrama da arquitetura proposta para uma aeronave não tripulada autônoma.	130
Figura 79 – Crescimento das “elipses de erros” indicando o crescimento da incerteza de posição com a odometria.	146
Figura 80 – Rodas castor amplamente utilizada em robótica.	147
Figura 81 – Arquitetura Synchro-drive, mostrando vista superior, inferior e o mecanismo das rodas que permite o controle de direção de tração por meio de correntes.	148
Figura 82 – Arquitetura diferencial (vista inferior).	148
Figura 83 – Desenho conceitual de rodas de encoders para um veículo com tração diferencial.	149
Figura 84 – Encoder trailer. O carrinho (trailer) foi projetado para aumentar a precisão em um robô que utiliza esteiras como tração.	150
Figura 85 – Unidade de medição inercial montada sob plataforma mecânica de suporte cardã (Gimbaled IMU).	155
Figura 86 – Unidade de medição inercial montada sob plataforma mecânica flutuante (Floated IMU) e uma unidade de medição inercial solidária (Strapdown ISA).	156
Figura 87 – Esquema de montagem de um sistema de navegação inercial solidário (Strapdown INS).	156
Figura 88 – Diagrama de processamento de sinais de navegação utilizando unidade de medição montada sob plataforma mecânica de suporte cardã (Gimbaled IMU).	159
Figura 89 – Diagrama de processamento de sinais de navegação utilizando unidade de medição solidária (Strapdown ISA) e sistema de coordenadas de Euler.	160
Figura 90 - Frames do barramento CAN.	162
Figura 91 – Esquema de arbitração do barramento CAN.	163

Lista de tabelas

Tabela 1 – Estágios de desenvolvimento das faculdades mentais em máquinas.....	14
Tabela 2 – Tabela síntese das características de sensores de medidas relativas e absolutas.	21
Tabela 3 – Classificação dos sensores utilizados em aplicações de robótica móvel.	22
Tabela 4 – Aplicações representativas de fusão sensorial não militares.	28
Tabela 5 – Análise comparativa dos métodos de navegação estudados.	62
Tabela 6 – Prós e contras das topologias de sistema apresentadas.	67
Tabela 7 – Guia de análise de barramentos elétricos.	68
Tabela 8 – Comparativo de redes WPAN.	72
Tabela 9 – Vantagens e desvantagens das redes WPAN.	73
Tabela 10 – Sugestão de trabalhos científicos para desenvolvimento de conceitos de autonomia.	134
Tabela 11 – Vantagens e desvantagens das plataformas inerciais mecânicas e analíticas.	157
Tabela 12 – Densidade de energia em baterias.	170
Tabela 13 – Comparativo entre baterias.	170
Tabela 14 – Sugestão de mensagens CAN para o módulo SP.	171
Tabela 15 – Sugestão de mensagens CAN para o módulo CO.	173
Tabela 16 – Sugestão de mensagens CAN para o módulo UMI.	174
Tabela 17 – Sugestão de mensagens CAN para o módulo TT.	176
Tabela 18 – Sugestão de mensagens CAN para o módulo MDGPS.	177
Tabela 19 – Sugestão de mensagens CAN para o módulo RM.	178
Tabela 20 – Sugestão de mensagens CAN para o módulo LM.	179
Tabela 21 – Sugestão de mensagens CAN para o módulo VC.	180
Tabela 22 – Sugestão de mensagens CAN para o módulo MAC.	181

Tabela 23 – Sugestão de mensagens CAN para o módulo MA.	182
Tabela 24 – Sugestão de mensagens CAN para o módulo CB.....	183
Tabela 25 – Sugestão de mensagens CAN para o módulo ME.....	186
Tabela 26 – Sugestão de mensagens CAN para o módulo RB.....	188

Lista de abreviaturas e siglas

ACK	Acknowledgement
CAN	Controller Area Network
CB	Computador de Bordo
CO	Módulo de Cálculo Odométrico
CPU	Central Processor Unit
CRC	Cyclic Redundancy Check
CSMA/CD	Carrier Sense Multiply Access/Collision Detect
DTMF	<i>Dual-Tone Multi-Frequency</i> signaling
EMI	Electromagnetic Interference
GPS	Global Positioning System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
ISA	Inertial Sensor Assembly
ISO	International Organization for Standardization
LM	Módulo de Localização por Mapas
MA	Módulo de Alimentação
MAC	Módulo de Atuação e Controle
MC	Módulo de Comunicação
MDGPS	Módulo de Dados GPS
ME	Módulo de Emergência
NDA	Non-Destructive Arbitration
RB	Registrador de Bordo
RM	Módulo de Registro de Mapas
RTOS	Real-Time Operating System
SP	Módulo de Sensores de Posição
TT	Módulo de Triangulação/Trilateração
UMI	Módulo de Unidade de Medição Inercial
VC	Módulo de Visão Computacional

Sumário

1	Introdução	1
1.1	Motivação	3
1.2	Objetivos	7
1.3	Trabalhos Correlatos	8
1.4	Organização da Dissertação	9
2	Conceitos de Inteligência.....	11
2.1	Conceitos	11
2.2	Atividades Mentais	12
2.3	Composição da Inteligência.....	13
2.4	Inteligência Artificial	14
3	Atividades Mentais Artificiais.....	19
3.1	Sensação/Percepção	19
3.1.1	Sensores	20
3.1.2	Erros em Sensores	23
3.1.3	Fusão Sensorial.....	28
3.1.4	Robótica Probabilística.....	32
3.2	Orientação	33
3.2.1	Táctil.....	33
3.2.2	Odométrica	34
3.2.3	Inercial	37
3.2.4	Sinais de Referência	39

3.2.5 Marcas	47
3.2.6 Vídeo	52
3.2.7 Mapas	57
3.2.8 Tabela comparativa	61
3.3 Ação Motora	64
3.4 Comentários Gerais.....	64
4 Interconexões.....	65
4.1 Topologia	65
4.2 Barramento Elétrico de Comunicação	68
4.2.1 Comunicações sem fio.....	71
4.3 Barramento de alimentação.....	73
4.4 Sistema Operacional	75
4.5 Conceitos Auxiliares.....	76
4.5.1 Cabo de Depuração.....	76
4.5.2 Pulso de Sincronismo	77
4.5.3 Módulo de Emergência.....	77
5 Critérios e Resultados.....	79
5.1 Sistema Autônomo Completo	79
5.1.1 Critérios / Requisitos	79
5.2 Topologia Escolhida	80
5.3 Barramento Elétrico de Comunicação Escolhido	81
5.4 Barramento de Alimentação Escolhido	82
5.5 Sistema Operacional	83
5.6 Resultados	83

5.6.1 Módulo de sensores de proximidade (SP)	87
5.6.2 Módulo de cálculo odométrico (CO).....	88
5.6.3 Módulo de Unidade de Medição Inercial (UMI).....	90
5.6.4 Módulo de triangulação/trilateração (TT)	92
5.6.5 Módulo de dados GPS (MDGPS).....	95
5.6.6 Módulo de registro de mapas (RM).....	97
5.6.7 Módulo de Localização por marcas (LM)	99
5.6.8 Módulo de visão computacional (VC)	101
5.6.9 Módulo de atuação e controle (MAC).....	103
5.6.10 Módulo de alimentação (MA)	105
5.6.11 Computador de bordo (CB)	107
5.6.12 Módulo de comunicação (MC).....	109
5.6.13 Módulo de emergência (ME).....	112
5.6.14 Registrador de bordo (RB)	114
5.6.15 Cabo de Depuração (CD)	116
5.7 Discussão / Comentários.....	117
5.7.1 Fase 1 – Robô Comandado.....	117
5.7.2 Fase 2 – Robô Autônomo Simples	118
5.7.3 Fase 3 – Robô Autônomo Simples	118
5.7.4 Fase 4 – Robô Autônomo com Redução de Erros de Estimativa.....	118
5.7.5 Fase 5 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias	119
5.7.6 Fase 6 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas	119

5.7.7 Fase 7 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas em Ambientes Externos	119
5.7.8 Fase 8 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas em Ambientes Externos com Acompanhamento Externo e Controle de Emergências.....	119
6 Estudo de Casos.....	123
6.1 Arquitetura geral mínima bidimensional	123
6.1.1 Estudo de casos.....	124
6.2 Arquitetura geral mínima tridimensional.....	127
6.2.1 Estudo de casos.....	128
7 Conclusão e Trabalhos Futuros	133
7.1 Conclusão.....	133
7.2 Trabalhos Futuros	133
Apêndice A – Erros na navegação odométrica.....	145
A.1 Erros.....	145
A.2 Redução de Erros Odométricos	146
Apêndice B – Sistema de Navegação Inercial.....	153
B.1 Dead Reckoning.....	153
B.2 Sistemas de Navegação Inercial.....	153
Apêndice C – Barramento CAN - CANBus.....	161
C.1 Funcionamento.....	162
C.2 Identificadores.....	164
C.3 Endereçamento e arbitração	164
C.4 Checagem de Erro.....	165
C.5 Realização	166

Apêndice D – Baterias Recarregáveis	169
Apêndice E – Mensagens CAN	171

1 Introdução

A palavra ROBÔ foi primeiramente utilizada pelo escritor checo Karel Čapek em sua peça teatral chamada R.U.R (“Rossum’s Universal Robots” – Robôs Universais do Rossum) escrita em 1920 e apresentada pela primeira vez no começo de 1921. Nesta peça teatral, Čapek apresenta os primeiros Robôs se referindo a humanoides para desempenharem todo o trabalho físico.

A ideia de se fazer uma máquina capaz de se locomover tomando decisões é uma ideia antiga e uma das vertentes de seu estudo se baseia em estudos que envolvem seres humanos e animais analisando as suas capacidades e atitudes, sejam elas instintivas ou cognitivas.

O casamento entre a teoria de controle, a ciência da informação e a biologia buscando explicar os princípios comuns entre controle e comunicação tanto entre animais como em máquinas é chamado de cibernética. Começando no fim da década de 1940 com Norbert Wiener e com W. Grey Walter na década de 1950 alguns dos princípios da cibernética foram criados (ARKIN, 1998).

Em 1984, o ciberneticista italiano Valentino Braitenberg desenvolveu uma teoria em que veículos simples (com pouco ou nenhum processamento) podem apresentar comportamentos semelhantes a características humanas. Esta forma de analisar características psicológicas de forma simples e instintiva culminou com os chamados “Veículos de Braitenberg” onde sensores são ligados diretamente aos motores, criando-se a aparência de reações “instintivas” dos robôs (BRAITENBERG, 1986). Esta filosofia fez surgir a “BEAM Robotics” (Biology, Electronics, Aesthetics, and Mechanics – do inglês, Biologia, Eletrônica, Estética e Mecânica) que é baseada em sistemas robóticos inspirados na natureza. A expressão “BEAM Robotics” foi criada por Mark Tilden em 1990 durante uma discussão no centro de ciências de Ontario, Canadá. Estes sistemas inspirados na natureza são implementados utilizando-se somente sensores, motores, comparadores e circuitos analógicos de forma a se obter flexibilidade e robustez (SEALE, 2003). Assim, a

busca da robótica se dá no entendimento de como as características desejadas podem ser obtidas em uma máquina, tentando-se obter o sistema mais simples possível.

Uma das áreas de maior importância em robótica de qualquer tipo é a guiagem (também chamada de navegação). Os sistemas/algoritmos de guiagem robótica já são estudados há muitos anos em diversas áreas do conhecimento.

Guiagem robótica pode ser definida (por livre adaptação) como o processo de determinação de caminho para um alvo e sua trajetória. Assim, quando se fala em uma máquina autônoma, está se falando de uma máquina que consegue se mover e concatenar objetivos com meios de guiagem; de forma a tomar decisões sem o auxílio de uma mente humana ou de qualquer outro meio externo ativo.

Leonard e Durrant-White (1991) traça um paralelo com o ser humano, quando se deseja chegar a algum lugar é necessário saber em que lugar se está, aonde se quer chegar e como se quer chegar neste lugar, ou seja, se faz necessário responder as perguntas “onde?”, “para onde?” e “como?”. Respondendo estas perguntas, o ser humano é capaz de obter a posição desejada e desenhar uma figura mental da trajetória.

Basicamente, o processo de guiagem robótica pode ser imaginado da mesma forma que nós, humanos, de fato fazemos. Porém, este processo se torna bem mais complicado quando desejamos autonomia e transferimos para uma máquina o processo de decisão sobre como ir e como se comportar com adversidades no caminho.

A Figura 1 mostra uma compilação dos diversos modelos de guiagem adotado por Siegwart, Nourbakhsh e Scaramuzza no livro “Introduction to Autonomous Mobile Robots” (SIEGWART, NOURBAKHSH e SCARAMUZZA, 2011). Este modelo mostra um paralelo entre conceitos de engenharia e conceitos de psicologia e utiliza diversas atividades semelhantes às chamadas “faculdades mentais” características dos seres humanos.

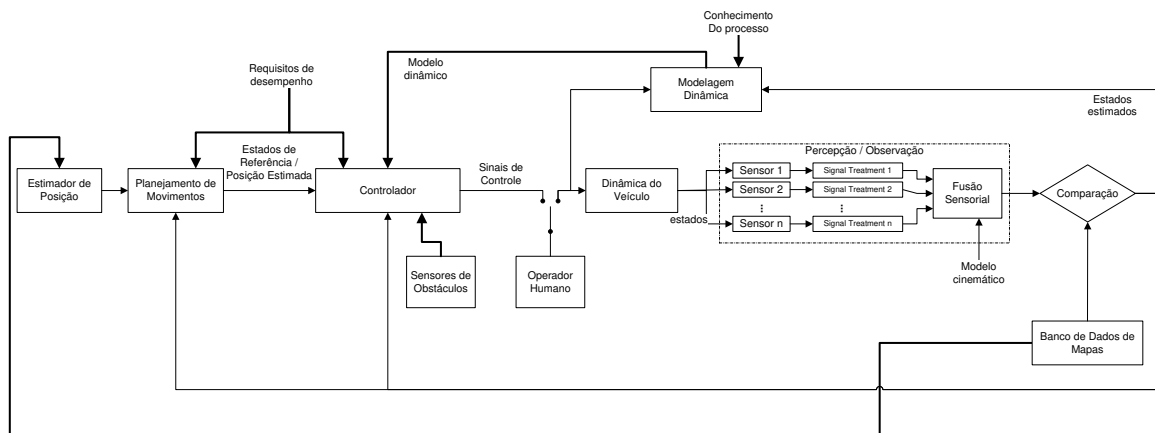


Figura 1 – Diagrama em blocos de referência de um sistema de navegação autônoma proveniente de uma compilação de diagramas de livros e websites.

Fonte: (SIEGWART, NOURBAKHSH e SCARAMUZZA, 2011).

As partes descritas na Figura 1 serão discutidas ao longo deste documento. Ainda, outras formas de se obter uma máquina inteligente serão citadas e estarão presentes na análise qualitativa final.

1.1 Motivação

A principal motivação para a realização deste trabalho foi obter uma plataforma de desenvolvimento para sistemas robóticos autônomos. Ao verificar dificuldades relacionadas ao estudo de robótica autônoma no âmbito da academia e ainda uma dificuldade em trabalhar numa plataforma única para diversos estudos de autonomia, o autor propôs a buscar uma solução para estes problemas através da pesquisa dos diversos tipos de guiagem robótica e de outras características que permitam que um dispositivo robótico móvel atinja a autonomia.

A busca pela autonomia de decisão é algo bastante estudado por diversos campos da ciência. Esta busca tem por objetivo facilitar o controle de dispositivos móveis. Reduzindo-se os comandos a simples instruções da vontade do usuário do sistema robótico.

Cabendo-se, assim, ao sistema robótico toda a tomada de decisão quanto aos métodos e forma de se atingir o objetivo, ou seja, autonomamente.

O estudo de itens relacionados à autonomia pode passar por diferentes níveis de abstrações. Tomando-se como referência a Figura 2, o estudo da autonomia em robôs móveis pode partir do nível mais concreto, ou o núcleo do diagrama que é o estudo de sua mecânica e suas relações com o meio. No nível logo acima está o estudo de sensores que mostra (no sentido contrário a parte mecânica) a interação do ambiente com o dispositivo móvel. Em seguida, segue o nível de abstração da conversão dos dados obtidos pelos sensores em dados válidos, ou seja, a percepção. A etapa de localização se configura como o quarto nível de abstração onde se somam as características obtidas da percepção sobre a posição atual do robô e sua atitude em relação a um referencial. Quando se combina a posição atual com a posição futura através de um caminho, atinge-se o quinto nível de abstração. A guiagem, o sexto nível de abstração, se caracteriza pela soma da trajetória com os dados de “consciência” do dispositivo móvel, ou seja, dados sobre o estado dele e do ambiente ao seu redor. Ao se combinar a tomada de decisão, que é indicada como a sétima camada de abstração, com a guiagem atinge-se a oitava camada de abstração, ou seja, a autonomia.

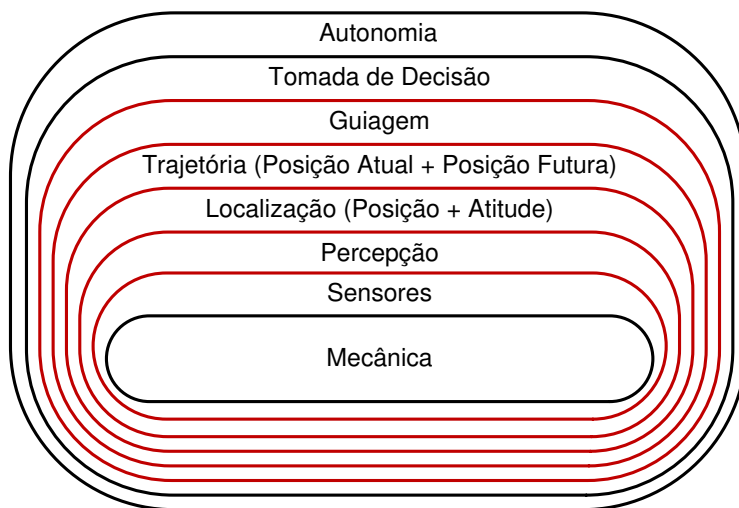


Figura 2 – Diagrama representativo dos níveis de abstração do estudo de autonomia

Contudo, observando-se os níveis de abstração da Figura 2, verifica-se que o nível mais baixo (mecânica) é amplamente desenvolvido por pesquisas e cálculos da mecatrônica. E os dois últimos níveis (tomada de decisão e autonomia) são caracterizados pelo estudo de algoritmos que podem ser implementados por software ou por outras tecnologias de hardware evolutivo. Porém, o estudo de algoritmos é extenso e não será tratado neste trabalho.

Sendo assim, este trabalho se atém no estudo de itens relacionados da segunda a sexta camadas de abstração. Desta forma, desejamos obter respostas para as perguntas “onde estou?” e “para onde vou?” no âmbito da robótica móvel. Deixando para outros trabalhos responder a pergunta “como irei?”. A Figura 3 mostra alguns dos temas tratados no parágrafo anterior em conjunto com a escolha dos itens referentes ao desenvolvimento de *hardware*.

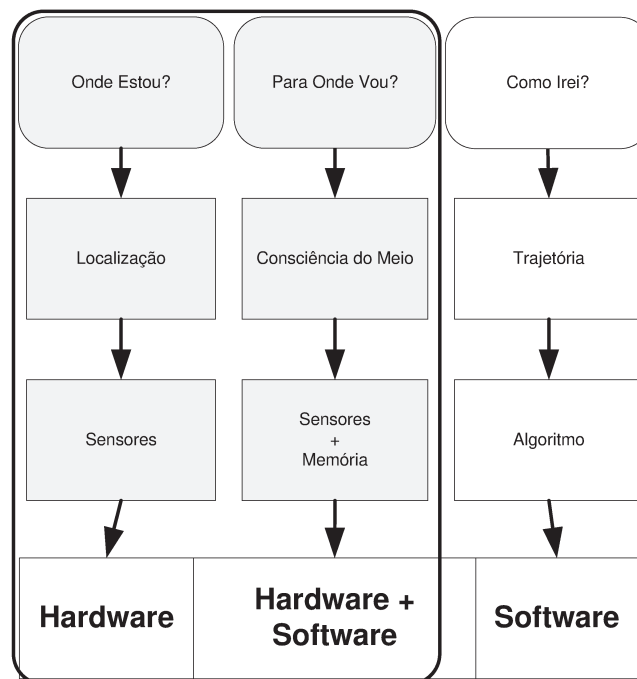


Figura 3 – Escolha das perguntas para desenvolvimento deste trabalho.

O formato apresentado na Figura 2 como representação dos níveis de arbitração do estudo de autonomia é semelhante ao formato hierárquico aninhado do controlador inteligente de Arkin (1998, p. 25) mostrada na Figura 4. Para este tipo de controlador inteligente, o modelo lógico da tomada de decisão é mostrado na Figura 5.

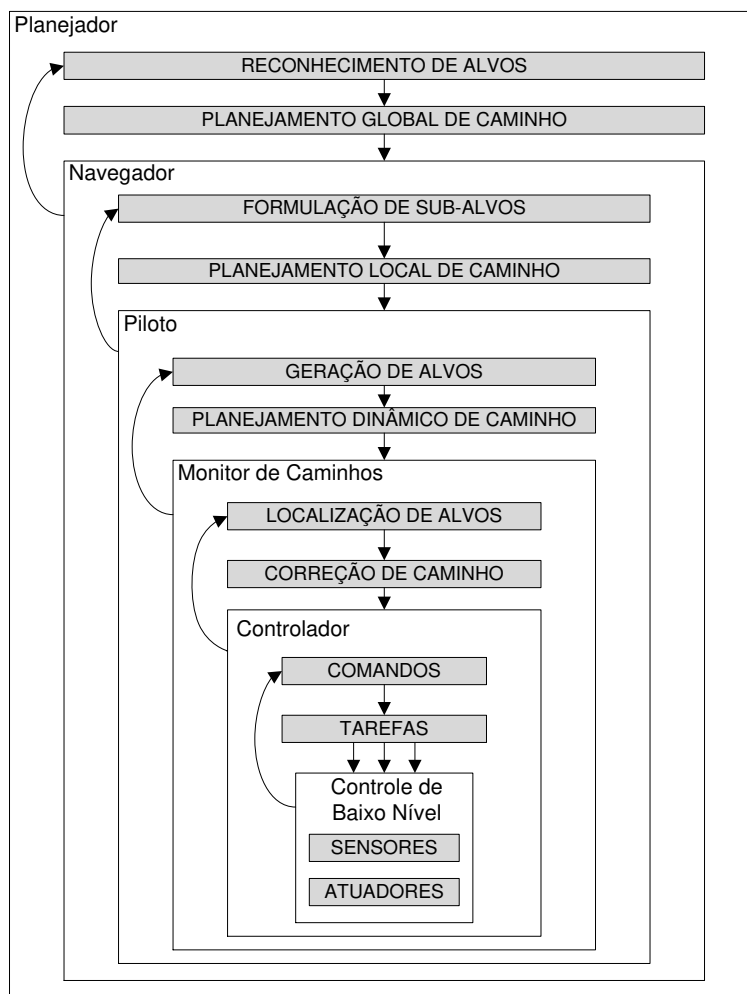


Figura 4 – Controlador inteligente hierárquico aninhado.

Fonte: (ARKIN, 1998)

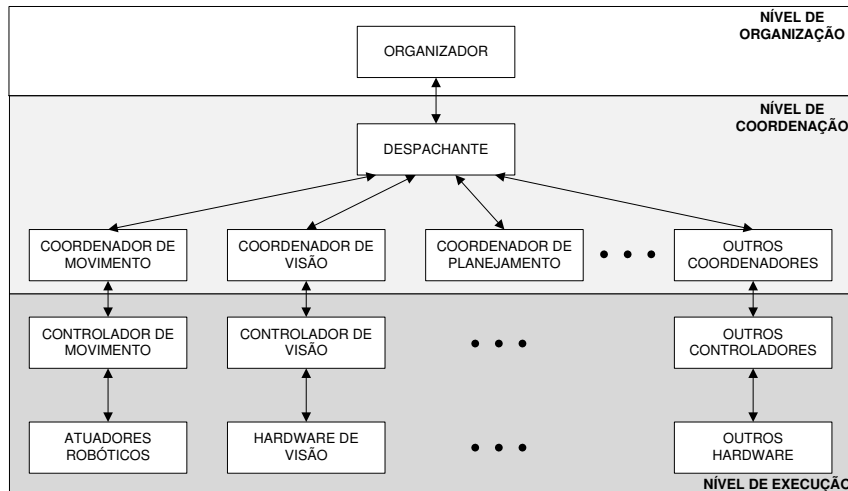


Figura 5 – Modelo lógico do controlador inteligente hierárquico.

Fonte: (ARKIN, 1998)

Existem diversos métodos, uns mais eficientes que outros, para desenvolver a guiagem de uma máquina móvel. Este trabalho foi baseado nos recursos de *hardware* para permitir que as aplicações de *software* atinjam comportamentos de autonomia.

1.2 Objetivos

Este trabalho tem como objetivo obter uma arquitetura de hardware escalável para veículos móveis de forma a se atingir comportamento autônomo. Adicionalmente, este trabalho se propõe a indicar uma plataforma de estudo de conceitos de autonomia.

Para isto, será buscado primeiramente um “sistema autônomo completo”¹ baseado nas faculdades mentais humanas como modelo. Este sistema autônomo deve ser capaz de suportar a maioria das possibilidades de hardware e software tratadas neste documento. Com o “sistema autônomo completo” também deve ser possível utilizar a

¹ A busca por um “sistema autônomo completo” se apresenta bastante pretenciosa. Esta busca marca o alvo. O objetivo com a qual o autor se baseou para a escolha do sistema. Evidentemente, por motivos de limitação de tempo e conteúdo de uma dissertação de mestrado, muitas das opções não são tratadas neste trabalho.

mesma plataforma eletrônica para diversos tipos de aplicações mecânicas. Ou seja, será buscado um sistema capaz de conter todas as formas de guiagem que possam contribuir para um comportamento autônomo. Após esta busca pela arquitetura do sistema completo, esta arquitetura de hardware será simplificada de forma a encontrarmos arquiteturas de hardware com o mínimo de blocos/módulos possíveis.

Juntando-se os objetivos acima, será proposto também um método de como alcançar o desenvolvimento das arquiteturas de hardware propostas.

Como objetivos intermediários foram estudados conceitos de autonomia e de inteligência artificial, métodos de localização e navegação e arquiteturas modulares existentes no mercado industrial e automotivo para uso em robótica móvel.

Resumidamente, os objetivos são listados abaixo:

- Estudar a definição de autonomia;
- Estudar métodos de localização;
- Estudar métodos de guiagem;
- Estudar tipos de arquiteturas de hardware (topologias, interfaces e conexões);
- Escolher uma plataforma e seus métodos de localização e guiagem;
- Verificar a aplicabilidade desta plataforma através de estudo de casos;
- Propor um método para a aplicação da plataforma em projetos.

1.3 Trabalhos Correlatos

Em relação à forma de execução e as escolhas deste trabalho, diversos outros trabalhos possuem relacionamento ou resultados semelhantes com este. Porém, com relação à motivação e arquitetura proposta são sugeridos os trabalhos abaixo:

O trabalho “Uma Plataforma Móvel para Estudos de Autonomia” (AUGUSTO, 2007) possui o mesmo objetivo. Mas, não se restringe a arquiteturas de hardware. Desta forma, seus resultados são em torno de uma arquitetura de software que possibilita a

modularidade de diversos algoritmos. Esta mesma arquitetura de software pode ser utilizada com a arquitetura de hardware proposta neste presente trabalho.

O trabalho “Arquitetura de Hardware e Software para Supervisão e Controle de um Carro Autônomo” (ARRUDA, 2012) utilizou-se de uma proposta semelhante, porém, se restringindo a um veículo automotivo autônomo. Neste caso, trata-se de uma particularidade deste trabalho.

Por fim, o trabalho “Implementação de uma rede CANOPEN para controle de veículos autônomos” (SILVA, 2011) implementou uma rede CANopen como arquitetura de hardware para veículos autônomos. Isto também seria uma particularidade deste presente trabalho visto que não restringimos a rede pela qual os dados trafegarão. Porém, as redes CAN e similares se mostraram muito adequadas para utilização em robôs móveis autônomos e são recomendadas neste trabalho, pois possuem vantagens em relação a outros tipos de redes. Tais vantagens serão discutidas ao longo deste trabalho.

1.4 Organização da Dissertação

Esta dissertação está organizada em 7 capítulos, sendo estes divididos em 4 blocos. O primeiro bloco se dedica a introdução e discussão sobre a motivação (capítulo 1) e conceitos de inteligência (capítulo 2). O segundo bloco se dedica a uma revisão bibliográfica sobre as capacidades mentais humanas desenvolvidas artificialmente; nele são vistos métodos de localização e guiagem, análise de erros (capítulo 3) e ainda um estudo sobre as interconexões em um sistema robótico (capítulo 4). O terceiro bloco se dedica a escolha de critérios e apresentação de resultados (capítulo 5) e ao estudo de casos sobre a plataforma escolhida (capítulo 6). O quarto e último bloco se concentram nas conclusões e no estudo da implantação dos projetos desta plataforma em um ambiente acadêmico e trabalhos futuros (capítulo 7).

Além destes, foram inseridos alguns apêndices utilizados como referências sobre erros em sistemas com navegação odométrica e uma análise de contramedidas para identificar e compensar erros odométricos (apêndice A), alguns detalhes a mais sobre

sistemas de navegação inercial (apêndice B), o barramento CAN (apêndice C); sobre baterias recarregáveis (apêndice D) e sugestões das mensagens CAN para cada módulo implementado (apêndice E).

2 Conceitos de Inteligência

2.1 Conceitos

O conceito de inteligência é fundamental para muitos estudos de diversas áreas do conhecimento (e.g. psicologia, psiquiatria, neurociência, inteligência artificial, robótica, cibernética, gestão do conhecimento etc.). Inteligência pode ser definida como “a totalidade das habilidades cognitivas do indivíduo, a resultante, o vetor final dos diferentes processos intelectivos” (DALGALARRONDO, 2000), ou seja, a soma de todas as atividades mentais.

Em robótica, o conceito de inteligência vem do conceito de inteligência artificial, que possui diversos conceitos na história da robótica. Simplificando e compondo livremente alguns conceitos apresentados por Russel e Norvig (2010), inteligência artificial se refere ao estudo de processos de pensamento e de razão utilizando-se de medidas de desempenho racionais ideais em comparação às humanas. Ou ainda, inteligência artificial é o campo de conhecimentos onde se estudam sistemas capazes de reproduzir (todas ou algumas das) atividades mentais humanas (NILLSON, 1986).

Autonomia, do grego antigo, se define como “aquele quem dá a si mesmo sua própria lei”. Autonomia é um conceito encontrado na moral, na política e na filosofia bioética. Com estes contextos, se refere à capacidade do indivíduo racional de tomar uma decisão com base em suas informações, sem ser coagido (1911 ENCYCLOPAEDIA BRITANNICA, 2012; WIKIPEDIA, 2012). Do conceito de autonomia vem o conceito de robô autônomo. Robôs autônomos são robôs que podem executar tarefas desejadas em ambientes não estruturados sem serem guiados continuamente por humanos (WIKIPEDIA, 2012).

Uma vez que o desenvolvimento destes conhecimentos (de inteligência) envolve a reprodução das atividades (ou faculdades) mentais humanas, cabe-se o estudo de tais atividades de forma a analisar e entender as suas formas de atuação e se buscar a autonomia em dispositivos móveis robóticos.

2.2 Atividades Mentais

Conhecidas por diversos nomes as Faculdades Mentais, Habilidades Mentais, Habilidades Cognitivas ou Atividades Mentais, são as atividades que nosso cérebro desenvolve de extrema importância para tornar-nos os seres desenvolvidos que somos. O estudo destas habilidades nos leva a conhecer-nos melhor e, para engenheiros, a desenvolver técnicas para se obter uma inteligência artificial semelhante. Segundo Yoneyama e Nascimento Jr. (2000) as principais atividades mentais conhecidas são apresentadas a seguir:

- Sensação – fenômeno elementar resultante de estímulos mecânicos, físicos, químicos ou elétricos sobre o organismo;
- Percepção – é a tomada de conhecimento de um objeto exterior considerado real. As sensações são integradas de modo que as condições internas e externas do organismo sejam conhecidas;
- Representação – é a organização das imagens de objetos. Corresponde ao processo de modelamento em engenharia;
- Conceituação – construção simbólica que busca captar a essência dos objetos, agrupando-os em classes;
- Juízo – é a capacidade de exprimir os vínculos e as relações entre os fatos e os objetos da natureza;
- Raciocínio – é a concatenação disciplinada de juízos;
- Memória – é a capacidade de armazenamento de informações para utilização posterior;
- Atenção – é a capacidade de concentrar a atividade psíquica sobre um dado estímulo que a solicita;
- Consciência (Cognição) – é um complexo de fenômenos psíquicos que permite conhecer a si próprio e ao mundo;

- Orientação – é a capacidade do indivíduo de ter consciência de sua situação temporal e espacial em relação ao meio;
- Afetividade – é a capacidade de experimentar sentimentos ou emoções;
- Volição – é o elemento psíquico que leva a iniciar e realizar atividades voluntárias;
- Linguagem – é o mecanismo que permite a expressão simbólica.

2.3 Composição da Inteligência

O ser humano, como um todo, age integralmente; ou seja, não existe forma de separar as atividades mentais humanas e defini-las como responsável ou totalmente independente. O que nos faz seres humanos é justamente o complexo conjunto de todas as atividades mentais. Isto é o que nos faz ser inteligentes. Porém, se é possível considerar, algumas das atividades mentais são mais importantes que outras para o desempenho de funções específicas. Ou seja, as atividades mentais atuam em pequenos conjuntos. Que podem ser mais ou menos importantes para cada área da vida do indivíduo. Por exemplo, para o ser humano se locomover (apenas se locomover sem destino pré-definido) são necessárias às atividades motoras do indivíduo, atividades de sensação/percepção (para identificar obstáculos) e atividade de orientação (para proporcionar equilíbrio mesmo em terrenos irregulares).

Semelhantemente aos seres humanos, em robôs é possível dividir as atividades de processamento em subcamadas. Para o desenvolvimento de uma navegação autônoma, são necessárias, principalmente (da mesma forma que em seres humanos), as atividades de processamento de Ações motoras, Sensação, Percepção, e Orientação. Porém, para adicionar outras possibilidades para o dispositivo robótico móvel ainda podem ser utilizados Representação (para identificação de objetos), Memória (reconhecimento de trajetos e objetos), Atenção (foco no que precisa ser processado mais rapidamente no

momento) e Consciência (monitoramento do estado do veículo por ele mesmo), que juntas podem auxiliar na formatação de ações mecânicas de movimento e seu controle.

Assim, o sucesso de uma atividade complexa depende do sucesso de várias outras competências. Por exemplo, em um robô em movimento, a Sensação se caracteriza pela interpretação dos dados extraídos dos sensores internos e externos; a Percepção se caracteriza em como o robô desenvolve os dados da Sensação para determinar a sua posição no espaço; a Orientação se caracteriza em como o meio pode ser representado e em como traçar caminhos neste ambiente; e a Cognição se caracteriza em como o robô é representado neste ambiente, ou seja, em quais graus de liberdade pode trabalhar e sua dimensão física em si. Por fim, a ação final do movimento se caracteriza em como o robô utiliza seus atuadores (motores, pistões, etc.) e os controla de forma a atingir os objetivos traçados na orientação/cognição. Isto coincide com a opinião de Siegwart e Nourbakhsh (2004).

Na seção a seguir, serão detalhados alguns conceitos de cada uma das atividades mentais citadas anteriormente e que são objetos de estudo em inteligência artificial.

2.4 Inteligência Artificial

A Tabela 1 mostra a importância do estudo de cada atividade mental para a engenharia e qual o estágio de desenvolvimento de sistemas e tecnologias que utilizam estes conceitos.

Tabela 1 – Estágios de desenvolvimento das faculdades mentais em máquinas.

Faculdade Mental	Definição neurocientífica*	Definição em Robótica	Aplicações em Robótica	Implementação em Robótica
Sensação	Fenômeno elementar gerado por estímulos físicos, químicos ou biológicos variados,	Sinais elétricos resultante de estímulos mecânicos, físicos,	Detecção de objetos, estáticos ou móveis, que interagem com o organismo. Tais como:	Hardware

Faculdade Mental	Definição neurocientífica*	Definição em Robótica	Aplicações em Robótica	Implementação em Robótica
	originados fora ou dentro do organismo, que produzem alterações nos órgãos receptores, estimulando-os.	químicos ou elétricos sobre a máquina;	obstáculos, cargas e outros organismos ou máquinas.	
Percepção	Tomada de consciência, pelo indivíduo, do estímulo sensorial.	Pode ser definido como o processamento analógico ou digital dos dados/sinais obtidos dos sensores. Na percepção, as sensações são processadas de modo que as condições internas e externas do sistema sejam conhecidas;	Processamento de todas as informações obtidas dos vários sensores presentes no organismo de forma a gerar informações úteis para o sistema.	Hardware
Representação	É a “re-apresentação” de uma imagem na consciência, sem a presença real, externa, do objeto que num primeiro momento gerou uma imagem sensorial.	Corresponde ao processo de modelamento em engenharia.	Em geral, são necessárias representações espaciais, através de modelos matemáticos de objetos (sejam obstáculos ou caminhos ou mesmo a carga a ser transportada) de forma a tornar o algoritmo de localização/orientação determinístico quando comparamos o que foi detectado com o evento esperado.	Software ou hardware
Conceituação	É o elemento estrutural básico do pensamento, nele se exprimem os caracteres essenciais dos objetos e fenômenos da natureza. Cada entidade presente na natureza requer um conceito.	Construção simbólica que busca captar a essência dos objetos, agrupando-os em classes.	Complementa o processo de representação agrupando os possíveis objetos em classes. Isto torna possível a tomada de decisão. Finalizando com as ações previstas no sistema.	Software
Juízo	É o processo que conduz ao estabelecimento de relações significativas	Caracteriza-se como a relação entre classes de objetos ou	Profundamente ligado à conceituação, o juízo adiciona recursos à tomada de decisão.	Software

Faculdade Mental	Definição neurocientífica*	Definição em Robótica	Aplicações em Robótica	Implementação em Robótica
	entre os conceitos básicos. O juízo consiste, a princípio, na afirmação de uma relação entre dois conceitos.	conceitos.		
Raciocínio	É a função que relaciona os juízos.	É a concatenação disciplinada de juízos. Utilizando-se da lógica.	O raciocínio é a especialidade das máquinas. Desde que lógica é uma das bases da engenharia, é improvável que uma máquina obtenha conclusões erradas em uma concatenação de juízos.	Software
Memória	É a capacidade de registrar, manter e evocar os fatos já ocorridos.	Refere-se à capacidade de armazenamento e acesso das informações.	Necessária em qualquer sistema robótico, a memória se caracteriza como item principal de catalogação e referência. Com a memória é possível armazenar dados referentes a mapas e/ou informações de objetos.	Hardware + Software
Consciência	É a soma total das experiências conscientes de um indivíduo em um determinado momento. A capacidade de tomar ciência dos deveres éticos e assumir as responsabilidades, os direitos e deveres concernentes a essa ética.	É o complexo de fenômenos processados que permite a máquina saber o estado de si mesma, do ambiente onde está inserida e sua influência no ambiente em que está inserida.	É frequentemente tratado como o “estado” da máquina. Quando tudo está definido e localizado é possível tomar decisões. Sua complexidade depende diretamente do nível de consciência necessário para a tomada de decisão desejada.	Software
Atenção	É a “direção da consciência”, o estado de concentração da atividade mental sobre determinado objeto. (CUVILLIER, 1937)	É a capacidade de concentrar a atividade de processamento sobre um dado estímulo que a solicita.	Esta atividade é necessária para acelerar o processamento de dados filtrando somente os dados necessários para a aplicação desejada.	Hardware + Software
Orientação	É a capacidade de situar-se quanto a si mesmo e ao ambiente.	É a capacidade de o mecanismo ter consciência de sua situação temporal e espacial em relação	Necessário para qualquer máquina móvel autônoma. Para poder se movimentar conscientemente, o robô tem que conhecer seu	Hardware + Software

Faculdade Mental	Definição neurocientífica*	Definição em Robótica	Aplicações em Robótica	Implementação em Robótica
		ao meio.	estado atual em relação ao espaço e ao tempo.	
Afetividade	É a dimensão psíquica que dá cor, brilho e calor a todas as vivências humanas.	É a capacidade de experimentar sentimentos ou emoções.	De opinião controversa nos estudos. Frequentemente este comportamento aparece em obras de ficção científica.	Software
Volição	É a dimensão complexa da vida mental, relacionada intimamente à esfera instintiva e afetiva, assim como à esfera intelectual (avaliar, julgar, analisar, decidir) e ao conjunto de valores, princípios, hábitos e normas socioculturais do indivíduo.	É o elemento psíquico que leva a iniciar e realizar atividades voluntárias independentes dos julgamentos pré-concebidos no sistema.	De opinião controversa nos estudos. Frequentemente este comportamento aparece em obras de ficção científica.	Software
Linguagem	É o ato de representar o pensamento, afetividade, volição, e outras atividades mentais.	É o mecanismo que permite a expressão simbólica do sistema.	É a forma amplamente utilizada para acompanhar o funcionamento do sistema.	Hardware + Software

Fonte: * (YONEYAMA e NASCIMENTO JR., 2000; DALGALARRONDO, 2000).

Desta forma, um robô autônomo deve possuir algumas atividades mentais sintetizadas artificialmente em seu processamento de forma a funcionar autonomamente. A principal atividade mental presente em todos os robôs autônomos é a consciência, por definição. Segundo Yoneyama e Nascimento Jr. (2000), “Consciência é um conjunto complexo de fenômenos físicos que permitem o conhecimento de si próprio e do mundo ao redor”. Mas, isto só é possível quando outras atividades mentais estão presentes como a sensação, percepção e orientação. Estas atividades mentais são responsáveis por informar o corpo humano sobre sua posição em determinado tempo e no ambiente, tanto internamente como externamente. Ou seja, estas atividades mentais fornecem as informações necessárias para atingir algum nível de consciência.

O capítulo seguinte procura descrever melhor alguns métodos para as atividades de Sensação/Percepção e Orientação que foram estudados para o desenvolvimento da proposta de arquitetura de *hardware* e dos objetivos deste trabalho. Em seguida será tratado como interconectar estas atividades em um dispositivo robótico móvel de forma a atingir a autonomia.

3 Atividades Mentais Artificiais

Primeiramente, vamos buscar entender melhor as faculdades ou atividades mentais de Sensação e Percepção aplicadas em um ambiente robótico. Em seguida, exploraremos as atividades de Orientação que em robótica pode ser chamada de guiagem ou, num sentido mais amplo, navegação.

3.1 Sensação/Percepção

O ser humano possui cinco sentidos; a saber:

- Visão – responsável pela entrada de imagens no corpo humano;
- Audição – responsável pelo equilíbrio e pela recepção de informações sonoras;
- Tato – responsável pelo controle de contato e pelo controle de temperatura do corpo;
- Olfato – responsável pela identificação de qualidade de alimentos através de impulsos elétricos diferentes para cada cheiro específico; e
- Paladar – responsável pela identificação das qualidades dos alimentos ingeridos.

Estes sentidos são responsáveis pela entrada de todos os sinais externos ao nosso corpo. A partir destes sensores, o corpo humano processa todas as nossas capacidades de representação, conceituação e orientação.

Em eletrônica existem alguns sensores que tentam imitar estes sistemas sensoriais biológicos visando dar mesmas capacidades a veículos móveis. A seguir, vamos examinar cada uma delas.

3.1.1 Sensores

Segundo Siegwart e Nourbakhsh (2004), os sensores nos seres humanos podem ser classificados como proprioceptivos e exteroceptivos. Propriocepção foi o termo criado pelo neurofisiologista e prêmio Nobel Charles Sherrington em 1906 para referenciar a capacidade de reconhecer a localização espacial do corpo, sua posição e orientação, a força exercida pelos músculos e a posição de cada parte do corpo em relação às demais, sem utilizar a visão. Assim, sensores proprioceptivos medem valores internos ao sistema; por exemplo, velocidade do motor, carga nas rodas, ângulos nas junções de braços, tensão da bateria, etc. O sistema exteroceptivo é um conjunto de receptores sensitivos formados por órgãos terminais sensitivos especiais distribuídos pela pele e as mucosas que recebem os estímulos de origem exterior e os nervos aferentes que levam a informação sensitiva aferente ao sistema nervoso central. O sistema exteroceptivo recebe estímulos externos ao corpo, ao contrário do sistema proprioceptivo que recebe os estímulos que procedem do interior do corpo. Os sensores exteroceptivos obtêm informações do ambiente onde o robô está; por exemplo, medidas de distâncias, intensidade de luz, amplitude sonora. De forma que, as medidas obtidas de sensores exteroceptivos são interpretadas pelo robô de forma a obter características significativas do ambiente.

Quanto a sua aplicação, a principal característica que podemos considerar em um sensor é a do tipo de resposta (relativos ou absolutos). Esta classificação restringirá ou indicará qual a forma a ser usada. Sensores relativos ou absolutos recebem esta denominação referente ao tipo de resposta obtida, se esta é relativa à outra medida de referência ou pode ser utilizada como referência absoluta do sistema.

Os sensores de medidas relativas se caracterizam pela existência de uma referência de onde todas as medidas de posição partem. Os métodos de orientação que derivam deste tipo de sensores são a odometria e a navegação inercial. No método da odometria é possível, através de *encoders* solidários às rodas ou aos mecanismos de movimento, determinar a distância percorrida e com isto a posição atual. No método de navegação inercial a posição relativa é determinada com o uso de acelerômetros e giroscópios.

Os métodos de guiagem relativa são muito atraentes para a robótica móvel, pois dispensam o uso de equipamentos externos ao veículo. Porém, os erros mecânicos (sistemáticos ou não) e de quantização podem resultar em uma localização imprecisa necessitando de análises e correções nas leituras através de estimativas absolutas de posição (BORENSTEIN, EVERETT e FENG, 1996).

Os sensores de medidas absolutas se caracterizam por fornecer a informação da posição do veículo em relação a uma referência estática. Os métodos de orientação que derivam deste tipo de sensores são os que utilizam sinais de referência, os que utilizam reconhecimento de padrões/marcas e os que utilizam mapas, vídeo, dentre outros.

A Tabela 2 sintetiza as vantagens e desvantagens da utilização de sensores de medidas relativas e absolutas em robôs móveis.

Tabela 2 – Tabela síntese das características de sensores de medidas relativas e absolutas.

Sensores	Vantagens	Desvantagens
<i>Medidas Relativas</i>	Dispensam componentes externos.	Erros são difíceis de serem corrigidos.
<i>Medidas Absolutas</i>	Posição real; Erros minimizados; Malha fechada.	Necessitam de componentes externos.

Sensores ainda podem ser classificados quanto sua ação (ativos ou passivos). Sensores passivos medem a energia do ambiente que entra no sensor. Exemplos de sensores passivos incluem termômetros, microfones, e câmeras CCD e CMOS. Sensores ativos inserem energia no ambiente, e então medem a reação do ambiente. Visto que sensores ativos podem gerenciar interações mais controladas com o ambiente, eles frequentemente alcançam desempenho superior. Embora os sensores ativos acrescentem o risco de a energia emitida poder afetar muitas características do que está sendo medido. Ainda assim, eles são bastante importantes quando se deseja medir grandezas de pequena intensidade. Além disso, um sensor ativo pode sofrer interferências entre o sinal e outras fontes além do seu controle. Por exemplo, sinais emitidos por outros robôs próximos, ou sensores similares

no mesmo robô, podem influenciar nos resultados das medidas. Exemplos de sensores ativos incluem *encoders* de rodas, sensores de ultrassom, e medidores de distância a laser.

A Tabela 3 sintetiza os diversos sensores disponíveis no mercado e suas características.

Tabela 3 – Classificação dos sensores utilizados em aplicações de robótica móvel.

<i>Classificação Geral</i>	<i>Utilização</i>	<i>Sensor</i>	<i>Proprioceptivo/ Exteroceptivo</i>	<i>Relativo/Absoluto</i>	<i>Passivo/Ativo</i>
Sensores táteis	Detecção de contato físico ou proximidade; chaves de segurança	Chaves de contato;	Exteroceptivo	Absoluto	Passivo
		Chaves de fim de curso;	Exteroceptivo	Absoluto	Passivo/Ativo
		Sensores de proximidade;	Exteroceptivo	Absoluto	Passivo/Ativo
		Barreiras ópticas.	Exteroceptivo	Absoluto	Ativo
Sensores de motores	Detecção de velocidade e posição através do motor/rodas	Encoder de escovas	Proprioceptivo	Relativo	Passivo
		Potenciômetros	Proprioceptivo	Absoluto	Passivo
		Synchros, Resolvers	Proprioceptivo	Relativo	Ativo
		Encoders Ópticos	Proprioceptivo	Relativo/Absoluto	Ativo
		Encoders Magnéticos	Proprioceptivo	Relativo	Ativo
		Encoders Indutivos	Proprioceptivo	Relativo	Ativo
		Encoders Capacitivos	Proprioceptivo	Relativo	Ativo
Sensores de direção	Orienta o robô em relação a um ponto fixo de referência	Bússola	Exteroceptivo	Absoluto	Passivo
		Giroscópios	Proprioceptivo	Relativo	Passivo
		Inclinômetros	Exteroceptivo	Relativo	Ativo/Passivo
Sensores baseados em sinais de referência (<i>beacon</i>)	Detecção de localização através de mensagens fixas de referência	GPS	Exteroceptivo	Absoluto	Ativo
		Óptica ativa ou RF beacons	Exteroceptivo	Absoluto	Ativo
		Active Ultrasonic beacons	Exteroceptivo	Absoluto	Ativo
		Reflective beacons	Exteroceptivo	Absoluto	Ativo

<i>Classificação Geral</i>	<i>Utilização</i>	<i>Sensor</i>	<i>Proprioceptivo/ Exteroceptivo</i>	<i>Relativo/Absoluto</i>	<i>Passivo/Ativo</i>
Sensores de distância ativos	Detecção de distância por reflexão, tempo e triangulação.	Sensores de reflectividade	Exteroceptivo	Absoluto	Ativo
		Sensor de ultrassom	Exteroceptivo	Absoluto	Ativo
		Laser rangefinder	Exteroceptivo	Absoluto	Ativo
		Triangulação óptica (1D)	Exteroceptivo	Absoluto	Ativo
		Luz estruturada (2D)	Exteroceptivo	Absoluto	Ativo
Sensores de Movimento	Detecção de velocidade em relação a objetos fixos ou móveis	Radar Doppler	Exteroceptivo	Relativo	Ativo
		Som Doppler	Exteroceptivo	Relativo	Ativo
Sensores de Visão	Detecção de alcance visual, análise de imagem, segmentação e reconhecimento de objetos	Câmara CCD/CMOS	Exteroceptivo	Absoluto	Passivo

Fonte: (SIEGWART e NOURBAKHS, 2004).

Existem ainda estudos de sensores de olfato e paladar, para degustação de bebidas e outras substâncias, sensores químicos para medição da qualidade ar ou níveis de poluição. Tais sensores não serão tratados neste trabalho.

3.1.2 Erros em Sensores

Quando se trabalha com aquisição de dados em geral, seja a partir de sensores ou de outros sistemas, sempre é necessário analisar os erros inseridos nas interfaces do sistema. Existem dois principais tipos de erros: os sistemáticos e os não sistemáticos. Cada um pode inserir diferentes tipos de erro ao sistema.

3.1.2.1 Erros sistemáticos

Em robótica, existem diversos erros conhecidos e que devem ser analisados em cada projeto mecânico. Porém, o estudo dos erros é especialmente importante em métodos

de cálculo estimado (*dead-reckoning*) devido a sua dependência da qualidade dos dados para uma estimativa correta. Assim, estes erros serão melhores explicados a seguir.

Conforme o próprio nome indica, um erro sistemático é um erro intrínseco ao sistema. Assim, estão relacionados à geometria do robô, a forma de cálculo da posição, etc. Porém, conforme o uso do dispositivo robótico, desgastes podem ocorrer e assim, mudanças em seus erros sistemáticos. Desta forma, é recomendado calibrar o sistema frequentemente através de um método de calibração comprovadamente eficaz para o modelo cinemático utilizado (BORENSTEIN, EVERETT e FENG, 1996). Os tipos de erros mais comuns que ocorrem em sensores são ilustrados na Figura 6.

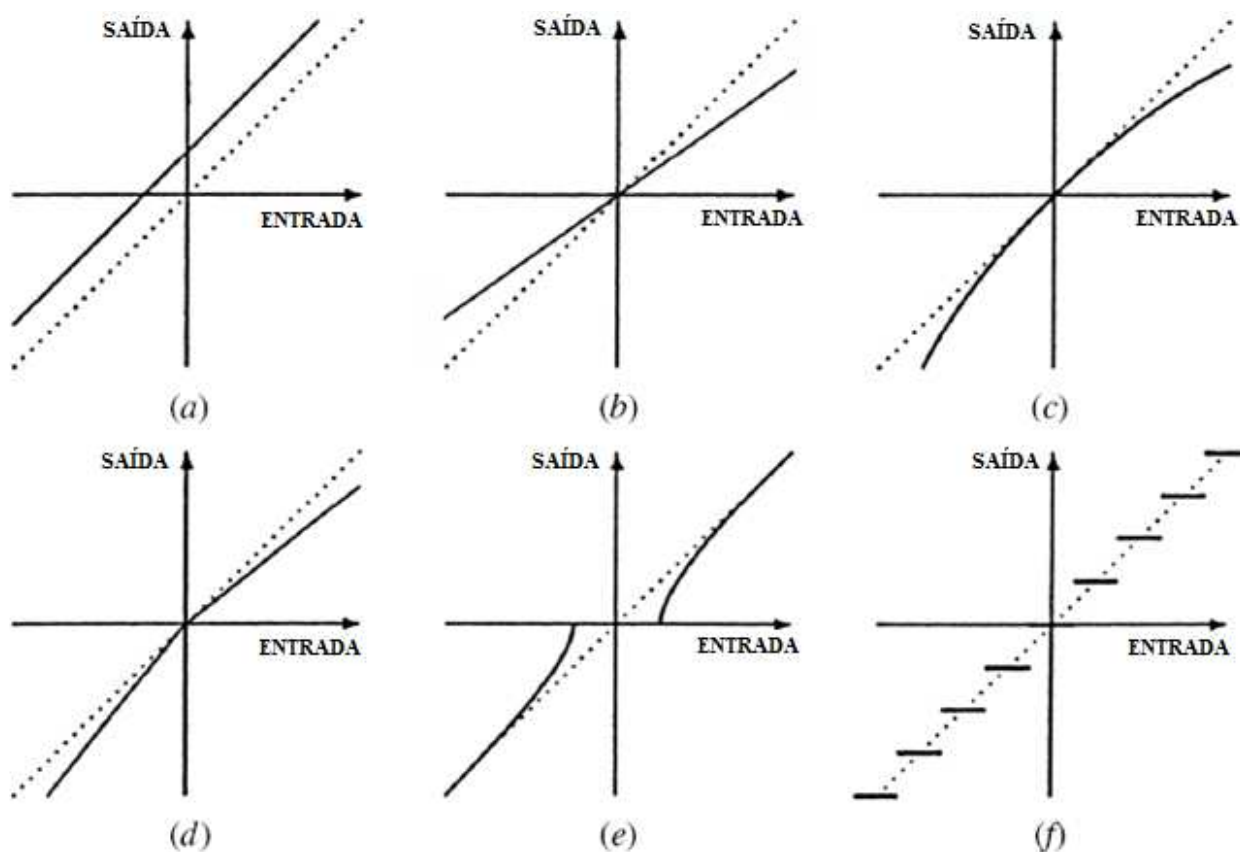


Figura 6 – Tipos de erros comuns. (a) Bias; (b) Fator de Escala; (c) Não-linearidade; (d) Assimetria; (e) Zona Morta; (f) Quantização.

Fonte: (GREWAL, WEILL e ANDREWS, 2007).

Segundo Borenstein, Everett e Feng (1996), os causadores de erros sistemáticos mais comuns nos veículos móveis são:

- Diâmetros de rodas diferentes;
- Distância entre eixos diferente da distância nominal;
- Desalinhamento de rodas;
- Resolução do *encoder*;
- Taxa de amostragem do *encoder*.

Em outros tipos de veículos móveis, os causadores de erros sistemáticos podem ser:

- Desbalanceamento de asas ou hélices;
- Eixos paralelos desalinhados em robôs com pernas;
- Sujeira ou perda de força em somente uma perna do robô;
- Profundidade de cursos diferentes em acionadores paralelos (e.g. ailerons, flaps, pernas de robôs, etc).

3.1.2.2 Erros não sistemáticos

Erros não sistemáticos são os erros que não são gerados pelo sistema robótico. Assim, são erros relacionados ao ambiente de operação do robô móvel. Sua dimensão e impacto sobre a localização do sistema são imprevisíveis.

Segundo Borenstein, Everett e Feng (1996), os causadores de erros não sistemáticos mais comuns em veículos móveis são:

- Terrenos não planos;
- Objetos inesperados no chão;
- “Patinação” das rodas devido a: Chão escorregadio; Aceleração/desaceleração bruscas; Mudança de trajetória muito rapidamente; Forças externas; Forças internas; Roda sem contato com o chão.

Em outros tipos de veículos móveis, os causadores de erros não sistemáticos podem ser:

- Rajadas de vento;

- Chuva;
- Impacto de outro dispositivo.

3.1.2.3 Resumo

Sejam erros sistemáticos, que por estarem sempre presentes no sistema e podem ser previstos e corrigidos por calibrações, até mesmo erros não sistemáticos, que podem comprometer a análise dos dados, os erros sempre devem ser considerados. A Figura 7 mostra o caminho que um estímulo externo percorre até ter uma resposta dos atuadores e os erros que o sistema está sujeito neste caminho. Nesta figura, o caminho do estímulo externo até os atuadores passa pela estrutura mecânica, pelos sensores, passa pelo conversor analógico/digital que entrega os dados obtidos para o microcontrolador do sistema. O microcontrolador processa os dados e segundo a sua malha de controle comanda o acionamento do atuador. O comando passa então pelo conversor digital/analógico e chega no atuador que transfere a energia para a mecânica do veículo.

Neste exemplo descrito na Figura 7 surgem erros sistemáticos e não-sistemáticos. A figura mostra alguns dos principais erros que podem surgir durante as atividades de sensação e atuação de um sistema.

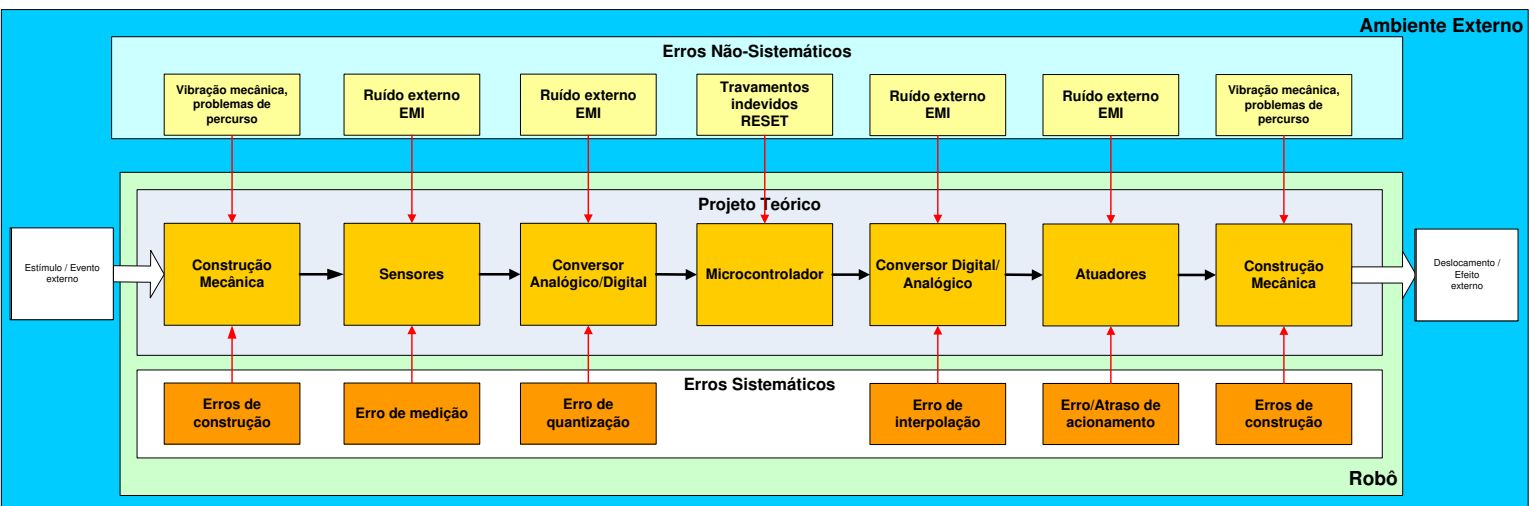


Figura 7 – Sistema robótico considerando erros possíveis.

3.1.3 Fusão Sensorial

O conceito de fusão (multi)sensorial é relativamente novo quando comparado com as outras áreas da engenharia. Quando humanos e animais evoluíram, eles desenvolveram a habilidade de uso de múltiplos sensores como auxílio à sobrevivência.

Nas últimas duas décadas, muita atenção tem sido dada a fusão (multi)sensorial tanto para aplicações militares como civis. Técnicas de fusão de dados combinam dados de múltiplos sensores e as informações relacionadas de forma a alcançar inferências mais específicas que poderiam ser encontradas utilizando-se um único sensor individual. Fusão de dados se refere a combinação de dados de múltiplos sensores (de mesmo tipo ou de diferentes tipos), onde a fusão de informação se refere a combinação de dados dos sensores, relatórios humanos, banco de dados, etc. A tecnologia de fusão sensorial, tem rapidamente avançado a partir de uma desorganizada coletânea de técnicas a uma emergente disciplina de engenharia com terminologia padronizada, técnicas matemáticas robustas, e princípios de projeto de sistema estabelecidos. (LIGGINS, HALL e LLINAS, 2009)

Aplicações de fusão multissensorial são amplas. Tais aplicações são listadas simplificadaamente na Tabela 4.

Tabela 4 – Aplicações representativas de fusão sensorial não militares.

Aplicações Específicas	Inferências feitas por processos de fusão sensorial	Dados primários observáveis	Volume de vigilância	Plataformas dos sensores
<i>Manutenção situacional</i>	Deteção, caracterização de falhas do sistema, recomendações de manutenção/correção	Sinais eletromagnéticos, sinais acústicos, magnéticos, temperaturas, raios X, restos de lubrificantes, vibração	De Microscópico a alguns metros	Navios, aviões, terrestres (e.g. fábricas)
<i>Robótica</i>	Localização/Reconhecimento de objetos, guia para movimentação do robô (e.g. “mãos” e “pés”)	Televisão, sinais acústicos, Sinais eletromagnéticos, raios X	De microscópicos a alguns metros sobre o robô.	Corpo do robô
<i>Diagnóstico médico</i>	Localização/Identificação de tumores, anormalidades, e doenças	Raios X, Ressonância magnética,	Volume do corpo humano	Laboratório

Aplicações Específicas	Inferências feitas por processos de fusão sensorial	Dados primários observáveis	Volume de vigilância	Plataformas dos sensores
		temperatura, infravermelho, inspeção visual, dados químicos e biológicos, auto relatos de sintomas por humanos.		
Monitoramento ambiental	Identificação/localização de fenômenos naturais (e.g. terremotos, clima)	Radares SAR, sísmicos, radiação eletromagnética, amostras de núcleo, dados químicos e biológicos	Centenas de quilômetros (monitoramento de locais)	Satélites, aviões, terrestres, amostras subterrâneas

Fonte: (LIGGINS, HALL e LLINAS, 2009).

Segundo Liggins, Hall e Llinas (2009), fusões de dados a partir de múltiplos sensores proporcionam três vantagens principais sobre dados obtidos de um único sensor, que são listadas a seguir:

1. Vantagem estatística: se vários sensores idênticos são utilizados, combinando as observações resultariam em uma estimativa melhorada de resultados.
2. Melhoria no processo de observação: utilizando o posicionamento relativo ou movimento de múltiplos sensores o processo de observação pode ser melhorado cobrindo uma maior área.
3. Aumento de observabilidade: ampliando a base de observáveis físicos pode resultar em melhoras significativas na observação do sistema.

Além disto, três possibilidades de fusão funcionam como alternativas básicas que podem ser utilizadas para fusão de dados multissensoriais. As três possibilidades são listadas abaixo:

- Fusão direta dos dados dos sensores. Nesta construção, os dados são combinados ao saírem dos sensores. Esta combinação é mostrada no esquema da Figura 8;

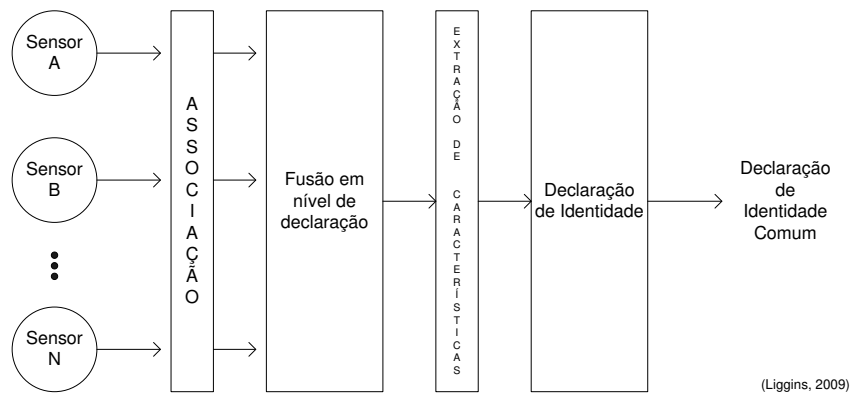


Figura 8 – Fusão direta de dados.

Fonte: (LIGGINS, HALL e LLINAS, 2009).

Se os dados dos vários sensores são proporcionais, isto é, se os sensores são mensuráveis com o mesmo fenômeno físico (dois sensores de imagens ou dois sensores acústicos) então os dados brutos dos sensores podem ser diretamente combinados. Técnicas para fusão de dados brutos de sensores, tipicamente envolvem métodos de estimativas clássicas como o filtro Kalman. Reciprocamente, se os dados dos sensores são não-proporcionais então os dados devem ser fundidos ao nível de vetores de características/estados ou de decisões.

- Representação dos dados dos sensores através de vetores, com subsequente fusão dos vetores. Esta opção é mostrada no esquema da Figura 9; ou

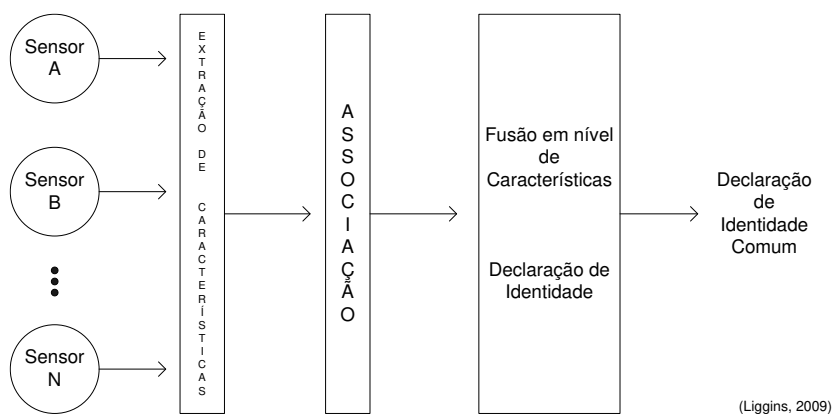


Figura 9 – Representação dos dados por vetores com subsequente fusão dos vetores.

Fonte: (LIGGINS, HALL e LLINAS, 2009).

Fusões ao nível de características envolvem a extração de características representativas a partir dos dados dos sensores. Um exemplo de extração de características é percebido quando um cartunista utiliza de características faciais chaves para representar o rosto humano. (LIGGINS, HALL e LLINAS, 2009).

Evidências confirmam que humanos utilizam uma função cognitiva baseada em características para reconhecer objetos. No caso de fusão multissensorial no nível de características, são extraídas características das observações de múltiplos sensores e combinadas em um único vetor de características concatenadas que é uma entrada para técnicas de reconhecimento de padrões tal como redes neurais, algoritmos de agrupamento (*clustering algorithms*) ou algoritmos de métodos abstratos (*template methods*).

- Processando cada sensor para alcançar inferências de alto-nível ou decisões, que podem ser combinadas posteriormente. Esta fusão de dados acontece em nível decisório e é mostrada no esquema da Figura 10.

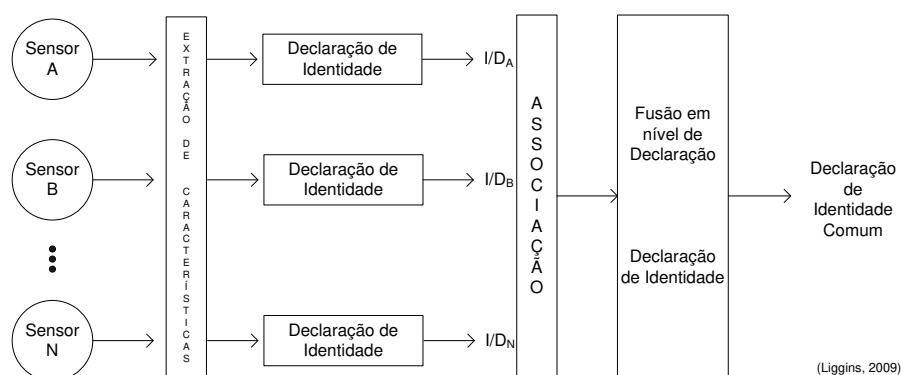


Figura 10 – Processamento individual de cada sensor com subsequente combinação de resultados.

Fonte: (LIGGINS, HALL e LLINAS, 2009).

Fusão em nível de decisão combina a informação do sensor depois que cada sensor executou uma determinação preliminar da localização, atributos e identidade da “entidade”. Exemplos de métodos de fusão em nível de decisão incluem métodos de sistema de votação (*voting techniques*), inferência clássica, inferência Bayesiana e o método de “*Dempster-Shafer*”.

Mais informações sobre fusão (multi)sensorial podem ser encontradas no livro de Liggins, Hall e Llinas (2009) intitulado “Handbook of Multisensor Data Fusion. Theory and Practice”.

3.1.3.1 Filtro de Kalman

O filtro de Kalman é, essencialmente, um estimador para o chamado “problema linear-quadrático”. Idealizado por Rudolf Emil Kalman o filtro Kalman digitaliza a ideia de previsibilidade presente no filtro analógico *Wiener-Kolmogorov* (de Norbert Wiener e Andrei Nikolaevich Kolmogorov).

A primeira referência de uso do filtro Kalman foi no projeto Apollo da NASA na estimativa de trajetória do problema de controle. Desde então o filtro Kalman e suas variações é frequentemente utilizado em problemas de controle.

Em robótica, o filtro Kalman adquiriu uma função especial no contexto de fusões sensoriais. Usado para combinar dados obtidos de diferentes sensores em uma estimativa estatisticamente ótima. Se um sistema pode ser descrito através de uma modelo linear e as incertezas dos sensores e do sistema podem ser modelados como ruídos Gaussianos brancos, então o filtro de Kalman proverá uma estimativa estatisticamente ótima para os dados fundidos. Isto significa que, sob certas condições, o filtro de Kalman é capaz de encontrar a melhor estimativa baseada na correção de cada medida individual.

Ainda, é possível encontrar diversos estudos de variantes do filtro de Kalman. Kalman extendido, kalman-bucy, dentre outros povoam o desenvolvimento de dispositivos robóticos móveis no mundo.

3.1.4 Robótica Probabilística

Uma das formas de se tratar os erros de aquisição de dados de sensores é de não utilizar os dados propriamente ditos. Em vez disto, pode-se utilizar a densidade de probabilidade da ocorrência dos dados e suas possíveis consequências para o veículo robótico. A vantagem deste tipo de abordagem é que ocorre uma redução de processamento para algumas aplicações. Esta forma de trabalhar com os dados é chamada de Robótica Probabilística e não será tratada neste trabalho. Mais informações sobre a Robótica

Probabilística podem ser consultadas no livro de Sebastian Thrun, Wolfram Burgard e Dieter Fox intitulado “*Probabilistic Robotics*” (THRUN, BURGARD e FOX, 2006).

3.2 Orientação

A partir dos sensores disponíveis, é possível obter informações sobre o estado do veículo, localizá-lo e utilizar estes dados de forma a permitir uma decisão sobre qual trajetória deve-se tomar para chegar ao objetivo.

A guiagem pode ser buscada em um projeto de forma determinística ou probabilística. Existem diversos métodos/modelos tanto determinísticos como probabilísticos, porém, como o nosso trabalho está baseado em arquiteturas de *hardware* iremos tratar especialmente os métodos determinísticos. Visto que, com o *hardware* de um sistema de navegação determinístico, é possível utilizar algoritmos probabilísticos e assim potencializar a utilização dos dados obtidos. Nas seções a seguir são discutidos alguns métodos determinísticos para guiagem.

3.2.1 Táctil

Guiagem táctil se refere a qualquer método que explore o contato físico como forma de identificar obstáculos. Assim, neste método são colocados sensores de impacto, sensores de contato, ou mesmo sensores de proximidade no corpo do veículo móvel. Quando os sensores detectam algum contato (ou proximidade) o controlador identifica que não é possível prosseguir por aquele caminho. E assim, deverá mudar de direção/trajetória nos passos seguintes.

Guiagem táctil se refere a qualquer método que depende de proximidade para se localizar. É um método simples onde chaves, sensores ópticos ou sensores de proximidade identificam obstáculos e algoritmos tomam decisões sobre como prosseguirão através dos obstáculos. Devido a simplicidade deste método, ele é vastamente utilizado na indústria em

geral, inclusive em brinquedos comerciais. A Figura 11 mostra alguns dos sensores utilizados neste tipo de guiagem.

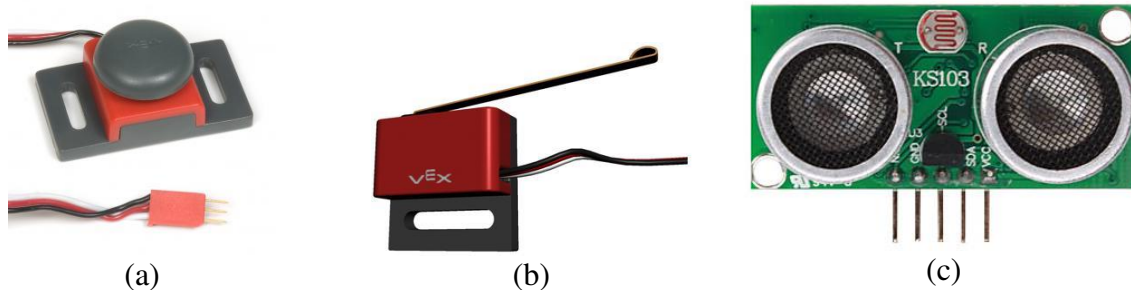


Figura 11 - Sensores utilizados em guiagem táctil. (a) chave de contato; (b) chave de fim de curso; (c) detector de distância por ultrassom.

Fonte: (VEX ROBOTICS INC., 2012-2013)

3.2.2 Odométrica

A navegação odométrica, também chamada de odometria, é um tipo de navegação por cálculo estimado. Sabe-se que a odometria proporciona uma boa precisão em curto prazo, é de baixo custo de implantação, e requer taxas de amostragem muito altas. A navegação odométrica é um dos métodos mais utilizados para estimar a posição de um robô (BORENSTEIN, EVERETT e FENG, 1996).

A ideia fundamental da odometria é a integração da informação de incremento do movimento (obtido através de *encoders* ou de outros sensores de movimento) ao longo do tempo, o qual envolve uma inevitável acumulação de erros. A acumulação de erros de orientação causa grandes erros na estimação da posição, os quais vão aumentando com a distância percorrida pelo robô. Apesar desta limitação, muitos pesquisadores continuam utilizando a odometria como uma parte importante do sistema de navegação de um robô. Porém, devem-se usar outras medidas de posicionamento absolutas combinadas a odometria para proporcionar uma estimativa de posição mais confiável. O Apêndice A mostra mais detalhes sobre os erros na odometria e algumas formas de reduzir tais erros que são encontradas na literatura especializada.

As equações da navegação odométrica são bastante simples e dependem basicamente da geometria do movimento do robô. Desta forma, os movimentos das rodas

podem ser convertidos em deslocamento linear relativo ao chão. A Figura 12 mostra detalhes sobre o disco utilizado coerentemente com o eixo da roda que vai ser “monitorada” pela odometria.

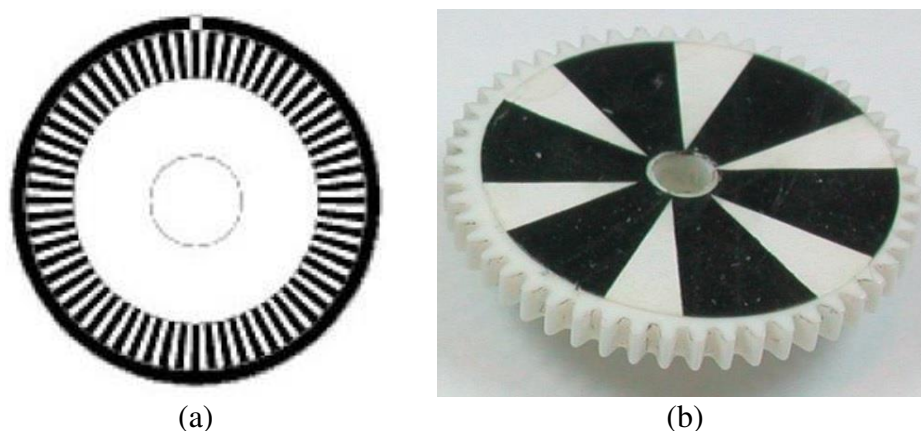


Figura 12 – *Encoder* óptico incremental com sinais em quadratura; (a) desenho esquemático de um disco óptico incremental; (b) foto de um disco óptico incremental utilizado no robô Asuro.

Fonte: (SIEGWART e NOURBAKHSH, 2004; BORENSTEIN, EVERETT e FENG, 1996).

Para determinação da direção de giro, utilizam-se dois fotorreceptores deslocados lateralmente pela distância equivalente a metade de cada marcação do disco. A Figura 13 mostra os pulsos obtidos dos fotorreceptores.

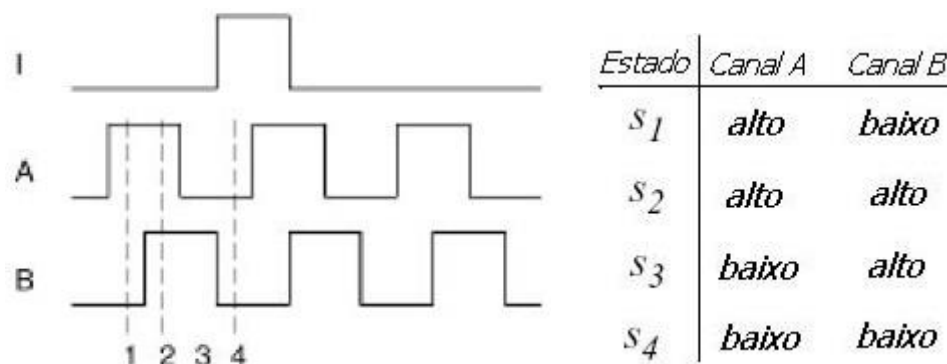


Figura 13 – Pulsos obtidos nas saídas dos fotorreceptores.

Fonte: (SIEGWART e NOURBAKHSH, 2004; BORENSTEIN, EVERETT e FENG, 1996).

Abaixo estão as equações utilizadas para determinação da distância percorrida (ΔS) pelo veículo móvel contabilizada pelo número de pulsos detectados pelos fotorreceptores (n). Isto é demonstrado pelas equações (1), (2) e (3) abaixo.

$$S(n) = \frac{n * (2 * \pi * r)}{N} + S_0 \quad (1)$$

$$k = \frac{2 * \pi * r}{N} \quad (2)$$

$$\Delta S = k * n \quad (3)$$

Onde: $\begin{cases} r = \text{Raio da roda do veículo} \\ n = \text{Número de pulsos recebidos} \\ N = \text{Número de furos no disco} \\ S_0 = \text{Posição anterior do veículo} \end{cases}$

A partir das informações de variação de posição no tempo se obtém informações de velocidade média do trecho que pode ser representada pela equação (4).

$$Vm = \frac{\Delta S}{\Delta t} \quad (4)$$

Onde: $\begin{cases} Vm = \text{Velocidade média do veículo no trecho} \\ \Delta S = \text{Variação de espaço ou distancia percorrida} \\ \Delta t = \text{Tempo gasto para percorrer } \Delta S \end{cases}$

A Figura 14 mostra o robô Asuro que utiliza a odometria como forma principal de localização ou contabilização de distâncias percorridas. O robô Asuro possui disco óptico incremental nas duas rodas para utilizar odometria como forma principal de determinação das distâncias percorridas. Em (a) e (b) são mostradas as vistas de topo e lateral do robô respectivamente; e em (c) a vista em detalhamento dos fotorreceptores e dos discos ópticos.

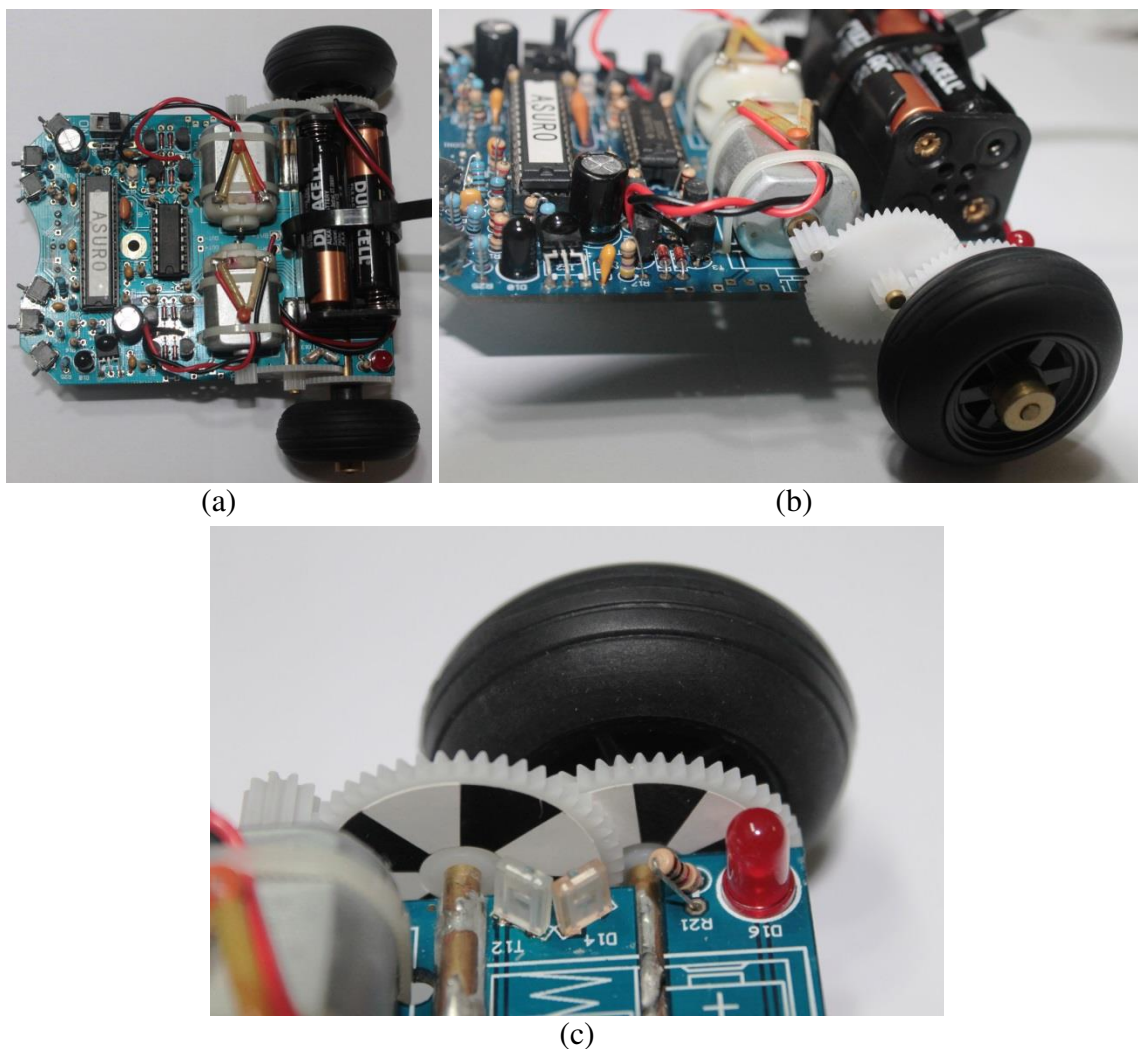


Figura 14 – Robô Asuro.

Fonte: (DLR GERMANY AND AREXX - THE NETHERLANDS, 2006)

3.2.3 Inercial

Assim como a odométrica, a guiagem inercial é um tipo de navegação por cálculo estimado. Inicialmente desenvolvida para o uso em aeronaves, esta tecnologia foi rapidamente adaptada para uso em mísseis, missões espaciais, submarinos nucleares e posteriormente em mecanismos robóticos/celulares/tablets.

Também chamada de navegação Newtoniana, o princípio de operação envolve a medida contínua de acelerações em cada um dos três eixos direcionais, integrando no

tempo para encontrar a velocidade e a posição em relação ao seu próprio eixo de coordenadas (*eixo inercial*). Giroscópios são utilizados para obter uma referência dos três acelerômetros em todo o processo com relação ao eixo inercial referente a atitude do eixo.

O processo de cálculo estimado na navegação inercial pode ser pensado como a habilidade de um passageiro com os olhos vendados dentro de um carro sentir-se pressionando o assento quando o veículo acelera ou sendo empurrado para frente quando o veículo freia; e sente-se empurrado para baixo quando o veículo sobe uma montanha ou se sente subir quando o carro passa por uma baixada ou começa a descer uma montanha. Baseado somente nestas informações, o passageiro sabe como o veículo está se movendo relativamente a ele mesmo, isto é, se ele está indo pra frente, pra trás, virando a direita, a esquerda, subindo ou descendo. Ou seja, utilizando-se somente as informações de aceleração do veículo. Ainda, o mesmo passageiro com os olhos vendados dentro de um carro pode sentir o carro virando a direita ou a esquerda, subindo ou descendo quando o carro está subindo uma montanha. E baseado nestas informações, este passageiro sabe pra qual direção o carro está virando, mas não quão rápido ou devagar ele está se movendo. Juntando as informações de velocidade e aceleração o passageiro poderá, se conhecer o ponto de partida do veículo, estimar a sua posição final. Esta é a proposta do método de navegação inercial.

Acelerômetros medem a aceleração linear do sistema em relação ao eixo inercial. Integrando-se duas vezes as acelerações é possível obter o deslocamento linear do veículo (equação (5)). Porém, para completar a integração é necessário o conhecimento da posição e velocidade iniciais do veículo. De forma que os movimentos do veículo em relação a outro objeto (outra referência) não são fornecidos pelo acelerômetro.

$$s(t) = \iint a(t) dt^2 \quad (5)$$

Giroscópios medem a velocidade angular do sistema em relação ao eixo inercial. Usando a orientação original do sistema nos eixos inerciais como condições

iniciais e integrando a velocidade angular no tempo, a orientação atual do sistema é conhecida (veja a equação (6)).

$$\varphi(t) = \int \omega(t) dt \quad (6)$$

Para alimentar as condições iniciais do sistema, este é inicializado com sua posição e velocidade a partir de outra fonte (um operador humano, um receptor GPS, última posição final etc), e em seguida ele atualiza sua posição e velocidade integrando as informações recebidas dos sensores de movimento. A vantagem de um sistema de navegação inercial é permitir o conhecimento da posição, velocidade, aceleração e atitude do sistema móvel em relação ao ambiente e a ele mesmo (GREWAL, WEILL e ANDREWS, 2007).

Para mais informações sobre os sistemas inerciais consulte o Apêndice B e para mais informações sobre o cálculo envolvido nos sistemas de navegação inerciais consulte o livro de Esmat Bekir com o título “Introduction to Modern Navigation Systems” (BEKIR, 2007).

3.2.4 Sinais de Referência

Os sistemas de guiagem por sinais de referência ativos ou “Active Beacons”, que pode ser traduzido (literalmente) por “Faróis ativos”, com referência aos faróis que auxiliam a navegação marítima são os sistemas auxiliares mais comuns em navios ou aviões. Estes sinais de referência podem proporcionar, com processamento mínimo, informações muito precisas sobre o posicionamento do veículo. Como resultado, este sistema permite altas taxas de transmissão de dados e alta confiabilidade no sistema (BORENSTEIN, EVERETT e FENG, 1996). Porém, os sistemas de navegação por sinais de referência incluem um alto custo na instalação e manutenção. Pois, além de precisar de bastante estrutura externa ao veículo, a montagem de “faróis” (*beacons*) pode representar o uso de equipamentos de precisão para ajustes do sistema. Segundo Kleeman (1992) citado por Borenstein, Everett e Feng (1996, p. 151):

“Embora faróis especiais sejam incompatíveis com noções de robôs em completa autonomia em um ambiente não estruturado, eles oferecem vantagens como precisão, simplicidade e velocidade – fatores que interessam em aplicações industriais e de escritório, onde os ambientes podem ser parcialmente estruturados.”

Existem dois tipos diferentes de sistemas por sinais de referência: Trilateração e Triangulação (BORENSTEIN, EVERETT e FENG, 1996). Ambos serão discutidos a seguir.

3.2.4.1 Trilateração

Trilateração é a determinação da posição do veículo através de medidas das distâncias entre o veículo e as fontes de referência obtidas pelo atraso de retorno de pulsos de sincronismo. Normalmente, em sistemas que utilizam trilateração, são utilizados três ou quatro transmissores distribuídos no ambiente e um receptor instalado no veículo que quer ser localizado. Mas, este modelo pode ser alterado colocando apenas um transmissor no veículo e receptores espalhados no ambiente.

A Figura 15 mostra a ideia básica de um sistema que utiliza trilateração. Com três referências P1, P2 e P3 pode-se determinar a posição do veículo com base nos tempos que os sinais tardaram até chegar no receptor. Esses tempos são proporcionais às distâncias r_1 , r_2 e r_3 . Com P1 determina-se que o veículo está no raio r_1 . Utilizando o sinal de P2 infere-se que o veículo somente pode estar em A ou B. Mas, utilizando o sinal de P3, pode-se garantir que o veículo está na posição B.

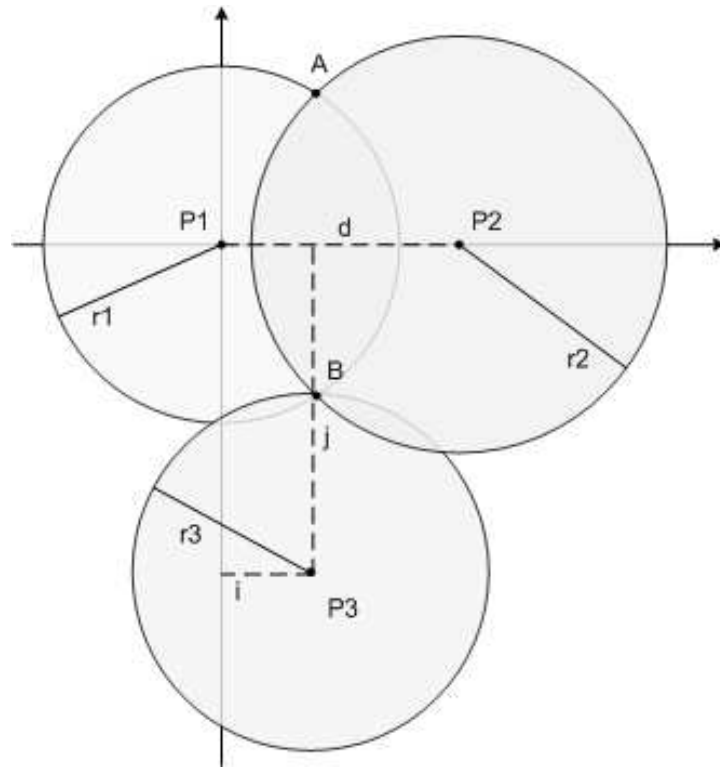


Figura 15 – Método de localização por trilateração.

Matematicamente, temos as equações das circunferências em (7) , (8) e (9).

$$r1^2 = x^2 + y^2 \quad (7)$$

$$r2^2 = (x - d)^2 + y^2 \quad (8)$$

$$r3^2 = (x - i)^2 + (y - j)^2 \quad (9)$$

Subtraindo-se a equação (8) de (7) temos:

$$r1^2 - r2^2 = x^2 + y^2 - [(x - d)^2 + y^2]$$

$$r1^2 - r2^2 = x^2 + y^2 - x^2 + 2xd - d^2 - y^2$$

$$r1^2 - r2^2 = 2xd - d^2$$

$$x = \frac{r1^2 - r2^2 + d^2}{2d} \quad (10)$$

Substituindo-se (10) em qualquer uma das três primeiras equações (7), (8) ou (9) é facilmente encontrado y . A solução é igualmente simples quando o problema envolve três dimensões. Basta utilizar-se a equação da esfera e têm-se três equações com três incógnitas. Contudo, quando se utiliza mais que três referências de sinal, é melhor calcular utilizando-se matrizes. Mais informações no livro de Borenstein, Everett e Feng (1996, p. 154-157)

Para calcular a distância (r_1 , r_2 e r_3) em relação aos pontos de referência (faróis) o sistema deve utilizar uma referência de tempo que pode ser interna ao sistema ou pode ser parte da informação transmitida. Um exemplo destes pulsos pode ser visto na Figura 16. Para aplicações de robótica móvel são utilizados sensores de ultrassom ou laser.

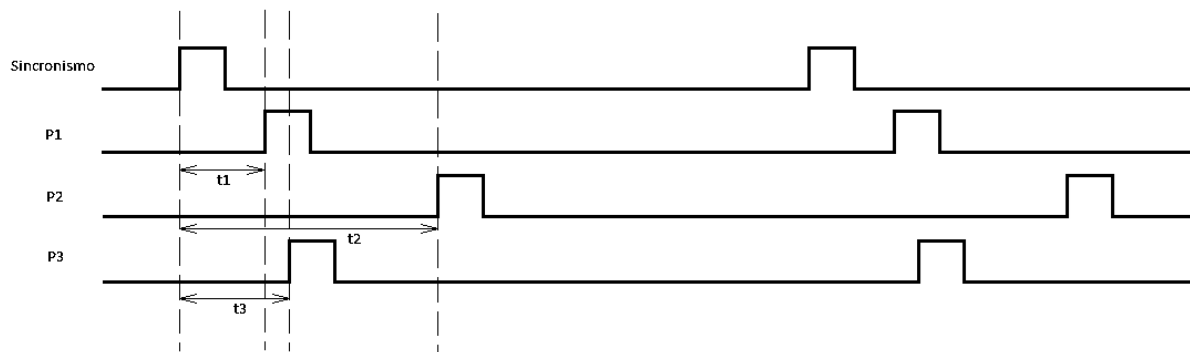


Figura 16 – Representação dos pulsos para o cálculo das distâncias no método de triangulação.

Utilizando-se ultrassom ou laser, a medida de distâncias por tempo de percurso utiliza a velocidade de propagação do som ou da onda eletromagnética utilizada como referência. A distância percorrida por uma onda sonora é dada por:

$$d = c * t \quad (11)$$

Onde, d é a distância percorrida; c é a velocidade de propagação do som no meio considerado; e t é o tempo de propagação da onda. Assim, temos as equações abaixo:

$$r1 = c * t1 \quad (12)$$

$$r2 = c * t2 \quad (13)$$

$$r^3 = c * t^3 \quad (14)$$

É importante ressaltar que a velocidade de propagação do som é aproximadamente um milhão de vezes menor que a velocidade da luz. Ou seja, para uma onda sonora percorrer uma distância de 3m ela viaja 10ms. Porém, para um raio de luz percorrer a mesma distância, teremos 10ns. Por este motivo que os sistemas que utilizam laser são mais recentes e bem mais sofisticados que os que utilizam ultrassom.

A qualidade de um sistema de trilateração depende basicamente de:

- Incertezas na determinação do tempo exato de chegada do sinal;
- Imprecisão na medida do tempo de propagação (principalmente com sistemas a laser);
- O cone de dispersão da onda transmitida (principalmente em sistemas a ultrassom);
- Interação com o ambiente (reflexões e absorções indesejadas);
- Variação da velocidade de propagação da onda;
- Velocidade de deslocamento do veículo.

Um importante exemplo de um sistema que utiliza este método é o sistema de posicionamento global (*Global Positioning System* – GPS), sua trilateração é representada na Figura 17. A Figura 17 contém as três partes principais do sistema GPS: Usuários (portando receptores), Satélites GPS (capazes de se comunicar com as estações de monitoramento e controle para sincronização do sistema) e as Estações de Monitoramento e Controle (capazes de gerenciar as atividades dos satélites no espaço).

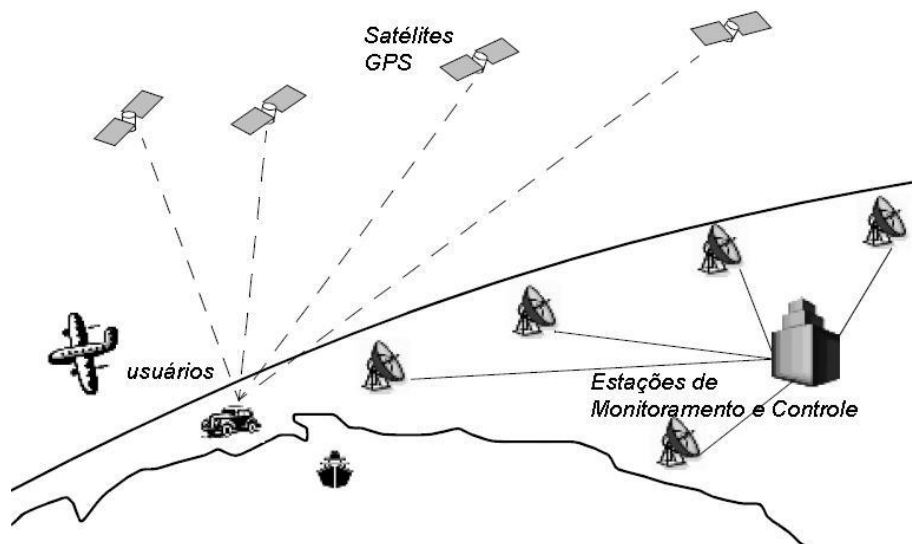


Figura 17 – Representação de um sistema de trilateração GPS.

Fonte: (SIEGWART e NOURBAKHS, 2004, p. 102)

3.2.4.2 Triangulação

Triangulação é a determinação da posição de um veículo móvel através das medidas dos ângulos entre a referência do veículo e o vetor distância entre o veículo e as fontes de referência. Neste método existem três ou mais receptores ativos ou refletores montados em locais conhecidos do ambiente e um transmissor ou transceptor rotativo instalado no veículo móvel que transmite pulsos e registra os ângulos pelos quais ele, ou os receptores, recebem os sinais emitidos em relação ao eixo longitudinal do veículo. Este sistema é representado na Figura 18. Uma representação dos gráficos dos pulsos recebidos é mostrada na Figura 19.

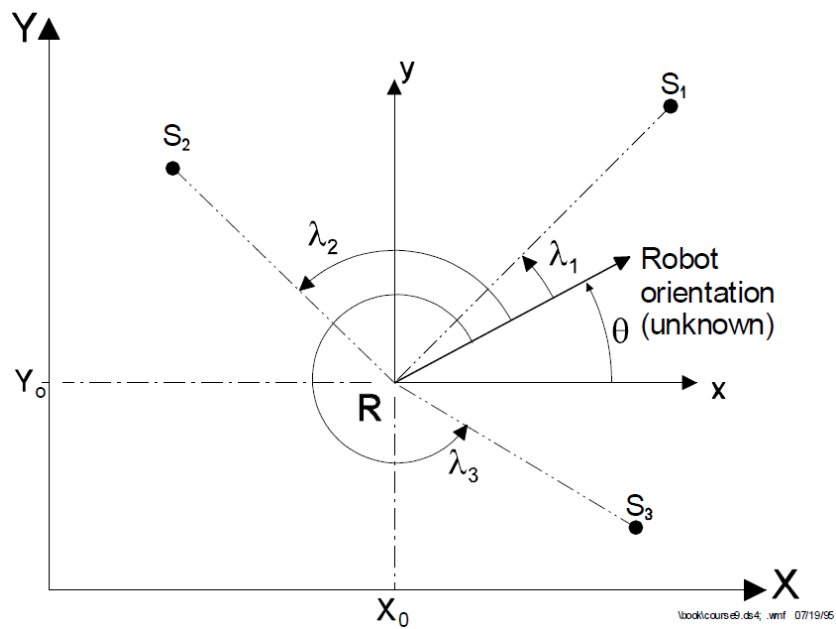


Figura 18 – Localização por triangulação. Um sensor rotativo instalado no veículo mede os ângulos λ_1 , λ_2 e λ_3 entre o eixo longitudinal do veículo e as três fontes de sinais de referência S1, S2 e S3.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996)

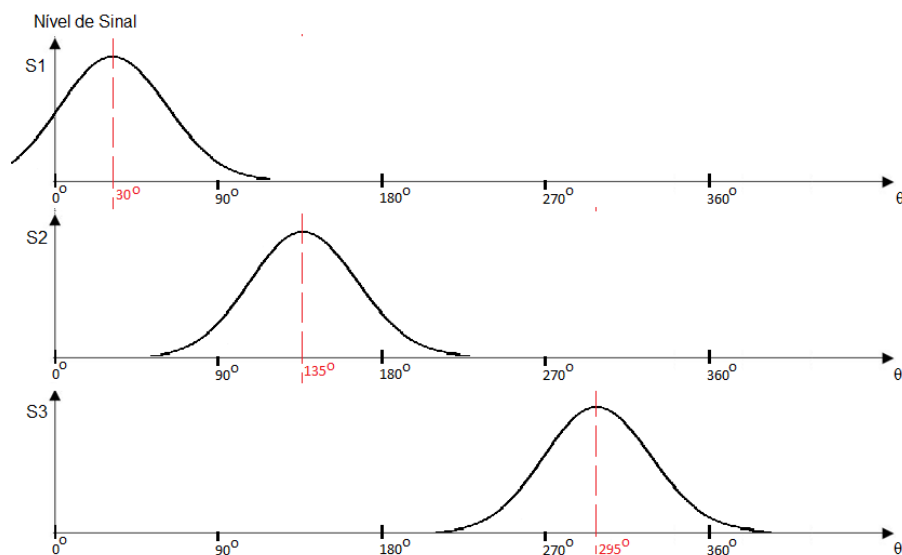


Figura 19 – Esquema representando as formas de onda recebidas pelo método de triangulação.

A Figura 20 mostra os ângulos das figuras geométricas formadas pela triangulação.

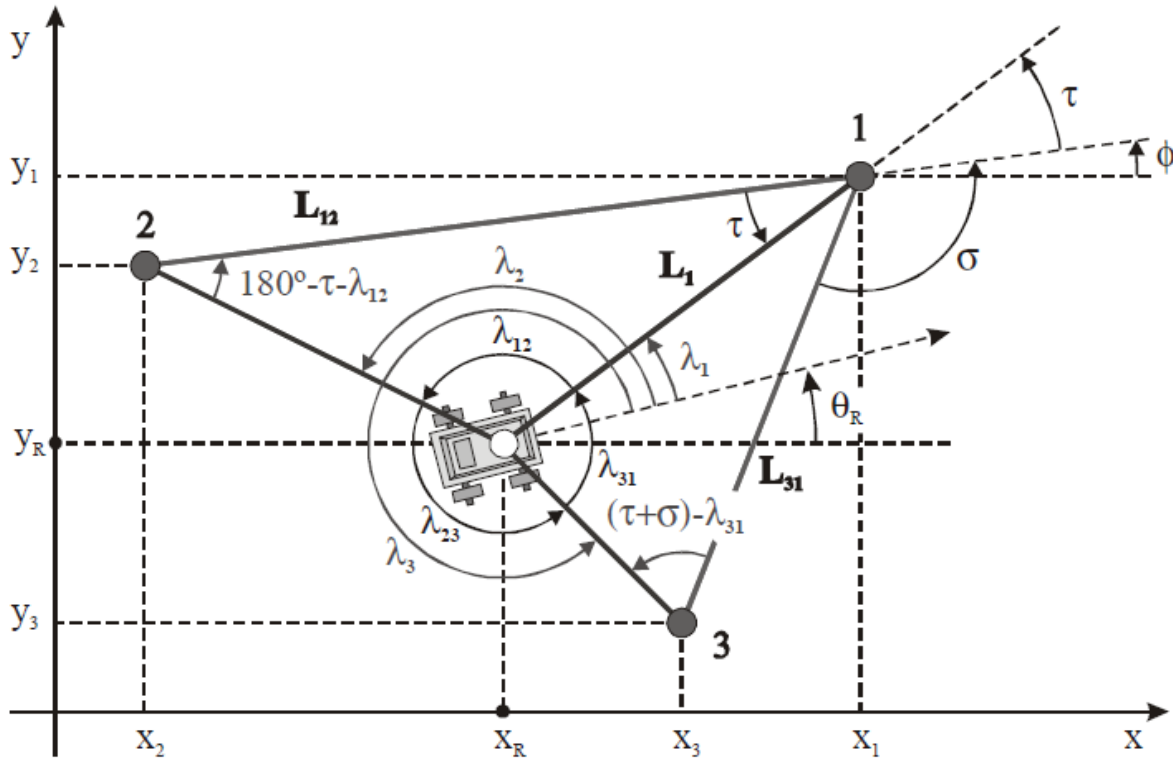


Figura 20 – Esquema mostrando os ângulos utilizados nos cálculos do método de triangulação.

Fonte: (SENA ESTEVES, CARVALHO e COUTO, 2003)

Segundo Cohen, Koss (1992) e Sena Esteves e Couto (2003), a partir destas três medidas de ângulo (λ_{12} , λ_{23} e λ_{31} da Figura 20), as coordenadas x e y e a orientação θ pode ser calculada. Assim, sistemas de navegação deste tipo podem ser feitos com baixo custo.

Um problema com esta configuração é que os transmissores devem ser extremamente potentes para garantir uma transmissão omnidirecional e cobrir grandes distâncias. Como fontes de sinais de alta potência são pouco práticas, as fontes de referência, normalmente, possuem um padrão de emissão cônico. Isto deixa algumas áreas do ambiente não cobertas pelo sistema.

Segundo Borenstein, Everett e Feng (1996, p. 152), existem duas configurações possíveis para um sistema que utilize a triangulação:

- Receptores fixos e transmissores rotacionais no veículo;
- Refletores fixos e transceptores rotacionais no veículo.

Cada variação possui suas vantagens e desvantagens que devem ser estudadas individualmente com a aplicação.

3.2.5 Marcas

Tudo aquilo que um robô pode reconhecer em seu sistema para utilização como referência de localização denomina-se marca. Marcas podem ser formas conhecidas (geométricas ou não) que podem conter informações adicionais (como em um código de barras) ou não. Em geral, as marcas estão em uma posição fixa conhecida (como no teto ou no chão da sala), com as quais o robô se localiza relativamente à marca. Caso as marcas estejam em locais variados, se torna necessário um controle de foco e *zoom* da câmera através de dois motores nas lentes e um algoritmo de identificação da nitidez da imagem. Este processamento adicional acrescenta atrasos neste processo de localização.

Marcas devem ser escolhidas de acordo com o ambiente para serem de fácil identificação, observando-se iluminação, contraste e linhas de “visada” do sensor de forma a serem detectadas.

A tarefa principal do uso de marcas para a localização é o reconhecimento preciso de uma marca e, baseado nas informações da memória o robô (mapas, rotas, etc), a determinação do local onde o veículo está. De forma a simplificar o problema, frequentemente são assumidos que a atual posição e orientação do veículo são conhecidas, então o robô deve procurar a marca em somente uma área restrita. Por esta razão, o uso de um sistema odométrico de alta precisão é um pré-requisito para uma detecção de marcas com sucesso.

O processo de posicionamento por marcas é mostrado no diagrama da Figura 21.

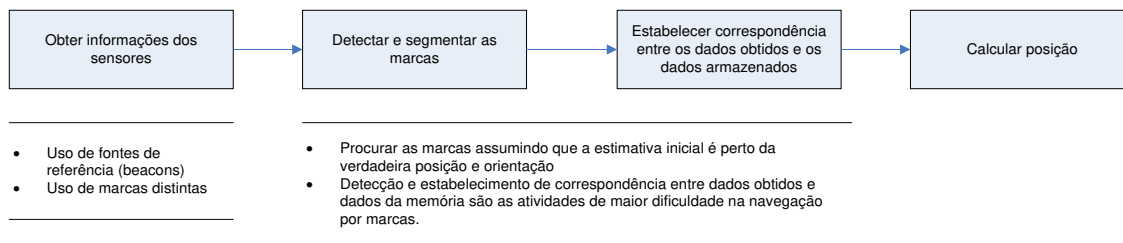


Figura 21 – Diagrama do processo de posicionamento por marcas.

Fonte: Adaptado de (BORENSTEIN, EVERETT e FENG, 1996).

Um dos problemas encontrados na navegação por marcas é a ausência da constância de cor e brilho nos sistemas de visão artificial. Porém, este problema deve ser analisado e solucionado utilizando-se filtros que compensem a variação de iluminação (ALMONFREY, VASSALLO, *et al.*, 2011).

Existem dois tipos principais de marcas em sistemas de localização robótica, a saber, marcas naturais e marcas artificiais.

- Marcas artificiais são objetos ou marcadores especialmente projetados que precisam ser colocados no ambiente com o único propósito de permitir a navegação do robô;
- Marcas naturais são quando os objetos ou formas que serão utilizadas na navegação do robô já estão no ambiente ou possuem outra função além da navegação do robô.

3.2.5.1 Marcas artificiais

A detecção de uma marca é muito mais fácil com marcas artificiais que com marcas naturais, pois as marcas artificiais são desenhadas para um contraste ótimo, e ainda, o tamanho e o formato exato da marca artificial são conhecidos previamente.

Pesquisadores do mundo todo têm usado diferentes tipos de marcas, geometrias, métodos/técnicas associadas para estimativa de posição. Porém, muitos sistemas de posicionamento por marcas artificiais são baseados em visão computacional.

Quando utilizado um sistema de visão computacional, as imagens obtidas estão em perspectiva. Assim, a escolha do tamanho e formato das marcas artificiais, normalmente, é um processo a que se deve dar grande importância.

A precisão alcançada no posicionamento por marcas artificiais depende da precisão com que os parâmetros geométricos das imagens das marcas são extraídos do plano da imagem, que depende da posição relativa e do ângulo entre o robô e a marca.

Em geral, a precisão diminui com o aumento da distância relativa. Normalmente existe uma faixa de ângulos relativos em que uma boa precisão pode ser alcançada. Caso o veículo saia desta faixa a precisão cai significativamente.

Existe também um tipo de marca, que são usadas porque não necessitam de sensores de visão para serem reconhecidas. Para estas marcas são utilizados sensores específicos de menor custo. Os mais frequentemente usados são os refletores de códigos de barra para leitores laser.

Ainda, outra funcionalidade se torna possível quando a marca é inserida nos veículos robóticos móveis e a câmera é colocada externamente no ambiente; é a chamada visão global que se refere ao uso de câmeras posicionadas em localizações fixas no ambiente de forma a aumentar a sensibilidade local a bordo de cada veículo. A Figura 22 mostra um esquema de uma sala com visão global. No caso mostrado na figura, foi utilizado um algoritmo probabilístico chamado de “Monte Carlo Localization” – MCL, o algoritmo de localização de montecarlo.

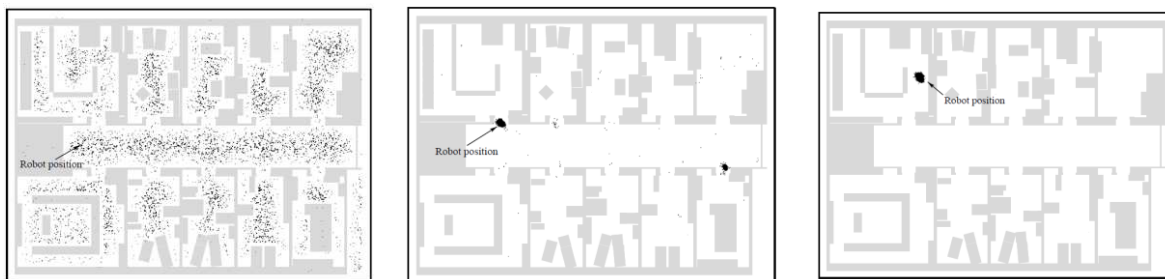


Figura 22 – Localização com o uso de visão global aplicado ao algoritmo de localização de Monte Carlo.

Fonte: (THRUN, 2000).

A Figura 23 mostra um diagrama de processamento das funções de controle de um veículo usando visão global.

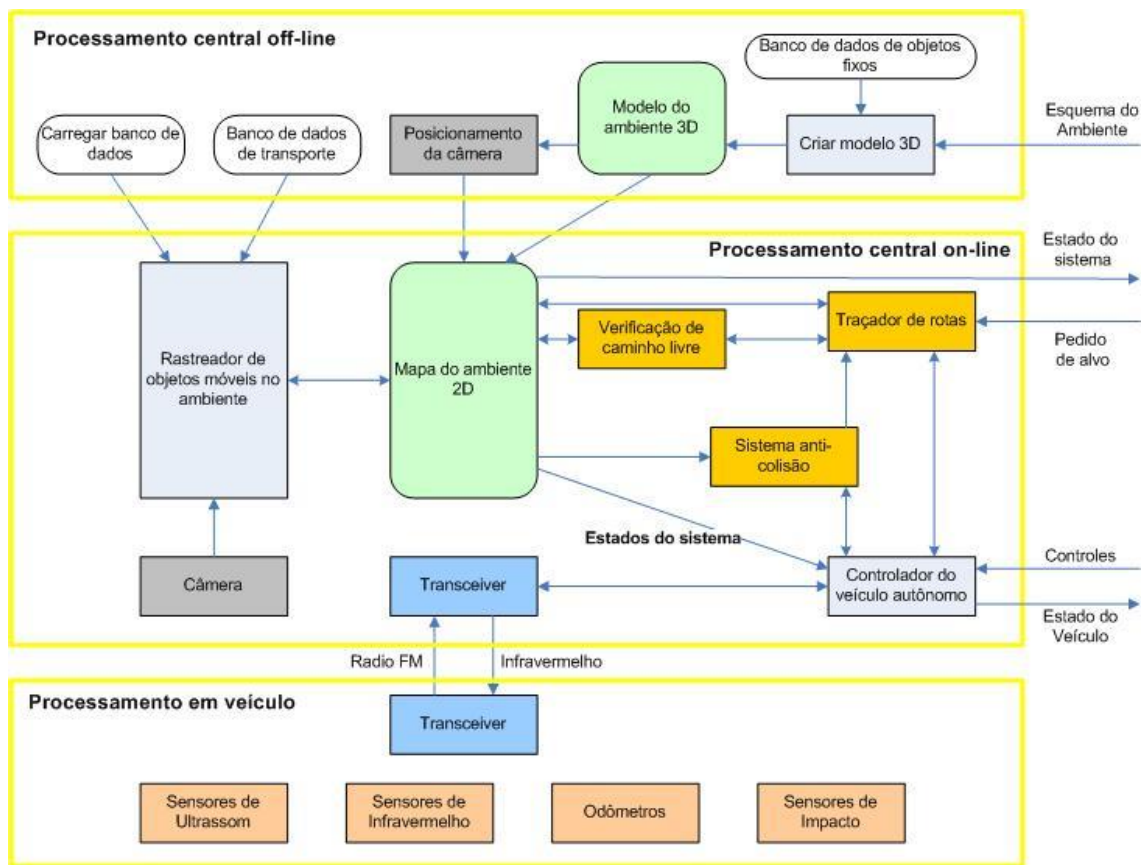


Figura 23 – Diagrama em blocos das funções processadas em um veículo autônomo usando visão global.

Fonte: Adaptado de (BORENSTEIN, EVERETT e FENG, 1996).

Em métodos de visão global, pontos característicos formando um padrão (ou marca) no robô móvel são identificados e localizados numa única visualização. Um método probabilístico é usado para selecionar o casamento mais provável de acordo com características geométricas destes padrões (ou marcas). A partir deste procedimento de procura reduzido, uma cadeia de previsão-verificação é aplicada para identificar e para localizar as marcas do sistema. Uma vantagem deste método é que ele permite ao operador monitorar a operação ao mesmo tempo.

3.2.5.2 Marcas naturais

O principal problema da navegação por marcas naturais é detectar e verificar o casamento das características dos dados obtidos nos sensores com os dados armazenados. O sensor que podemos escolher para esta tarefa é a visão computacional.

A maioria das marcas naturais, que são utilizadas em visão computacional, são longas esquinas, ou bordas verticais, tais como portas e junções de paredes, ou ainda luzes no teto.

Sensores de distância podem ser usados para navegação por marcas naturais, basta procurar grandes variações de distâncias, como uma esquina ou uma borda, ou longas paredes retas. Desta forma, um sistema de navegação por marcas naturais pode ser implementado sem a necessidade de uso de sensores de vídeo. Porém, o ambiente deve ser bastante simples para que o método seja eficiente.

A seleção de características das marcas é importante, pois irá determinar a complexidade na descrição, na detecção, e na comparação da característica. Quando é feita corretamente, esta seleção também irá reduzir a probabilidade de ambiguidade e aumentar a precisão no posicionamento.

Segundo Borenstein, Everett e Feng (1996), um sistema de posicionamento por marcas naturais geralmente tem os seguintes componentes básicos:

- Um sensor (normalmente visão computacional) para detecção de marcas, verificando o contraste com o ambiente;
- Um método de comparação das características observadas com um mapa de marcas conhecidas;
- Um método de localização computacional e de erros de localização a partir das comparações.

3.2.5.3 Resumo

Segundo Borenstein, Everett e Feng (1996, p. 183), em resumo, as características de uma navegação baseada em marcas são as seguintes:

- Marcas naturais oferecem flexibilidade e não requerem modificações no ambiente;

- Marcas artificiais são mais fáceis de implementar e podem ter informações adicionais codificadas como padrões ou formas;
- A distância máxima aceitável entre o robô e a marca é substancialmente menor que em sistemas de fonte de referência ativa, tornando necessária uma redução na área de trabalho do robô ou um aumento no número de marcas;
- A precisão no posicionamento depende da distância e do ângulo entre o robô e a marca. A navegação por marcas é bastante imprecisa quando o robô está distante das marcas. Porém, um alto grau de precisão é obtido quando o robô está perto de uma marca.
- Quando se utiliza visão computacional, condições ambientes, como iluminação, podem ser problemáticas. Pois, as marcas podem não ser reconhecidas ou outros objetos do ambiente (com formas semelhantes) podem ser erroneamente reconhecidos;
- As marcas devem estar próximas a área de trabalho do robô;
- Um sistema de navegação baseado em marcas necessita de outro sistema de localização inicial tal que o robô saiba onde procurar por marcas. Se a posição inicial não é conhecida, o robô iniciará um processo de busca que pode levar muito tempo;
- Um banco de dados de marcas e suas localizações no ambiente deve ser criado.

3.2.6 Vídeo

Quando marcas artificiais ou mapas de um ambiente não estão disponíveis, a seleção de marcas naturais e a construção de mapas deve ser parte do método de localização. Assim, um sistema de visão ou múltiplos sensores de visão podem capturar características de imagens ou regiões que se assemelham com marcas, traçando mapas. Por outro lado, marcas e modelos de objetos devem proporcionar informação espacial suficiente para que sejam capturados.

A maioria dos trabalhos desenvolvidos sugere metodologias que reportam as características da imagem detectada para características de objetos em um ambiente. A Figura 24 mostra um esquema em perspectiva onde é detectada uma marca natural do ambiente, combinando o método de navegação por marcas naturais e o sistema de vídeo. Técnicas baseadas em visão podem ser combinadas com os métodos de estimativa (*dead-reckoning*), sensores inerciais, sensores ultrassônicos e sensores a laser, através de técnicas de fusão sensorial.

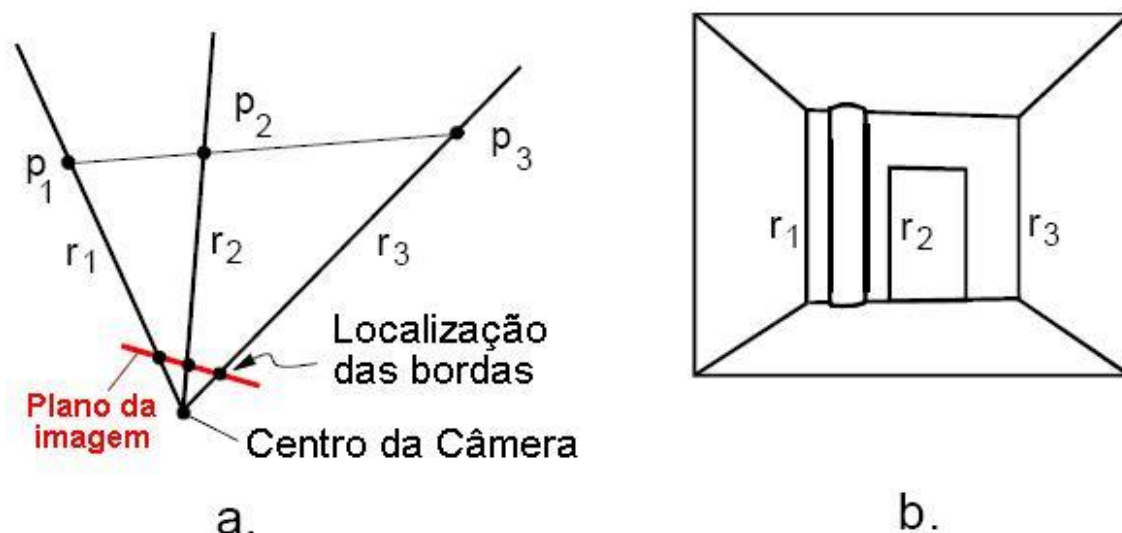


Figura 24 – Localização através de marcas naturais utilizando vídeo.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996)

Similarmente aos métodos baseados em marcas e mapas, o posicionamento por vídeo permanece no estágio de pesquisa de algoritmos mais eficientes. Isto é diretamente relacionado à maioria dos métodos de visão computacional, especialmente reconhecimento de objetos, que envolve identificação de classe de objeto e estimativa de orientação a partir de um objeto identificado. Um exemplo de robô móvel utilizando-se a visão computacional é mostrado na Figura 25.



Figura 25 – Robô Experimental SAVAR –
Robô que utiliza somente uma
câmera para se localizar.

Fonte: (MACHADO, ROSA e DE
ALMEIDA, 2011)

Grande parte do problema deste processo de reconhecimento se deve a restrição devido a que ao utilizar-se somente uma câmera dispõe-se apenas de uma imagem de duas dimensões. Porém, é possível estimar a terceira dimensão utilizando-se algum outro recurso ou sensor. Este problema relacionado à perspectiva do objeto é mostrado na Figura 26.

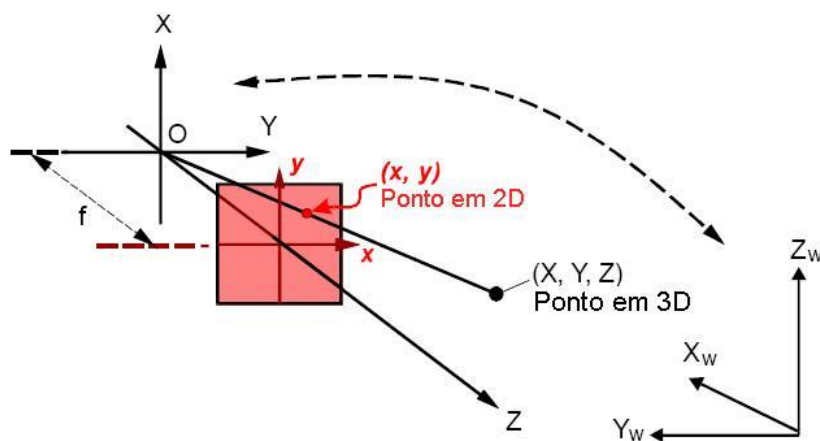
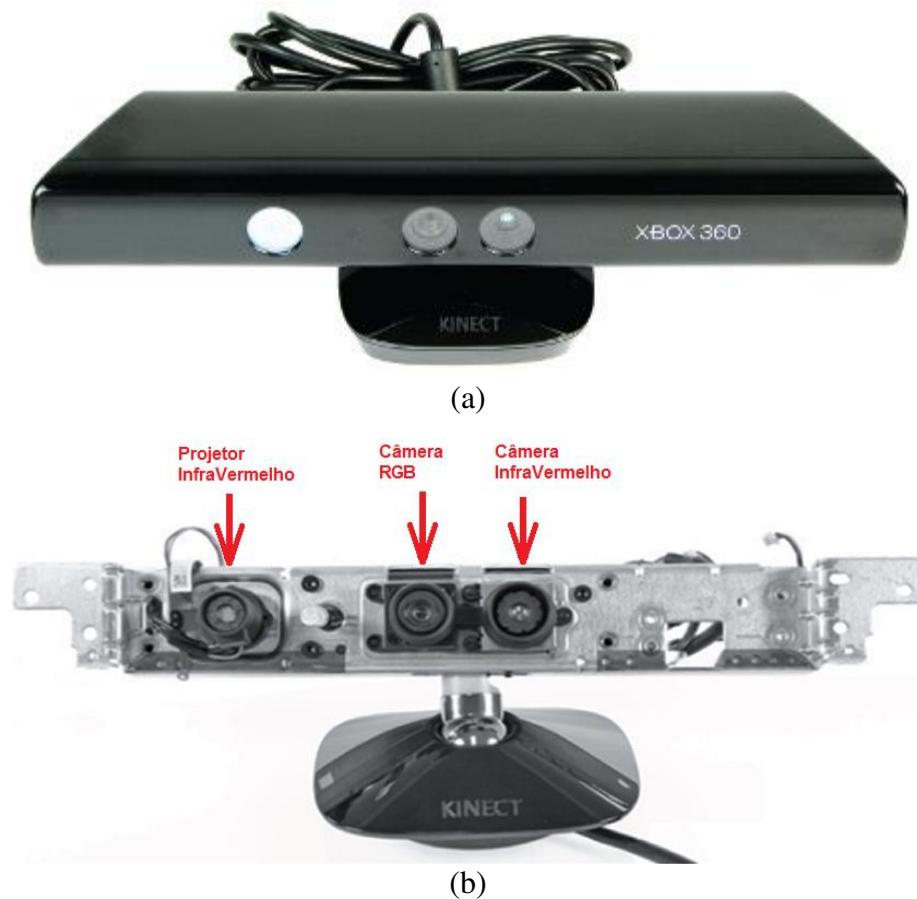


Figura 26 – Modelo de perspectiva de uma câmera.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996)

Atualmente existem pesquisas em muitas áreas de visão computacional e processamento de imagens que podem ser aplicadas ao posicionamento por vídeo. Além disso, para reconhecimento de objetos, importantes áreas de pesquisa merecem ser consideradas, estas áreas incluem: estruturas estéreo, movimento e contorno de movimento, modelamento de sensor de visão, e processamento de imagem em baixo nível.

Enfim, existem muitas técnicas de visão computacional que são potencialmente úteis, mas não foram especificamente aplicadas a problemas de posicionamento de robôs móveis e testadas sobre condições reais. O campo de localização robótica visual obteve grande crescimento com o lançamento do Microsoft Kinect, mostrado na Figura 27, este dispositivo possibilitou diversos avanços no campo da localização visual.



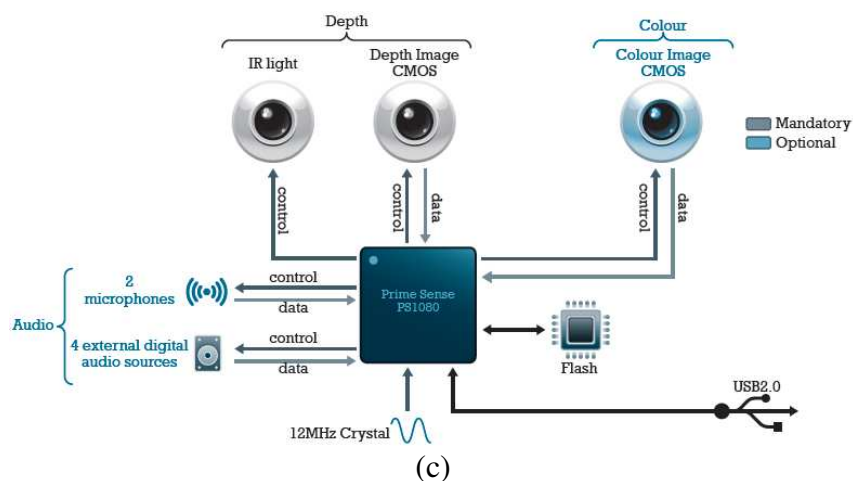


Figura 27 – Microsoft Kinect. (a) produto. b) componentes internos de vídeo. (c) diagrama esquemático dos componentes internos do Kinect.

Fonte: (a) (The Teardown, 2011); (b) (ZHANG, 2012); (c) (The Teardown, 2011).

A Figura 28 mostra simplificada como o Kinect obtém a posição do operador a partir das imagens captadas. Em (a) temos a imagem adquirida pela câmera infravermelha. Em (b) temos o reconhecimento de partes do corpo por mapeamento *pixel* por *pixel*. Em (c) é mostrado posicionamento hipotético das junções do corpo baseado na continuidade e no conhecimento prévio do corpo humano. Em (d) é mostrado o esqueleto humano da figura obtida.

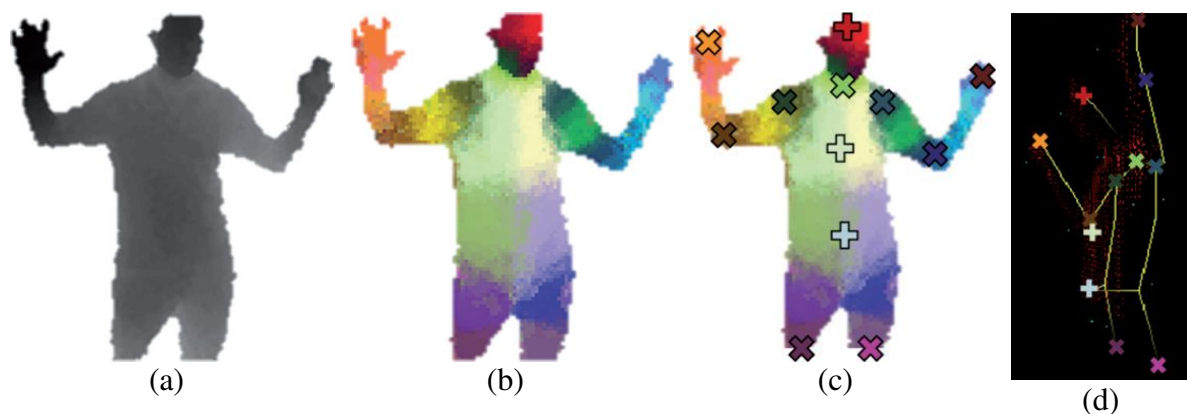


Figura 28 – Sequencia de rastreamento esquelético do Kinect.

Fonte: (ZHANG, 2012).

A Figura 29 mostra uma aplicação desenvolvida pela Parallax de forma a utilizar o Microsoft Robotics Studio para programar utilizando o Kinect. Esta plataforma se propõe a tornar a utilização do Kinect popular e não apenas ligada ao videogame Microsoft XBOX 360 para o qual o Kinect foi criado.



Figura 29 – Plataforma de desenvolvimento Eddie do consórcio Microsoft-Parallax. Conjunto composto de uma base robótica com motores e sensores, um *laptop* e um kinect.

Fonte: (PARALLAX INC., 2013)

3.2.7 Mapas

Posicionamento baseado em mapas é uma técnica em que o robô usa seus sensores para criar um mapa local de seu ambiente. Este mapa local é então comparado a

um mapa global previamente armazenado na memória. Se, ao comparar-se os ambientes local e global, um casamento é encontrado, então o robô pode registrar sua atual posição e orientação no ambiente. Um exemplo deste tipo de navegação é utilizado por Aguiar e Vassalo (AGUIAR e VASSALLO, 2011) e é mostrado na Figura 30. Em (a) é mostrado o Robô Observador – PIONEER 2-DX que possui uma câmera com uma lente grande angular que obteve a imagem de fundo mostrada em (b). Em (c) é mostrado o Robô navegador – PIONEER 3-AT que utiliza a ilustração (d) como parte do processo de escolha do nó para o caminho. Em (e) é mostrada uma comparação entre os caminhos planejados e percorridos pelo robô navegador PIONEER3-AT.

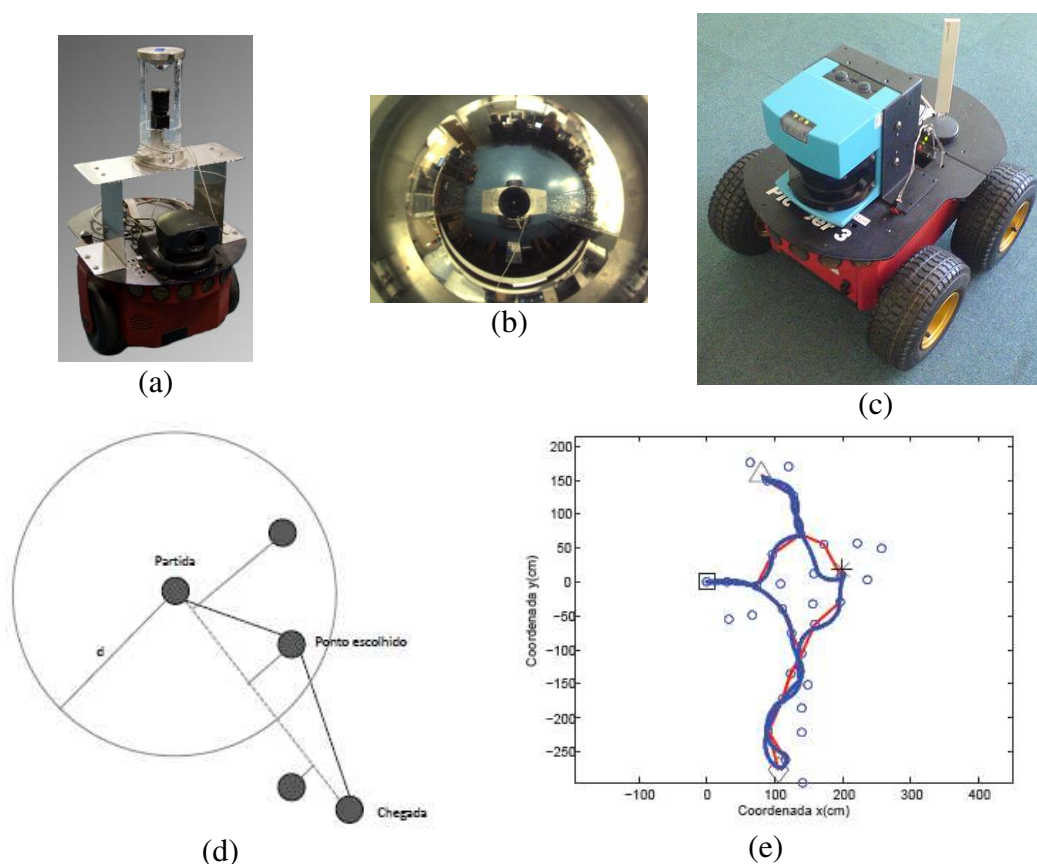


Figura 30 – Projeto utilizando navegação por mapas.

Fonte: (AGUIAR e VASSALLO, 2011).

O mapa pré-armazenado pode ser um modelo CAD do ambiente, ou pode ser construído previamente por dados de sensores. O método utilizado na construção do mapa indica o nível de detalhamento do mapa e consequentemente do posicionamento do veículo móvel.

A Figura 31 mostra um diagrama de blocos do processamento envolvido em um sistema de navegação por mapas.

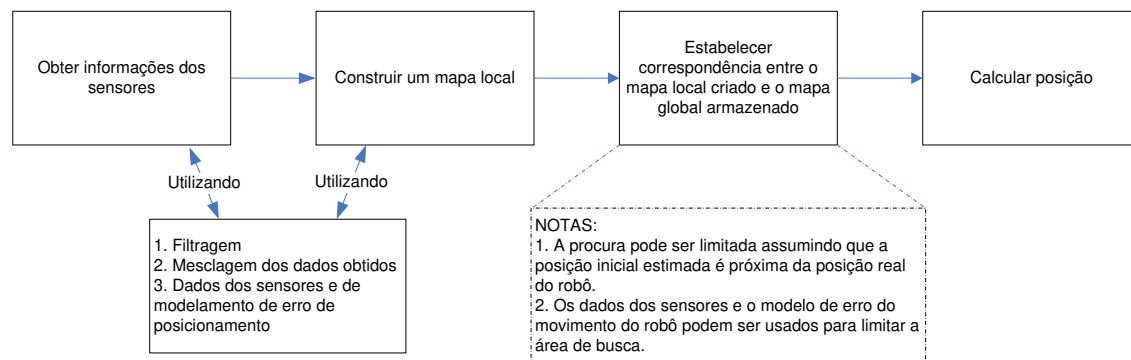


Figura 31 – Diagrama em blocos do processamento de um sistema de navegação por mapas.

Fonte: Adaptado de (BORENSTEIN, EVERETT e FENG, 1996).

Segundo Borenstein, Everett e Feng (1996), as principais vantagens de um posicionamento por mapas são as seguintes:

- Este método, utilizado em ambientes tipicamente interiores (indoor), usa a estrutura naturalmente existente destes ambientes para obter a informação de posição. Isto é feito sem modificar previamente o ambiente.
- Posicionamento por mapas pode ser usado para gerar mapas atualizados de um ambiente (a Figura 32 mostra um algoritmo básico de geração de mapas). Mapas de ambientes são importantes para outras tarefas de robôs móveis, como planejamento de rotas ou técnicas de desvios de obstáculos no caminho.
- Posicionamento por mapas permite ao robô aprender um novo ambiente e aumentar a sua precisão de posicionamento através da exploração.

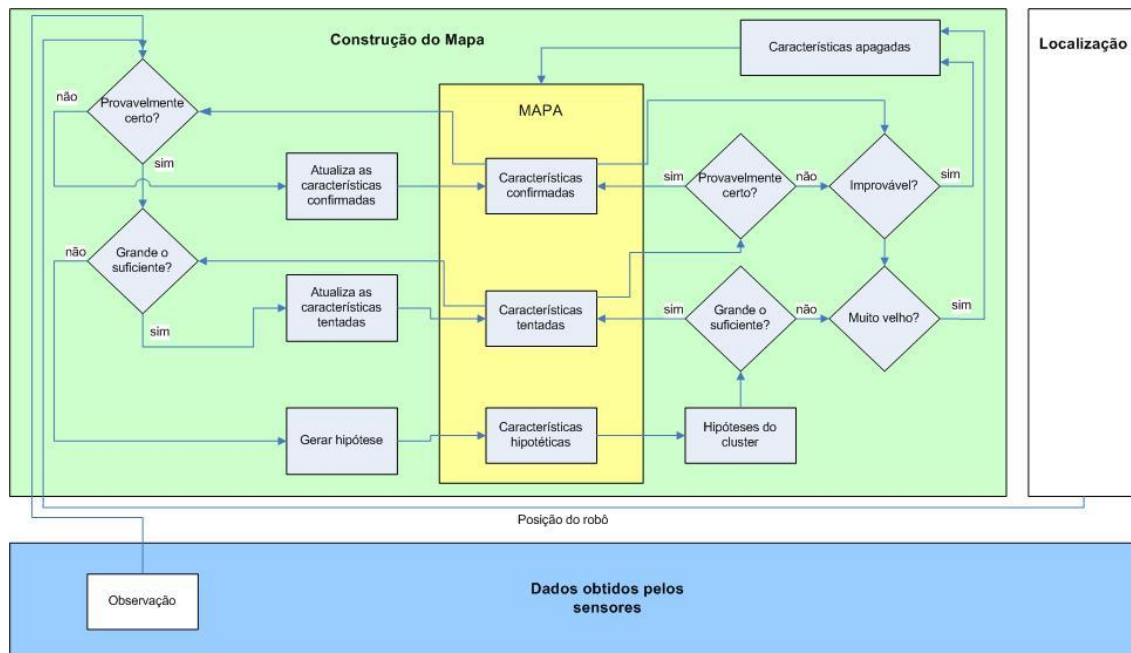


Figura 32 – Algoritmo básico para construção de mapas.

Fonte: Adaptado de (BORENSTEIN, EVERETT e FENG, 1996).

Segundo Borenstein, Everett e Feng (BORENSTEIN, EVERETT e FENG, 1996), as desvantagens do posicionamento por mapas são requisitos específicos para uma boa navegação. Por exemplo, para o posicionamento por mapas é necessário que:

- Existam objetos fixos suficientes, com características facilmente distinguíveis que possam ser usados para comparação e casamento.
- O mapa gerado pelos sensores seja suficientemente preciso (dependendo da tarefa) para ser útil;
- Uma quantidade significativa de sensores e processamento deve estar disponível.

Existem dois tipos de pontos de partidas fundamentalmente diferentes no processo de posicionamento por mapas. Por um mapa pré-existente, ou o robô deve construir seu próprio mapa do ambiente.

Posicionamento por mapas ainda está no estágio de pesquisa. Atualmente, esta técnica é limitada a ajustes de laboratório e bons resultados estão sendo obtidos somente em

ambientes bem estruturados (relativamente simples). É difícil julgar como é o desempenho de um robô de laboratório colocado em uma aplicação do mundo real.

3.2.7.1 Resumo

Segundo Borenstein, Everett e Feng (1996, p. 206), as informações relevantes de sistemas de posicionamento baseado em mapas são resumidas a seguir:

- Necessitam de uma capacidade de processamento significativa e grande capacidade sensorial;
- Necessitam processamento exaustivo, dependendo dos algoritmos e da resolução utilizada;
- Necessitam de uma estimativa de posição inicial, a partir de odometria, de forma a limitar a pesquisa inicial por características em uma pequena área;

Ainda, existem pontos críticos neste sistema que necessitam ser desenvolvidos:

- Seleção de sensores e associação de sensores para aplicações e ambientes específicos;
- Algoritmos precisos e confiáveis para casamento de mapas locais e mapas armazenados;
- Bons modelos de erro de sensores e do movimento do robô;
- Bons algoritmos para integração de um mapa local em um mapa global.

3.2.8 Tabela comparativa

Com as informações discutidas nas seções anteriores foi compilada a Tabela 5 onde são apresentadas as informações mais importantes referentes aos métodos de guiagem. Nesta tabela foi utilizada como abreviação a letra **B** para baixo, **M** para médio, **A** para alto e **MA** para muito alto. Ainda, foi também utilizado **Abs** como abreviação para absoluto e **Rel** para relativo.

Tabela 5 – Análise comparativa dos métodos de navegação estudados.

Método de navegação	Sensores Utilizados	Custo (ref. 2012)*	Complexidade de processamento	Precisão	Absoluto/Relativo	Dimensões	Exige mudanças no ambiente	Comentários
Tátil	<i>Bumpers</i> (sensores de contato) Sensores de proximidade por ultrassom ou <i>laser</i>	B	B	A	Abs	1	Não	Apesar de sua precisão alta, este tipo de sensor exige o contato físico ou proximidade como forma de ativação.
Odométrica	<i>Encoders</i>	B	B	B	Rel	1/ 2	Não	Necessita de métodos preditivos de correção de erro. Necessita de referências absolutas com uma certa frequência.
Inercial	Acelerômetros/Giroscópio	M ~ MA	M	M A	Rel	3	Não	Com o surgimento da tecnologia MEMS (Micro-Electro-Mechanical Machine Systems – Sistemas Eletromecânicos miniaturizados em escala micrométrica em silício) o custo deste sistema de navegação caiu significativamente.
Trilateração	Ultrassom <i>beacons</i> <i>Laser Range Finder+beacons</i>	A	B	A	Abs	2/ 3	Sim/ Não	Alto custo de instalação e manutenção. Necessita operar em ambientes conhecidos (indoor).
Triangulação	<i>Laser Range Finder+beacons</i>	A	B	A	Abs	2	Sim	Alto custo de instalação e manutenção. Necessita operar em ambientes conhecidos (indoor).

Método de navegação	Sensores Utilizados	Custo (ref. 2012)*	Complexidade de processamento	Precisão	Absoluto/Relativo	Dimensões	Exige mudanças no ambiente	Comentários
Marcas Naturais	Câmeras CCD	M	A	M	Abs	2	Não	Necessita de um método odométrico para indicar as possíveis áreas onde se buscam as marcas. Precisão diminui com a distância entre as marcas.
Marcas Artificiais	Laser Range Finder/Câmeras CCD	M	A	A	Abs	2	Sim	Necessita de um método odométrico para indicar as possíveis áreas onde se buscam as marcas. Precisão diminui com a distância entre as marcas.
Marcas Artificiais (Visão Global)	Câmeras CCD	MA	MA	A	Abs	3	Sim	Funciona como uma mistura da navegação por marcas artificiais com a navegação por vídeo.
Mapas	Laser Range Finder/Câmeras CCD	A	MA	M	Abs	2/3	Não	Necessita de um método odométrico para indicar as possíveis áreas onde se buscam as marcas. Precisão diminui com a distância das marcas.
Vídeo	Câmeras CCD	MA	MA	A	Abs	2/3	Não	Necessita de um método odométrico para indicar as possíveis áreas onde se buscam as marcas. Precisão diminui com a distância das marcas.

*Custos: Baixo= <R\$10,00; Medio = R\$10,00 ~ R\$100,00; Medio-Alto = R\$100,00 ~ R\$500,00; Alto = R\$500,00 ~ R\$1000,00; e Muito Alto = >R\$1000,00

3.3 Ação Motora

O terceiro principal componente da autonomia como atividade mental é a ação motora. Em robôs esta ação motora deve ser executada por um bloco específico e é extensamente estudada pela mecatrônica como ciência. Mais informações sobre tipos de plataformas mecânicas e seus cálculos para acionamentos podem ser encontrados nos livros de Siegwart, Nourbakhsh e Scaramuzza (2011) *“Introduction to Autonomous Mobile Robots”* e Sandin (2003) *“Robot Mechanisms and Mechanical Devices Illustrated”*.

3.4 Comentários Gerais

Com isto, foram abordadas algumas das atividades mentais representadas artificialmente com técnicas aplicáveis a robôs móveis autônomos. Alguns dos métodos são mais eficientes que outros. Mas, quando operam juntos, tornam-se ainda mais eficientes.

O próximo capítulo trata de como interconectar as técnicas e métodos apresentados anteriormente para a orientação do dispositivo robótico móvel de forma a obter um sistema modular, escalável, incremental e preparado para ter autonomia.

4 Interconexões

De forma a interconectar as atividades de Sensação, Percepção, Orientação e Ação Motora entre si e permitir outras atividades mentais (como a consciência e a atenção) de se manifestarem, faz-se necessário o estudo das interconexões. Estas interconexões fazem parte da arquitetura do sistema.

A arquitetura do sistema pode ser subdividida em topologia, barramento elétrico, barramento de alimentação e sistema operacional. Estas partes serão estudadas a seguir quanto as suas possibilidades.

4.1 Topologia

Um sistema robótico móvel geral, que integra diversos sensores, deve ser analisado e ter uma topologia condizente com a flexibilidade da aplicação. Se o sistema deve ser flexível ao ponto de permitir que sensores e atuadores sejam introduzidos e/ou retirados do sistema (conforme foi concluído na análise comparativa) durante o uso do mesmo, a topologia mais indicada é um tipo de topologia modular, onde cada módulo representa uma ou mais funções do sistema.

Segundo Siegwart e Nourbakhsh (2004), princípios básicos de engenharia de software adotam modularidade de software, e as mesmas motivações gerais se aplicam igualmente a aplicações robóticas. Mas, modularidade é mesmo mais importante em robótica móvel porque no decorrer de um projeto o hardware de um robô pode mudar dramaticamente, um desafio que a maioria dos computadores tradicionais não encontram.

Da mesma forma que modularidade é importante para o desenvolvimento de software, a modularidade pode ser igualmente importante para o desenvolvimento de hardware. Os sistemas modulares mais conhecidos são os sistemas automotivos, onde as arquiteturas elétricas são analisadas de forma a se obter diversos veículos com diferentes

configurações, porém, utilizando-se de um mesmo barramento e às vezes até os mesmos módulos.

Um sistema modular pode ter diversas arquiteturas, e cada arquitetura possui vantagens e desvantagens em relação às outras. Foram estudadas as arquiteturas “centralizada”, “cascateada”, “barramento serial sem controlador”, “barramento serial com controlador”, “distribuída”, e “distribuída com cascadeamento” (BECKE, BORGERT, *et al.*, 2000; PATRICK, 2002; HOLT, 2005; KENNAN e MCFADDEN, 1999). Estas topologias são mostradas na Figura 33.

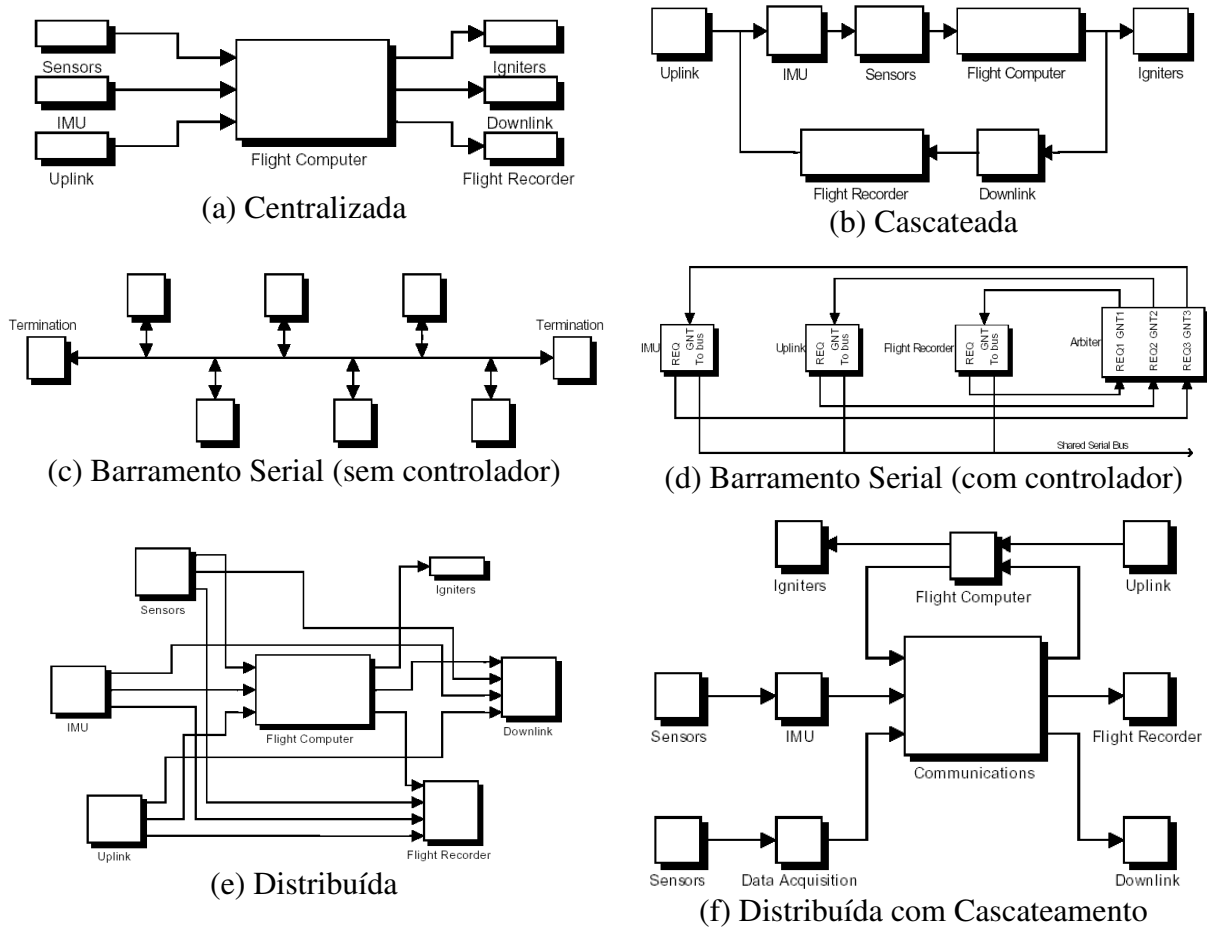


Figura 33 – Topologias de sistemas modulares.

Fonte: (KENNAN e MCFADDEN, 1999).

Tabela 6 – Prós e contras das topologias de sistema apresentadas.

Topologia	Prós	Contras
Centralizada	• Simples: fácil de implementar; • Consolidada: todos os recursos são parte do computador primário, assim nenhum protocolo de comunicação complexo tem que ser desenvolvido; • Leve: quantidade mínima de componentes mantém o peso baixo; • Pequeno: este sistema minimiza o espaço ocupado pela eletrônica.	• Não Escalonável: adições de módulos futuros irão demandar bastante trabalho devido ao limitado número de pinos; • Muito centralizado: uma falha em sistemas não críticos pode travar a unidade central; • Sobrecarregado: com todas as tarefas sendo executadas no computador central, o processador que sequencia o mapa de estados irá gastar muito tempo fazendo a fusão dos dados; • Requer um computador de bordo central específico para atender aos requisitos.
Cascadeada	• Simples: estrutura muito básica; • Facilidade de construção: utiliza somente comunicação serial integrada aos microcontroladores; • Expansível: esta arquitetura pode utilizar cascadeamento Daisy indefinidamente quando novos subsistemas forem adicionados.	• Largura de banda: cada subsistema deve ser capaz de retransmitir uma grande quantidade de dados; • Propriedade: cada subsistema deve saber que dado é necessário ao longo da cadeia. • Frágil: esta arquitetura não é tolerante a falhas. Se um subsistema se desconecta, todos são desconectados.
Barramento Serial (sem controlador)	• Enfatiza modularidade: um subsistema pode ser adicionado a arquitetura somente adicionando-se o <i>hardware</i> da interface ao subsistema; • Enfatiza confiabilidade: procedimentos de recuperação de falhas e características de prevenção podem ser implementadas; • Mesma interface de hardware para todos os módulos; • Provê um mecanismo conveniente para prioridade de pacotes de dados.	• Largura de banda limitada.
Barramento Serial (com controlador)	• Ordenado: nenhum conflito é permitido devido ao arbitrador; • Escalonável: pode ser expandido indefinidamente a medida que novos sistemas são adicionados; • Tolerante a falhas: se um subsistema se desconecta, os outros subsistemas não são afetados.	• Construção: requer adição de arbitração em hardware na interface de comunicação serial; • Largura de banda: um barramento serial é responsável por todos os dados do sistema, criando um gargalo.
Distribuída	• Largura de banda: cada barramento tem largura de banda significativamente reduzida; cada subsistema somente recebe os dados	• Aumento da complexidade de hardware e software: cada subsistema necessita de um aumento das portas de comunicação para acomodar a estrutura distribuída do

Topologia	Prós	Contras
<i>Distribuída com Cascadeamento</i>	a ele destinado;	sistema;
	• Tolerante a falhas: uma falha em qualquer subsistema somente irá afetar os subsistemas que dependem dos dados do subsistema falhado. • Minimiza a complexidade dos sistemas de comunicações com módulos que não são primariamente orientados a comunicação; • Módulos podem ser atualizados ou trocados por outros módulos que tenham as mesmas características de interface.	• Requer uma unidade central específica para atender aos requisitos de hardware quanto ao número de portas. • O processador de comunicações deve ser capaz de suportar significativa largura de banda referente ao grande número de conexões; • Requisitos de hardware com relação às portas definem a arquitetura do sistema.

Fonte: (KENNAN e MCFADDEN, 1999).

4.2 Barramento Elétrico de Comunicação

Após a definição da topologia do sistema em rede de hardware se faz necessária a decisão por um barramento elétrico que irá tornar possível tal topologia. Dependendo da topologia a ser escolhida o barramento elétrico já está decidido. Pois, alguns barramentos elétricos definem a arquitetura elétrica da rede que lhes é destinada.

Na Tabela 7 é mostrado um breve resumo sobre cada um dos barramentos estudados. Nesta tabela, Multiponto designa um barramento com muitos transmissores e receptores em linha. Este tipo de barramento normalmente utiliza transceptores como nós do barramento, mas nem todos os nós necessitam ser transceptores; *Multidrop* se refere a um transmissor e vários receptores na linha do barramento; e ponto-a-ponto se refere a somente um transmissor e um receptor na linha do barramento.

Tabela 7 – Guia de análise de barramentos elétricos.

Tipo	Modo de transmissão	Taxa de transmissão por canal	Taxa de transmissão por dispositivo	Distância	Padrão	Família de produtos
<i>Serial</i>	<i>Multiponto</i>	25 ou		1.5m	IEEE 1394-	IEEE 1394

Tipo	Modo de transmissão	Taxa de transmissão por canal	Taxa de transmissão por dispositivo	Distância	Padrão	Família de produtos
		50Mbps			1995	Backplane
		100-400Mbps 800Mbps		4.5m 100m	IEEE 1394-1995 IEEE 1394a-2000 IEEE 1394b-2002	IEEE 1394 Cable IEEE 1394 ... (PMD ext.)
		12Mbps		5m	USB 1.1	USB
		1Mbps		40m	CAN (ISO 11898)	HVD/CAN
		10Mbps a 20Mbps	5Mbps a 10Mbps			FLEXRAY
		35Mbps		10m (1200m)	TIA/EIA 485 (ISO8482)	TIA/EIA 485
		400Mbps		0.5m (~30m)	Texas Instruments Proprietary	LVDM
		500Mbps		0.5m (~30m)	TIA/EIA-899	M-LVDS
	<i>Multidrop</i>	10Mbps		10m (1200m)	TIA/EIA 422 (ITU-T V.11)	TIA/EIA-422
	<i>Ponto-a-ponto</i>	400/100Mbps	4 canais: 800/400Mbps	0.5m (~30m)	TIA/EIA-644 (LVDS) Texas Instruments Proprietary TIA/EIA-899 (M-LVDS)	LVDS/LVDM/M-LVDS
		512kbps		20m	TIA/EIA-232 (ITU-T V.28)	TIA/EIA-232
		2Gbps	4 canais: 1600/800Mbps	1m (~30m)	TIA/EIA-644 (LVDS)	LVDS
<i>Paralelo-para-serial</i>	<i>Ponto-a-ponto</i>	455Mbps	4 canais: 1.83Gbps	<15m	TIA/EIA-644 (LVDS)	LVDS SerDes/Flatlink
		1.25Gbps	1.25Gbps full duplex	<10m	IEEE P802.3z	Gigabit Ethernet
<i>Serial-</i>	<i>Ponto-a-</i>	2.5Gbps	4 canais:	<10m	IEEE	10 Gigabit

Tipo	Modo de transmissão	Taxa de transmissão por canal	Taxa de transmissão por dispositivo	Distância	Padrão	Família de produtos
<i>para-paralelo</i>	<i>ponto</i>		10Gbps full duplex		P802.3ae	Ethernet
		2.5Gbps	2.5Gbps full duplex	<10m		Serial Gigabit CMOS
		2.5Gbps	4 canais: 10.0Gbps full duplex	<10m	i.e. OIF-VSR4-03.0	SONET/SDH
<i>Paralelo</i>	<i>Multiponto</i>	35Mbps		10m (1200m)	TIA/EIA-485 (ISO8482)	TIA/EIA-485
		500Mbps		0.5m (~30m)	TIA/EIA-899	M-LVDS
		400/100Mbps		0.5m (~30m)	Texas Instruments Proprietary	LVDM
		33/66Mhz		0.2m	PCI Compact	PCI
		33/66Mhz		0.2m	PCI	PCI
		4Mhz CLK	16 canais: 64Mbps	10m	IEEE Std 1284-1994	AC1284, LVC161284, LV161284
		20Mhz CLK	32 canais: 640Mbps	0.5m	CMOS, JESD20, TTL, IEEE 1014-1987	AC, AHC, ABT
		33Mhz CLK	20 canais: 660Mbps	0.5m	LVTTL as stated in JEDS8-A, Junho 1994, IEEE 1014-1987	LVTH, ALVT
		40Mhz CLK	16 canais: 640Mbps	0.5m	VME64 Standard ANSI/VITA1-1991	ABTE
		60Mhz CLK	16 canais: 960Mbps	0.5m	IEEE Std 1194.1-1991	BTL/FB+
		60Mhz CLK	18 canais: 1.08Gbps	0.5m	JESD8-3	GTL/GTL+
		100Mhz	18 canais:	0.5m	JESD8-3	GTLP

Tipo	Modo de transmissão	Taxa de transmissão por canal	Taxa de transmissão por dispositivo	Distância	Padrão	Família de produtos
		CLK	1.8Gbps			
		200Mhz CLK	18 canais: 3.6Gbps	0.1m	EIA/JESD8-8, EIA/JESD8-9	SSTL

Fonte: (BECKE, BORGERT, *et al.*, 2000; PATRICK, 2002).

Dentre os barramentos mostrados na Tabela 7 os barramentos abaixo são os mais utilizados na indústria o que garante grande suporte de componentes e ferramentas para utilização dos mesmos.

- IEEE1394b (*Firewire*)
- *Universal Serial Bus* (USB)
- TIA/EIA-232
- TIA/EIA-422
- TIA/EIA-485
- CAN (*Controller Area Network* - ISO 11898)
- TIA/EIA-644 (LVDS)
- LVDM
- TIA/EIA-899 (M-LVDS)
- MIL-STD-1553
- *Flexray*

4.2.1 Comunicações sem fio

Com o avanço das tecnologias sem fio, está se tornando uma realidade se pensar em um sistema de controle robótico modular sem fio. Redes pessoais podem ser com fio através de cabos, neste caso estas redes se chamam PAN (*Personal Area Network*). Uma rede pessoal sem fio WPAN (*Wireless Personal Area Network*) também é possível com tecnologias de redes como IrDA, Bluetooth, UWB, Z-Wave e Zigbee. O padrão IEEE 802.15 é o 15º grupo do IEEE802 e é especializado em redes WPAN. A Tabela 8 mostra

um comparativo entre as redes WPAN: Bluetooth, Zigbee e o Bluetooth de baixa energia (Wibree).

Tabela 8 – Comparativo de redes WPAN.

	Bluetooth	Wibree	Zigbee
Norma (IEEE)	802.15.1		802.15.4
Ano de publicação	2002	*	2003 and 2006
Banda	2.4GHz	2.4GHz	2.4GHz, 868MHz, 915MHz
Antena	Compartilhada	Compartilhada	Independente
Potência	100mW	~10mW	30mW
Duração estimada da bateria	Dias - Meses	1-2 anos	6 meses - 2 anos
Alcance	10-30m	10m	10-75m
Taxa de Dados	1-3Mbps	1Mbps	25-250kbps
Número de usuários	Dezenas	Dezenas	Dezenas
Custo de Componentes (US\$) – estimativa de 2008	\$3	Bluetooth + 20¢	\$2
Topologia de redes	Ad hoc, ponto a ponto, estrela	Ad hoc, ponto a ponto, estrela	Mesh, Ad hoc, Estrela
Segurança	Encriptação de 128-bit	Encriptação de 128-bit	Encriptação de 128-bit
Tempo entre acordar e transmitir	3s	-	15ms

Fonte: (KOOKER, 2008)

Com base nestas informações foi compilada a Tabela 9 com as vantagens e desvantagens de cada uma das redes WPAN.

Tabela 9 – Vantagens e desvantagens das redes WPAN.

Rede WPAN	Vantagens	Desvantagens
Bluetooth	–Antena compartilhada –Rede Ad-Hoc –Encriptação de 128bits –<i>Frequency Hopping</i>	–Alto consumo –Baixo alcance –Alto custo
Wibree	–Antena compartilhada –Baixo Consumo –Rede Ad-Hoc –Encriptação de 128bits –<i>Frequency Hopping</i> adaptativo	–Baixo alcance –Alto custo
Zigbee	–Tri-band –Baixo Consumo –Alto alcance –Custo 50% menor que o Bluetooth –Rede Ad-Hoc –Rede mesh –Encriptação de 128bits	–Antena Independente

4.3 Barramento de alimentação

Ao passo que existe um barramento elétrico para dados é interessante utilizar os mesmos cabeamentos para alimentação do sistema. Porém, neste caso se faz necessário a criação/escolha de um módulo de alimentação onde são mantidas as baterias do sistema. Tais baterias devem ser analisadas quanto ao seu tipo de acordo com a sua utilização. Para mais informações sobre baterias consulte o Apêndice D.

Considerando um sistema modular é possível existir três tipos de distribuição de energia: Concentrado, Distribuído e Misto.

Se o sistema utilizar um único módulo para fornecer energia para os outros do sistema, a este sistema chamaremos de “sistema de alimentação concentrado”. Pois todo o armazenamento e controle de carga e descarga das baterias está concentrado em um único

módulo. Porém, a utilização de um único módulo de alimentação pode comprometer a confiabilidade do sistema. Pois, além da necessidade de fabricação (ou expansão) de um módulo toda vez que módulos forem atualizados (e necessitem de mais corrente para operar o sistema) existe ainda o problema que caso o sistema de alimentação falhe o sistema todo falha. Ainda, para dimensionar-se o módulo de alimentação ter-se-ia que utilizar de superdimensionamento para não ter-se que refazer tal módulo toda vez que se adiciona um módulo ao sistema. Desta forma, a centralização e criação de um módulo fonte de alimentação ficam comprometidas, pois torna o sistema impróprio a inovações.

Uma alternativa é utilizar em cada módulo um banco de baterias específico dimensionado para alimentar somente os dispositivos presentes no módulo. Ou seja, um sistema de alimentação distribuído. Com tal alternativa, cada módulo é dotado de baterias que são carregadas pelo barramento elétrico principal quando o sistema é alimentado. O controlador principal do sistema poderá obter as informações do nível de bateria de cada módulo individualmente através de mensagens provenientes de cada módulo e com isto monitorar a operação do veículo. O sistema de alimentação distribuído possui maior confiabilidade, pois caso ocorra uma falha da alimentação de um módulo, os outros módulos continuarão em funcionamento. Em contraste com o sistema de alimentação concentrado, em que se o módulo de alimentação falhar, todo o sistema é comprometido. Porém, o custo que envolve um sistema de alimentação distribuído é muito maior que o do sistema de alimentação concentrado.

Numa montagem mais avançada, é possível utilizar uma combinação dos dois tipos de sistema de alimentação num sistema de alimentação misto (concentrado e distribuído). Nesta configuração, os módulos vitais para o sistema robótico são providos de baterias. Enquanto que os módulos que não comprometem a operação emergencial do robô estes não possuem bateria interna e operam alimentados pelo barramento de alimentação que utiliza o mesmo cabeamento do barramento elétrico de comunicação.

4.4 Sistema Operacional

Um sistema operacional é um programa de computador especialmente feito para gerenciar o relacionamento entre aplicativos, dispositivos presentes no sistema, e os usuários. O trabalho básico de um sistema operacional é alocar os recursos do sistema computacional entre as rotinas que os requisitam. Naturalmente, a CPU (*Central Processing Unit* – unidade central de processamento) é o recurso mais escasso, então o agendamento de tarefas da CPU é o trabalho mais importante do sistema operacional (WOLF, 2001).

O sistema operacional é uma importante abstração, pois simplifica muito o código necessário para controlar os processos. Ao invés de escrever grandes quantidades de linhas de código para coordenar todos os processos, centralizamos estas funções em um único algoritmo ao organizar agendamentos. A simplificação da estrutura de controle deriva da aplicação de uma única estratégia de controle para todos os processos carregados na CPU. Desta forma, os sistemas operacionais devem proporcionar um mecanismo confiável para o agendamento e o chaveamento entre processos (WOLF, 2001).

O *kernel* é o menor² nível de qualquer sistema operacional e proporciona acesso a diversos serviços (acesso ao disco, gerenciamento de memória, agendamento de tarefas, e acesso a outros dispositivos físicos conectados) do núcleo do sistema a todos os outros programas e aplicações. Por isso, deve ser a primeira coisa a ser carregada na memória do computador quando o sistema inicia (Operating system - from Wikipedia, the free encyclopedia).

Um sistema operacional de tempo real (*Real-Time Operating System* – RTOS) é um tipo de sistema operacional desenvolvido especialmente para aplicações em tempo real. Exemplos deste tipo de aplicação são robôs industriais, controle industrial, equipamentos de pesquisa científica e sistemas embarcados em geral (JOHNSON, CLARKE, *et al.*, 2006).

² O *kernel* é o nível mais inferior hierarquicamente e no sentido do hardware eletrônico. Isto não se reflete no tamanho do código, podendo o *kernel* ocupar grande parte da codificação do sistema embarcado.

Um RTOS facilita a criação de um sistema de tempo real, mas não garante que o produto final será de tempo real; isto requer o correto desenvolvimento do *software*. Um RTOS não precisa ter necessariamente um alto fluxo de dados (*throughput*); ao contrário, um RTOS proporciona facilidades que, se usadas apropriadamente, garantem que os prazos do sistema possam ser alcançados de forma determinística (“*hard real-time*”) ou não determinística (“*soft real-time*”). Tipicamente, um RTOS irá utilizar algoritmos de agendamento especializados de forma a proporcionar ao desenvolvedor as ferramentas necessárias para produzir um ambiente determinístico no sistema final. Assim, um RTOS é avaliado pelo quão rapidamente e/ou de forma preditiva ele pode responder a um evento em particular. Consequentemente, mínimas interrupções e latência de chaveamento de processos são fatores-chaves num RTOS (JOHNSON, CLARKE, *et al.*, 2006).

4.5 Conceitos Auxiliares

Ainda, em nossas pesquisas, foram detectados três conceitos muito interessantes que foram incluídos nas análises de arquitetura nas próximas seções deste trabalho.

4.5.1 Cabo de Depuração

O cabo de depuração é apenas um cabo que se conecta ao veículo móvel utilizando o mesmo barramento da rede interna do sistema. Quando conectado, ele pode acessar todos os módulos individualmente, de forma a coletar os dados recebidos, verificar o funcionamento e até emular o funcionamento do sistema através de “mensagens teste”, ou seja, mensagens simulando o funcionamento ou mau funcionamento das partes do sistema.

Além do barramento de comunicação, o cabo de depuração pode conter alimentação para carga das baterias do sistema como um todo. Ou seja, dois fios do cabo são destinados a fornecer a corrente necessária para a carga de todos os módulos ao mesmo tempo.

Ainda, para que todos os módulos sejam acessados utilizando o mesmo cabo de depuração deve-se interligar o barramento com todos os módulos, inclusive os redundantes, de forma a permitir a leitura e o compartilhamento dos dados a qualquer momento. Isto pode ser feito sem maiores problemas desde que sejam atribuídos identificadores nas mensagens diferentes para cada um dos módulos do veículo como um todo.

4.5.2 Pulso de Sincronismo

Assim como no corpo humano existe um pulso de sincronismo (batimento cardíaco), um conceito auxiliar de um pulso de sincronismo é importante quando se utiliza um sistema modular interligado como uma rede. Neste caso, o pulso de sincronismo pode ser utilizado para verificar quais módulos estão ativos. A cada pulso, todos os módulos detectam a presença de um módulo controlador principal e cada módulo deve responder ao pulso com informações sobre o seu estado atual. Desta forma, o módulo controlador principal é informado sobre o estado de todos os módulos. Caso, o pulso de sincronismo não ocorra, outro módulo pode assumir o controle do sistema. Caso, alguma resposta falte isto indicará que algum módulo não está mais operando; neste caso, o controlador principal deve se adaptar excluindo o módulo defeituoso dos seus cálculos.

O barramento CAN utilizado nos projetos de foguetes experimentais de Kennan e McFadden (1999) ou de Santos (2006) utilizam uma mensagem de sincronismo onde são informados os estados de todos os módulos do barramento e assim são monitoradas a qualidade das informações recebidas no barramento.

4.5.3 Módulo de Emergência

De forma a garantir o funcionamento ininterrupto do sistema robótico, pode ser utilizado um módulo redundante ao módulo central de processamento que monitora a atividade deste módulo. Caso o módulo central falhe, o módulo de emergência entra em ação.

O módulo de emergência pode funcionar exatamente como o módulo central ou como um executor de procedimentos de emergência, de forma a possibilitar a recuperação do sistema robótico móvel, caso ele esteja em movimento e ocorra uma falha.

Este módulo é especialmente importante se o sistema móvel for um dispositivo voador não tripulado ou qualquer dispositivo móvel que necessita de um controle constante para se manter em movimento.

5 Critérios e Resultados

5.1 Sistema Autônomo Completo

Baseado nas características de um robô autônomo que desempenha ações semelhantes às atividades mentais humanas. E ainda que tal robô deva se comportar como um serviçal em atividades domésticas. Ou seja, atendendo a comandos complexos que envolvam pequenas decisões pelo caminho e um pré-conhecimento de tarefas.

5.1.1 Critérios / Requisitos

Conforme foi visto nos capítulos anteriores, necessita-se que o robô atenda as seguintes atividades mentais representadas pelos critérios a seguir:

1. Sensação/Percepção: Um sistema autônomo completo deve ter a possibilidade de receber e processar os cinco sentidos (visão, audição, tato, olfato e paladar) do ser humano ou qualquer equivalente;
2. Orientação: Um sistema autônomo completo deve saber o seu estado atual (situação do seu corpo);
3. Orientação: Um sistema autônomo completo deve saber onde está (localmente e globalmente);
4. Orientação: Um sistema autônomo completo deve saber reconhecer o lugar atual comparando com o seu passado;
5. Ação Motora: Um sistema autônomo completo deve ser modular escalável incremental de forma a se adaptar a qualquer plataforma mecânica e atingir seus objetivos educacionais.
6. Geral: O sistema autônomo completo deve ser robusto para suportar diversos problemas que os sistemas embarcados podem enfrentar;
7. Geral: O sistema autônomo completo deve ser de baixo custo para ser implementado em diversas aplicações independente do tamanho do projeto;

8. Geral: O sistema autônomo completo deve utilizar tecnologias com alto índice de confiabilidade;
9. Geral: O sistema autônomo completo deve utilizar tecnologias disponíveis e de fácil acesso ao mercado brasileiro de componentes eletrônicos.

Com estes critérios será definida a arquitetura do sistema autônomo completo nas seções a seguir.

5.2 Topologia Escolhida

Segundo o critério número 5 o sistema autônomo completo deve ser modular escalável incremental de forma a se adaptar a diversas plataformas mecânicas. Das seis topologias estudadas, a de maior capacidade incremental escalável, ou seja, a que possui maior facilidade na adição de módulos é a arquitetura de barramento serial.

Porém, para tornar o sistema mais robusto e atender o critério número 6, pode-se substituir o controlador central de forma a termos um mecanismo de arbitração executado em hardware internamente em cada um dos módulos (BECKE, BORGERT, *et al.*, 2000; KENNAN e MCFADDEN, 1999).

Assim, a arquitetura de *barramento serial com arbitração em hardware*, que ao utilizar um barramento serial com o controle de prioridades arbitradas por hardware garante a estrutura modular incremental e uma maior robustez ao sistema.

Na Figura 34, temos um exemplo de um sistema embarcado que utiliza uma arquitetura serial com arbitração em hardware.

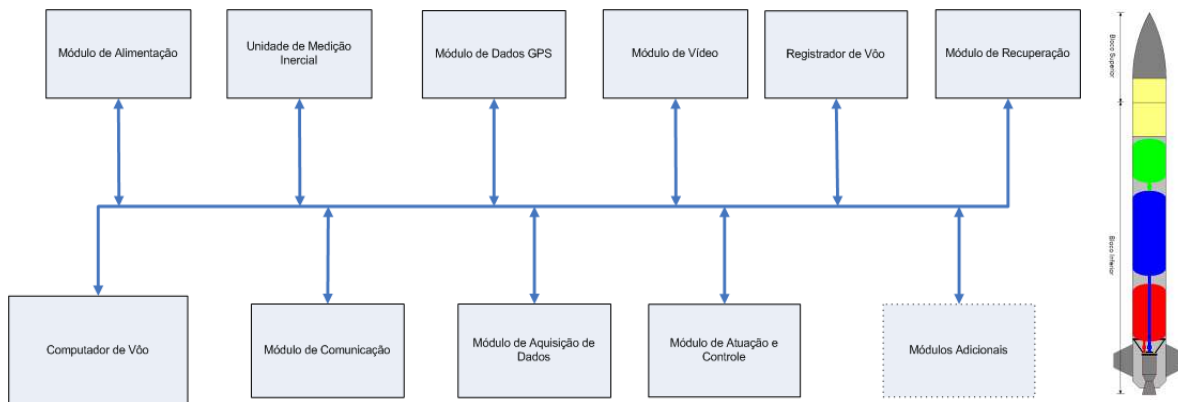


Figura 34 – Arquitetura básica da aviônica do foguete Sonda-L.

Fonte: (SANTOS, 2006).

5.3 Barramento Elétrico de Comunicação Escolhido

Para a escolha do barramento elétrico de comunicação foram tomados os critérios números 7, 8 e 9 como base. De forma que o barramento elétrico deve ser de baixo custo para que possa ser utilizado em larga escala sem agregar muito custo nos módulos pequenos, ter confiabilidade (deve ser a prova de erros) e o barramento deve ser comercial e seus controladores facilmente encontrados no mercado para que a montagem dos módulos não seja atrasada por falta de componentes.

Baseado nos critérios gerais acima e na topologia de barramento serial com arbitragem em *hardware* foi escolhido o barramento CAN (*Controller Area Network* – ISO11898) (BOSCH, 1991; CAN IN AUTOMATION GMBH, 1998; CAN IN AUTOMATION GMBH, 1998; CAN IN AUTOMATION GMBH, 1994) (que é explicado com maiores detalhes no Apêndice C) para dar continuidade a nossa pesquisa. Porém, o barramento CAN possui diversos semelhantes que também podem ser utilizados desde que também se enquadrem nos critérios e seja possível atender ao fluxo de dados necessário ao sistema.

5.4 Barramento de Alimentação Escolhido

De acordo com os critérios número 6, 7 e 8; de forma a garantir a confiabilidade, a robustez e o baixo custo do sistema autônomo completo foi escolhido o sistema de alimentação misto. Conforme exemplifica a Figura 35. Ou seja, para os módulos básicos essenciais pode-se utilizar uma alimentação distribuída, garantindo a confiabilidade; e para os demais módulos pode-se utilizar uma alimentação concentrada na forma de um Módulo de Alimentação com a finalidade de diminuir as dimensões e custos de cada um destes dispositivos. E conforme outros módulos sejam adicionados podem ser adicionados outros Módulos de Alimentação de forma a aumentar a disponibilidade de energia do sistema.

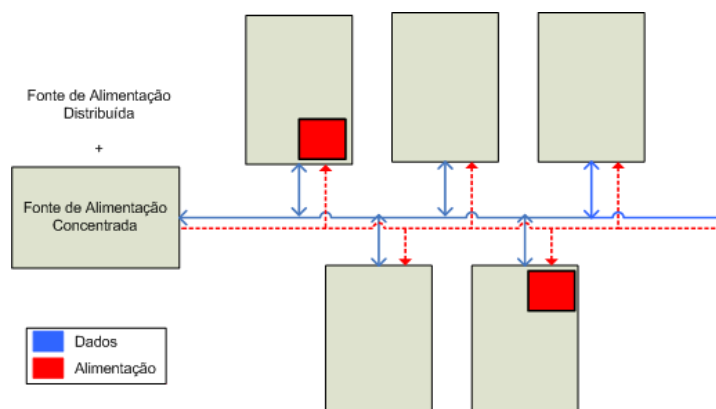


Figura 35 – Barramento de Alimentação Misto.

Fonte: (SANTOS, 2006)

Desta forma, os blocos referentes a alimentação, sejam eles o Módulo de Alimentação ou mesmo uma parte interna de qualquer outro módulo essencial, são formados por circuitos de carga de baterias. A escolha do tipo de bateria depende das necessidades e aplicações específicas de cada projeto. O Apêndice D possui um breve comparativo dos tipos de baterias existentes no mercado atual.

Quando o sistema está em operação, o Módulo de Alimentação deve alimentar todo o sistema. Porém, em caso de falha neste módulo, as baterias internas dos módulos

essenciais entram em operação garantindo a funcionalidade do sistema por mais alguns minutos.

5.5 Sistema Operacional

Existem diversos sistemas operacionais de tempo real no mercado e quase todos podem ser utilizados. A utilização de um sistema operacional se faz necessária somente nos módulos em que temos configurações ou comunicações com o espaço exterior. De forma a garantir a confiabilidade de um sistema determinístico e preemptivo. Porém, a definição e delineamento do sistema operacional e suas interações estão fora do escopo deste trabalho.

5.6 Resultados

Avaliando as características de cada tipo de sistema de orientação e os critérios de definição do sistema autônomo completo, ou seja, completamente autossuficiente em qualquer terreno temos a necessidade dos seguintes módulos abaixo:

- Um Módulo de Sensores
 - Táteis, Inclinação, e Gerais – não ligados a guiagem – de forma a permitir os cinco ou mais sentidos (critério número 1);
- Um sistema de guiagem odométrico
 - para poder se localizar localmente (critério número 3);
- Um sistema de navegação inercial
 - para poder identificar seu estado atual e sua atitude (critério número 2);
- Um sistema de localização GPS

- para dar apoio ao sistema de navegação inercial e gerenciar o controle de mapas garantindo a localização global (critério número 3);
- Um sistema de guiagem por mapas
 - para poder reconhecer os ambientes (critério número 4);
- Um sistema de guiagem por marcas artificiais
 - para poder identificar marcas de referencias absolutas (critério número 4).

Onde os cinco sistemas e o módulo de sensores, operando juntos, permitiriam um robô móvel operar em diversos ambientes, reconhecendo a sua localização e sua orientação, possibilitando criar mapas e traçar rotas de destino, utilizando como referências absolutas intermediárias o receptor GPS e as marcas artificiais.

Este sistema “completo” se assemelha ao sistema de navegação humano onde a localização é feita a partir do cruzamento de diversos sensores e dados adquiridos de diversas fontes, e a partir destes dados, são feitas comparações e análises de forma a termos uma cognição e determinação de rotas satisfatórias.

Porém, para implementar tal sistema é necessário um grande investimento monetário de forma a permitir uma grande capacidade computacional e sensores precisos.

Fazendo-se uma combinação mais simples dos sistemas citados é possível ter um sistema cuja navegação seja razoavelmente precisa com um custo relativamente baixo. Assim, cabe ao projetista decidir quais serão os métodos utilizados e qual a precisão possível para o custo desejado.

Com base na topologia de barramento serial com arbitração em hardware, no barramento elétrico de comunicação CAN, no barramento de alimentação do tipo misto e expandindo-se os seus sistemas acima descritos em módulos pode-se verificar a Figura 36 onde estão indicados todos os módulos necessários para um sistema autônomo que opere em todos os terrenos e em diversos tipos de ambientes. Ou seja, o sistema autônomo completo.

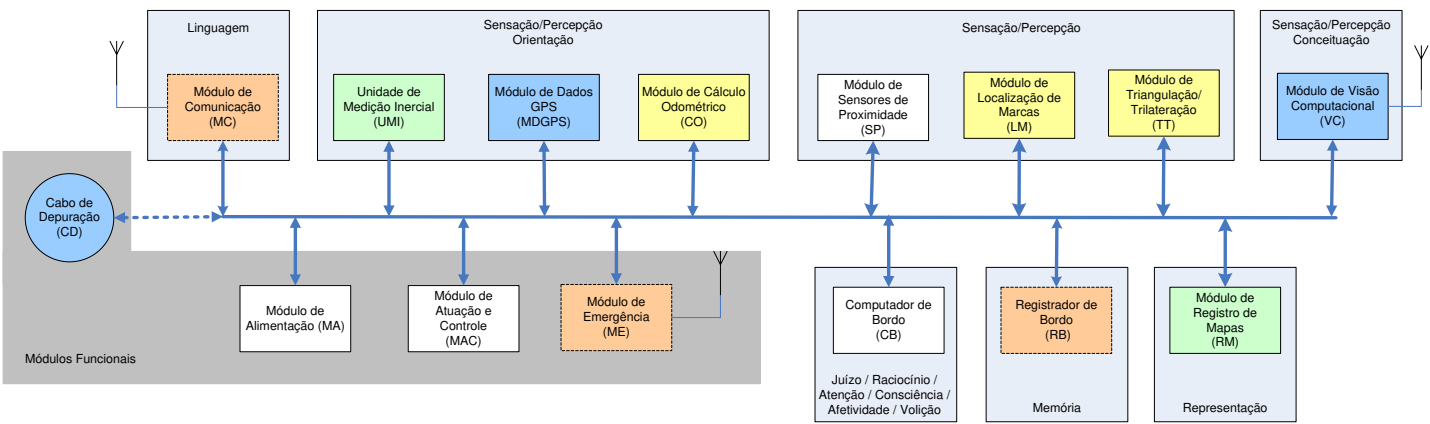


Figura 36 – Sistema autônomo completo com identificação das funcionalidades e atividades mentais humanas equivalentes.

Onde cada bloco referencia um tipo de sensor ou sistema de navegação estudado nos capítulos anteriores e deve incluir todo o processamento necessário para tornar a medida útil para ser transmitida no barramento e ser tratada em nível de tomada de decisão. As seções a seguir descrevem cada módulo do sistema. Com base nos trabalhos de Santos (2006), Kennan, McFadden (1999) e Silva (2011) foram preparados diagramas de bloco e fluxogramas de funcionamento dos módulos listados abaixo:

- Módulo de Sensores de Proximidade (SP);
- Módulo de Cálculo Odométrico (CO);
- Módulo de Unidade de Medição Inercial (UMI);
- Módulo de Triangulação/Trilateração (TT);
- Módulo de Dados GPS (MDGPS);
- Módulo de Registro de Mapas (RM);
- Módulo de Localização por Marcas (LM);
- Módulo de Visão Computacional (VC);
- Módulo de Atuação e Controle (MAC);
- Módulo de Alimentação (MA);
- Computador de Bordo (CB);
- Módulo de Comunicação (MC);
- Módulo de Emergência (ME);
- Registrador de Bordo (RB);
- Cabo de Depuração (CD).

Tais módulos serão detalhados nas seções a seguir. Contudo, em cada módulo foi posicionado um controlador como centro de controle de dados e interfaceamento com a rede. A escolha deste controlador fica a critério do projetista do sistema. Como opções podem ser utilizadas diversas tecnologias como microcontroladores, processadores, processadores digitais de sinais (DSP – Digital Signal Processor), dispositivos de lógica programável complexa (CPLD – Complex Programmable Logic Device), vetores de portas

lógicas programáveis por campos (FPGA – Field Programmable Gate Array), sistemas em chips (SoC – System on Chip) etc.

5.6.1 Módulo de sensores de proximidade (SP)

O módulo de sensores de proximidade (SP) é responsável por detectar objetos próximos através de sensores de ultrassom, “laser rangefinder” ou “bumpers”. Porém, é neste módulo que devem ser inseridos outros sensores para determinação de parâmetros físicos internos ou externos ao veículo. Sensores funcionais podem ser incluídos para detectar tensão de alimentação do sistema, temperatura do sistema, etc. Sensores externos ou ambientais podem monitorar temperatura externa, condições climáticas, pressão atmosférica, etc.

A Figura 37 mostra um diagrama de blocos proposto para o módulo SP e a Figura 38 mostra um fluxograma simplificado do software proposto para o controlador do módulo SP.

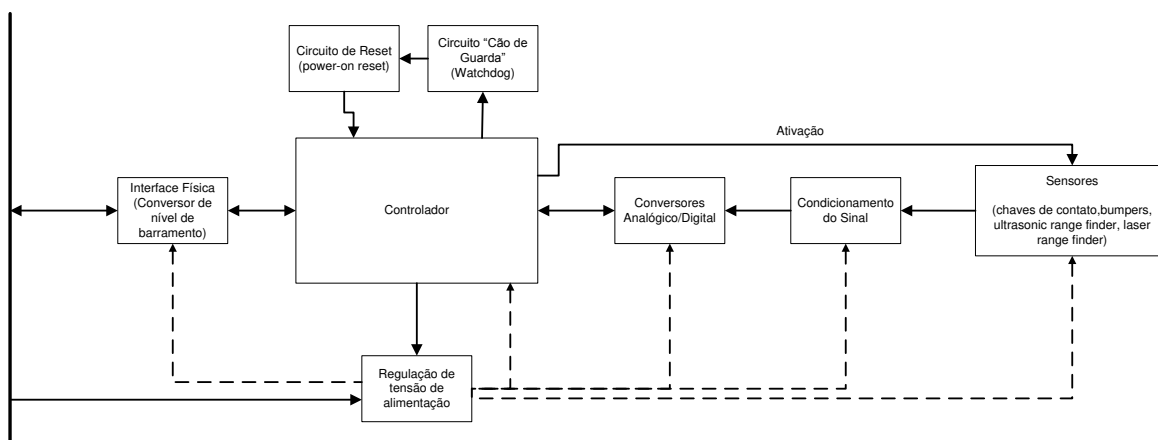


Figura 37 – Diagrama de exemplo do circuito eletrônico do módulo de sensores de proximidade (SP).

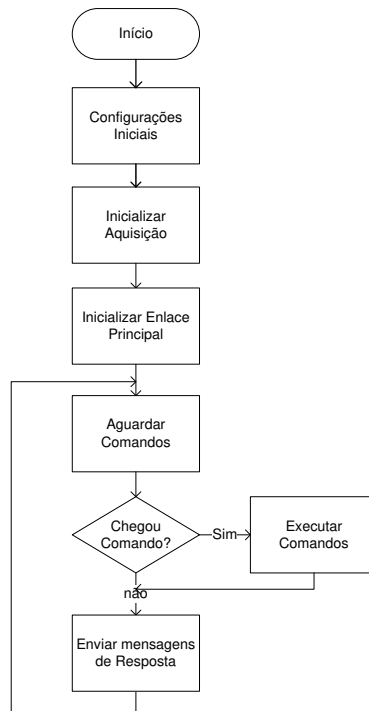


Figura 38 – Fluxograma exemplo do software recomendado para o módulo SP.

O módulo SP deve aceitar somente mensagens que são relativas às suas funcionalidades. Contudo, de forma a monitorar os eventos no veículo e coletar dados para o sistema, o SP tem a capacidade de gerar mensagens de DADOS e EVENTO. As mensagens do módulo SP são sugeridas no Apêndice E.

5.6.2 Módulo de cálculo odométrico (CO)

O módulo de cálculo odométrico é responsável por interpretar os dados dos *encoders* das rodas e calcular a posição estimada do veículo. As principais tarefas deste módulo são:

- Determinar o sentido de giro de cada roda monitorada;
- Determinar quantas voltas cada roda deu em torno de si mesma;

- Calcular a variação de posição do veículo robótico;
- Calcular a velocidade média;
- Estimar a aceleração média;
- Enviar todos os resultados obtidos para a rede.

A Figura 39 mostra um diagrama de blocos proposto para o módulo CO e a Figura 40 mostra um fluxograma simplificado do software proposto para o controlador do módulo CO.

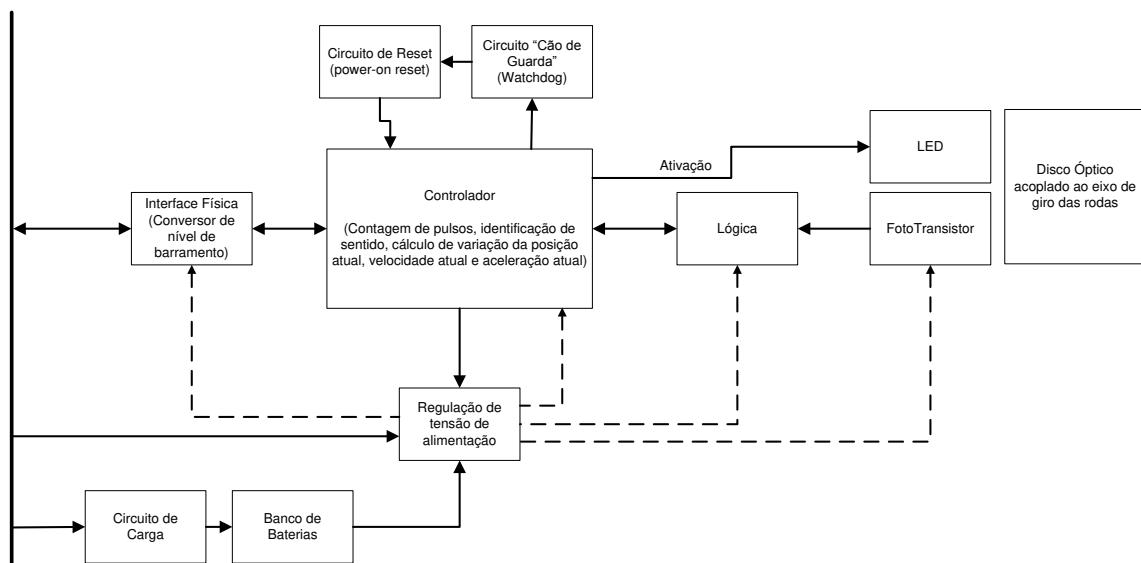


Figura 39 – Diagrama de exemplo do circuito eletrônico do módulo de cálculo odométrico (CO).

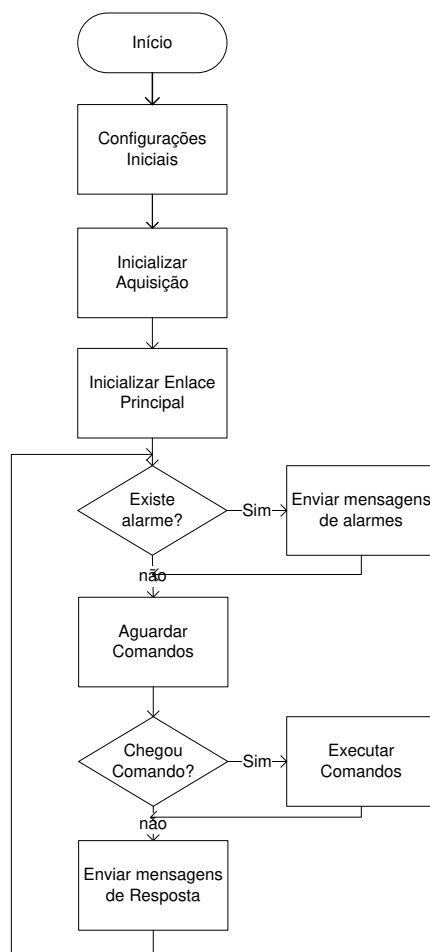


Figura 40 – Fluxograma exemplo do software recomendado para o módulo CO.

O módulo CO aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição e velocidade para o computador de bordo. As mensagens do módulo CO são sugeridas no Apêndice E.

5.6.3 Módulo de Unidade de Medição Inercial (UMI)

O Módulo de Unidade de Medição Inercial (UMI) é responsável pelas leituras do sistema de navegação inercial. Assim, deve ter sensores para medir aceleração linear (acelerômetros), velocidade angular (giroscópios), temperatura da unidade (termopares) e pressão atmosférica (barômetro). Os dados obtidos nestas medições devem ser usados para

calcular a posição, a orientação, e a velocidade do veículo robótico móvel. As principais tarefas deste módulo são:

- Medir a aceleração linear (eixos X, Y e Z);
- Medir a aceleração angular (planos α , β e γ);
- Calcular a aceleração atual;
- Calcular a velocidade atual;
- Calcular a posição atual;
- Calcular a orientação atual;
- Enviar todos os dados obtidos para a rede.

A Figura 41 mostra um diagrama de blocos proposto para a UMI e a Figura 42 mostra um fluxograma simplificado do software proposto para o controlador da UMI.

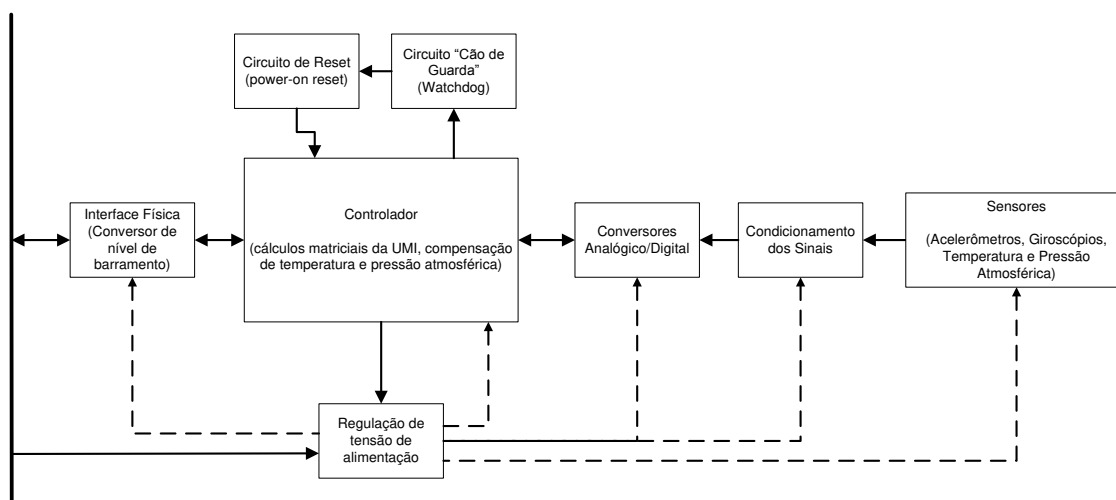


Figura 41 – Diagrama de exemplo do circuito eletrônico do módulo de unidade de medição inercial (UMI).

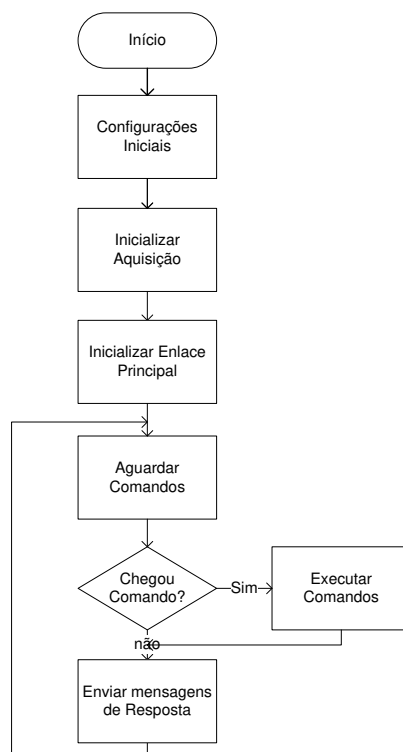


Figura 42 – Fluxograma exemplo do software recomendado para o módulo UMI.

O módulo UMI aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição, velocidade e aceleração lineares e angulares para o computador de bordo. As mensagens do módulo UMI são sugeridas no Apêndice E.

5.6.4 Módulo de triangulação/trilateração (TT)

O módulo de triangulação/trilateração (TT) é responsável pela localização do veículo dentro de ambientes controlados através dos métodos de localização por triangulação ou trilateração dependendo do sistema escolhido.

A Figura 43 mostra um diagrama de blocos proposto para o módulo TT e a Figura 44 mostra um fluxograma simplificado do software proposto para o controlador do módulo TT.

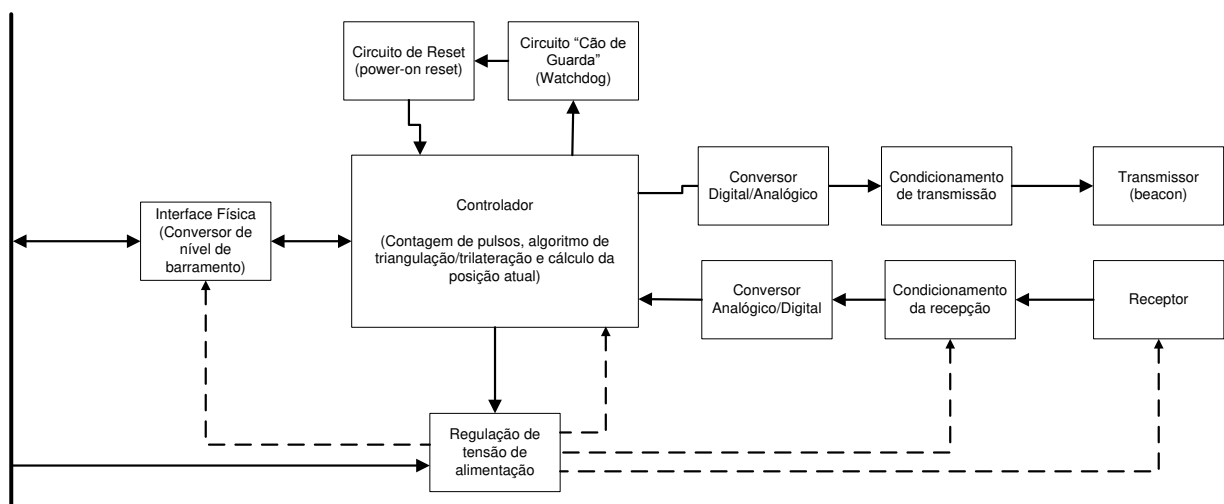


Figura 43 – Diagrama de exemplo do circuito eletrônico do módulo de triangulação/trilateração (TT).

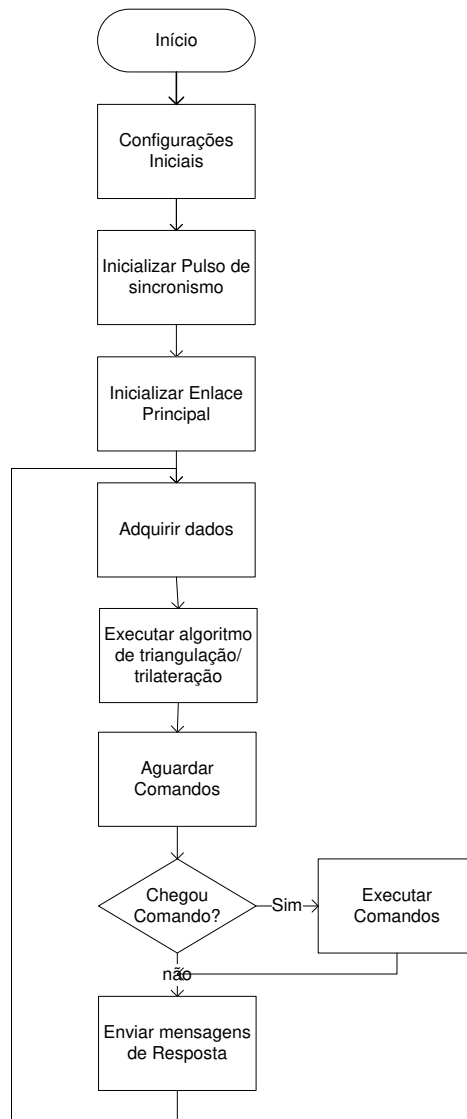


Figura 44 – Fluxograma exemplo do software recomendado para o módulo TT.

O módulo TT aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição segundo o eixo de coordenadas do ambiente para o computador de bordo. As mensagens do módulo TT são sugeridas no Apêndice E.

5.6.5 Módulo de dados GPS (MDGPS)

O módulo de dados GPS é responsável pela recepção dos sinais GPS a partir de algum receptor padrão disponível no mercado e embarcar os dados para a rede do sistema robótico. Este módulo consiste basicamente em um receptor GPS, sua antena e o circuito para embarcar os dados do GPS na rede de comunicação. Ou seja, um controlador e seus circuitos adjacentes e o conversor de nível de tensão para o barramento de dados.

A Figura 45 mostra um diagrama de blocos proposto para o módulo MDGPS e a Figura 46 mostra um fluxograma simplificado do software proposto para o controlador do módulo MDGPS.

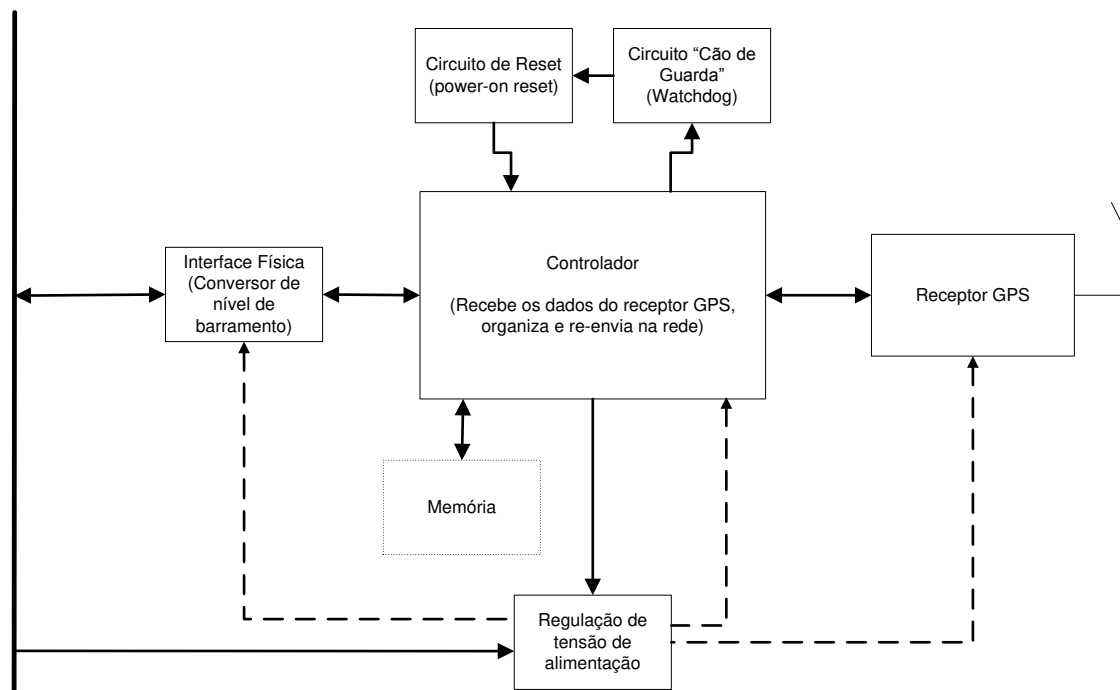


Figura 45 – Diagrama de exemplo do circuito eletrônico do módulo de dados GPS (MDGPS).

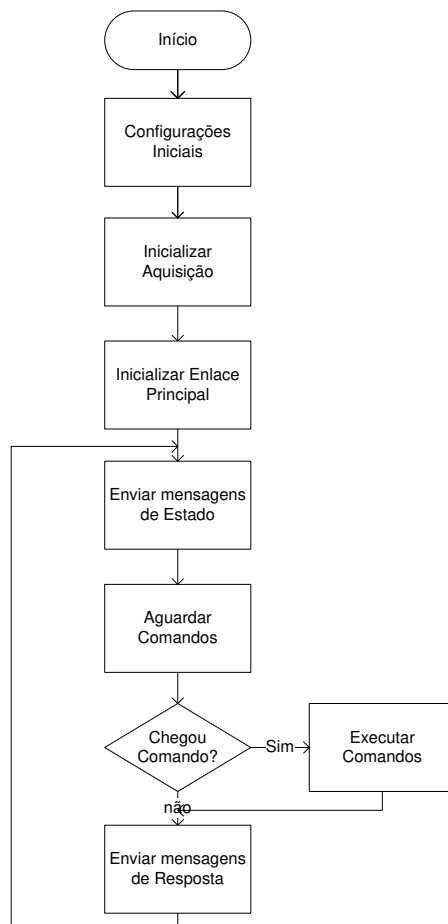


Figura 46 – Fluxograma exemplo do software recomendado para o módulo MDGPS.

O módulo MDGPS aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição segundo os dados obtidos dos satélites de GPS para o computador de bordo. As mensagens do módulo MDGPS são sugeridas no Apêndice E.

5.6.6 Módulo de registro de mapas (RM)

O módulo de registro de mapas (RM) é responsável por utilizar os dados obtidos por outros módulos para criar mapas ou indicar regiões de probabilidade.

A Figura 47 mostra um diagrama de blocos proposto para o módulo RM e a Figura 48 mostra um fluxograma simplificado do software proposto para o controlador do módulo RM.

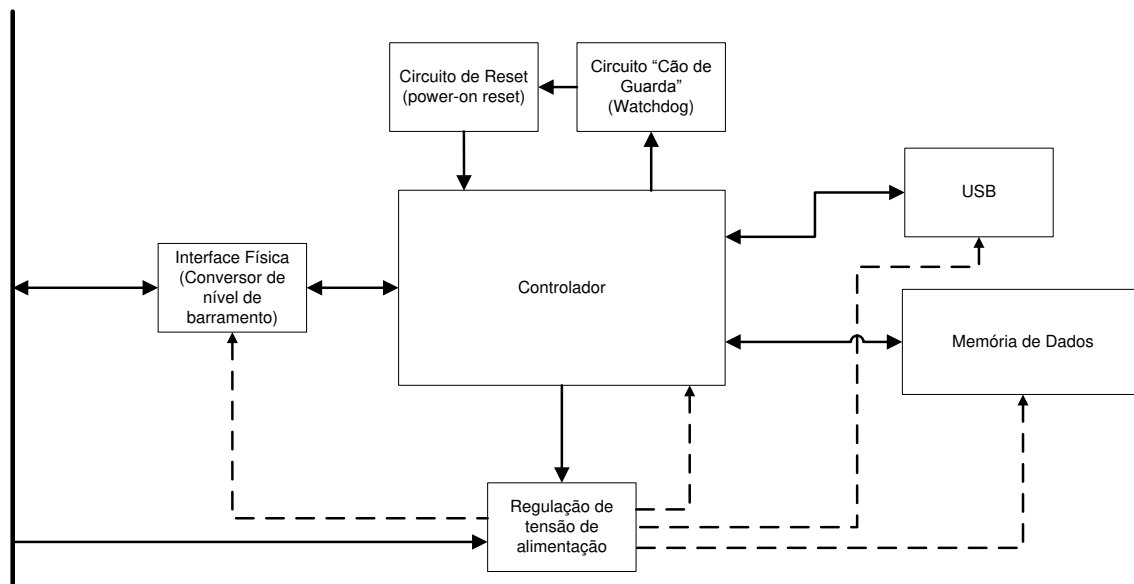


Figura 47 – Diagrama de exemplo do circuito eletrônico do módulo de registro de mapas (RM).

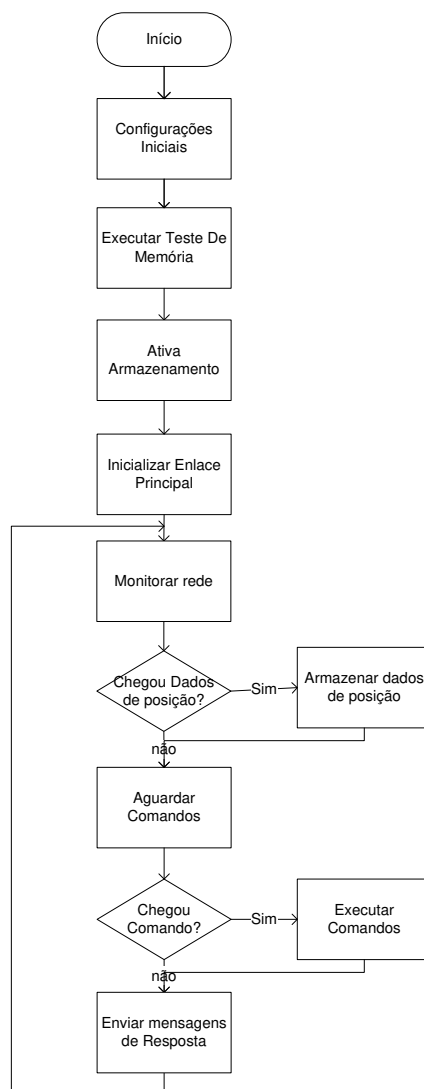


Figura 48 – Fluxograma exemplo do software recomendado para o módulo RM.

O módulo RM aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição segundo os dados obtidos dos sensores da rede e corrigidos pelos mapas para o computador de bordo. As mensagens do módulo RM são sugeridas no Apêndice E.

5.6.7 Módulo de Localização por marcas (LM)

O módulo de localização por marcas (LM) é responsável pelos recursos de identificação e memória de marcas. Este módulo consiste basicamente em uma câmera, os circuitos para embarcar as imagens no controlador e os componentes para a rede de comunicação. Neste módulo deverão estar presentes alguns algoritmos de identificação de marcas e cálculos de estimativa de posição e atitude a partir das marcas lidas.

A Figura 49 mostra um diagrama de blocos proposto para o módulo LM e a Figura 50 mostra um fluxograma simplificado do software proposto para o microcontrolador do módulo LM.

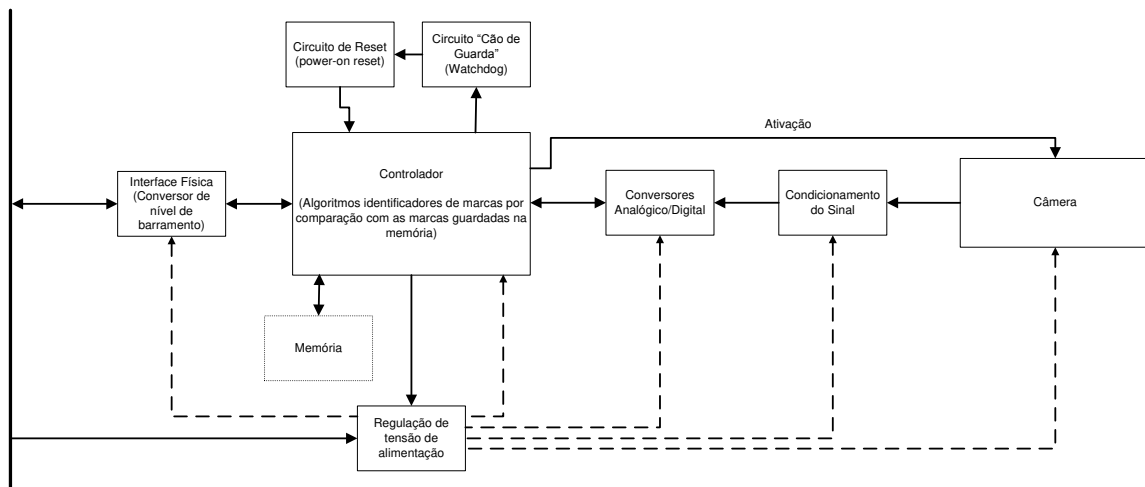


Figura 49 – Diagrama de exemplo do circuito eletrônico do módulo de localização por marcas (LM).

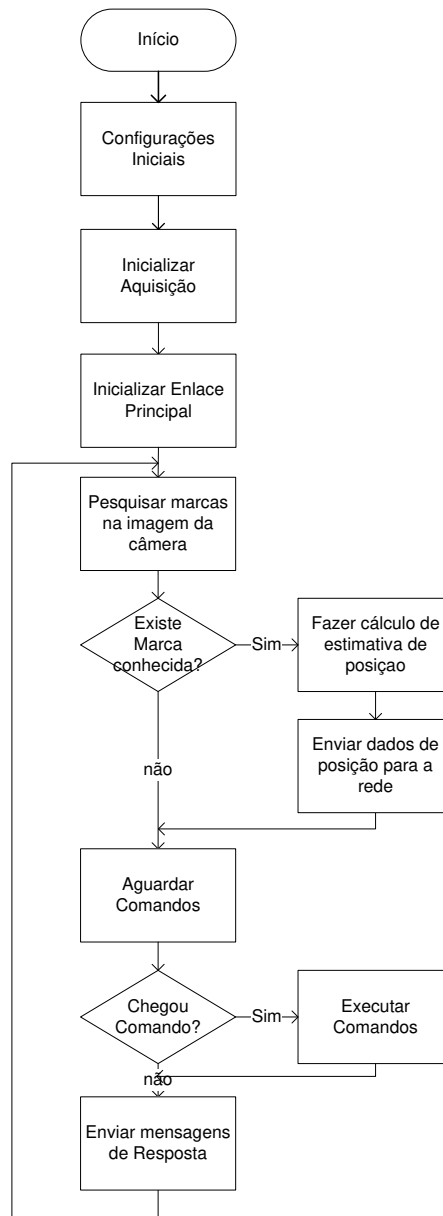


Figura 50 – Fluxograma exemplo do software recomendado para o módulo LM.

O módulo LM aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição segundo o eixo de coordenadas do ambiente para o computador de bordo. As mensagens do módulo LM são sugeridas no Apêndice E.

5.6.8 Módulo de visão computacional (VC)

O módulo de visão computacional (VC) é responsável pela aquisição de imagens e todo o processamento de identificação de marcas/objetos bem como os caminhos. Este módulo consiste basicamente em uma câmera, os circuitos para embarcar as imagens no controlador e os componentes para a rede de comunicação. Além disto, é possível inserir um circuito transmissor de vídeo com antena própria que funcionará como uma telemetria mostrando pessoas fora do veículo o que ele está captando. Neste módulo deverão estar presentes alguns algoritmos de identificação de objetos e marcas e cálculos de estimativa de posição e atitude a partir das marcas lidas. Este módulo se assemelha na aplicação ao módulo LM e detém vantagens em ambientes não controlados.

A Figura 51 mostra um diagrama de blocos proposto para o módulo VC e a Figura 52 mostra um fluxograma simplificado do software proposto para o controlador do módulo VC.

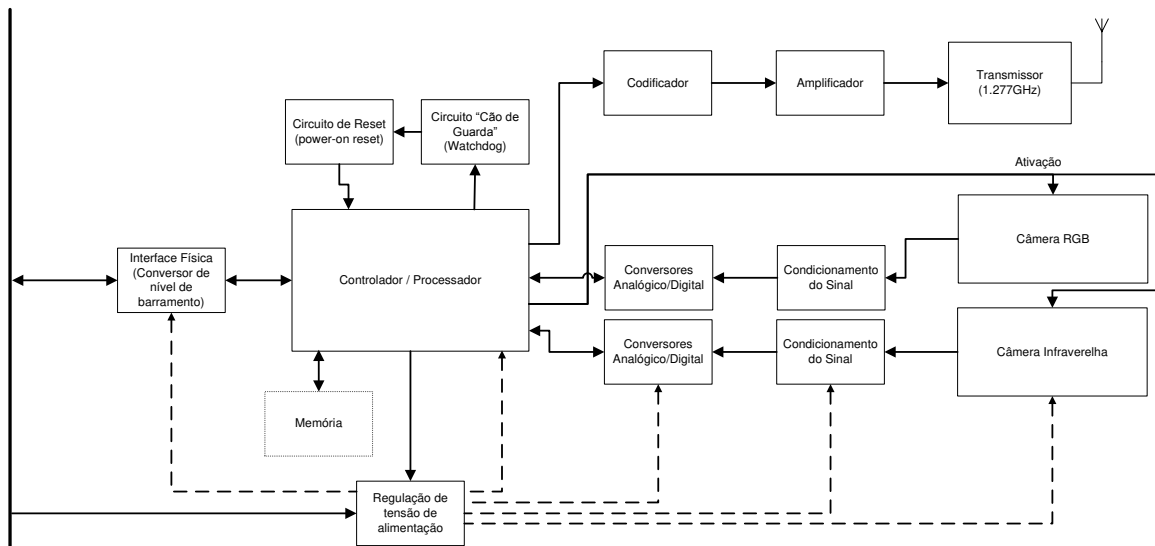


Figura 51 – Diagrama de exemplo do circuito eletrônico do módulo de visão computacional (VC).

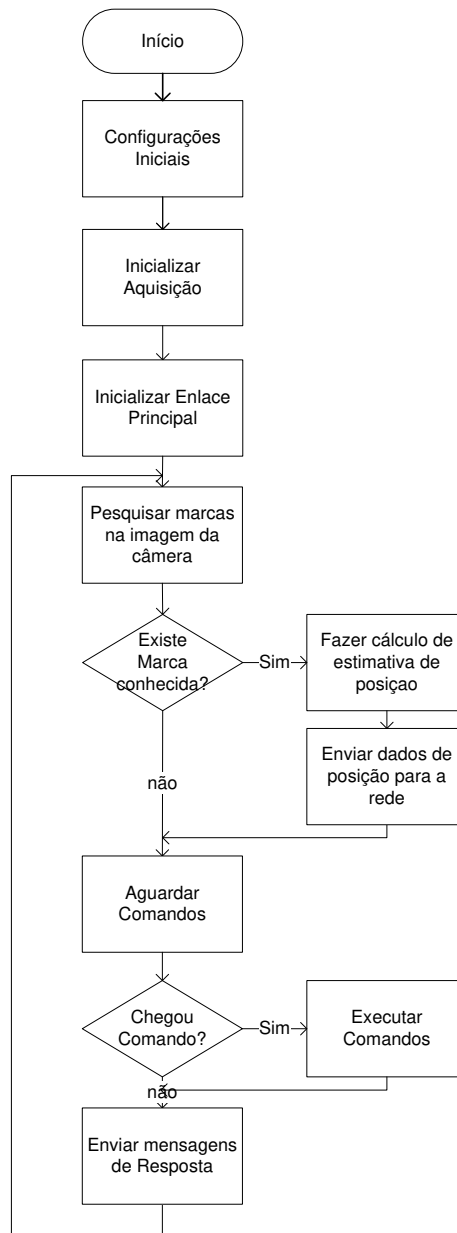


Figura 52 – Fluxograma exemplo do software recomendado para o módulo VC.

O módulo VC aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de posição de objetos e marcas segundo o eixo de coordenadas

do veículo para o computador de bordo. As mensagens do módulo VC são sugeridas no Apêndice E.

Os outros blocos presentes na Figura 36 são referentes a operação do sistema. Tais blocos são descritos a seguir.

5.6.9 Módulo de atuação e controle (MAC)

O módulo de atuação e controle (MAC) é responsável pela execução e controle de todos os movimentos do veículo. Desta forma, ele dependerá intrinsecamente da mecânica do veículo e seus acionamentos. Este módulo funciona como executor mecânico de todos os pedidos feitos a ele pela rede de dados. Desta forma, o MAC deve ser capaz de identificar o estado de sua “carga” e transmitir dados de estado para o sistema, além de permitir ativar ou desativar a execução de algum atuador em resposta a algum comando recebido. Ainda, deve ser possível ajustar a atuação para ocorrer somente algum tempo depois do comando ser recebido; isto pode ser feito através de um temporizador configurável pré-ajustado.

A Figura 53 mostra um diagrama de blocos proposto para o módulo MAC e a Figura 54 mostra um fluxograma simplificado do software proposto para o controlador do módulo MAC.

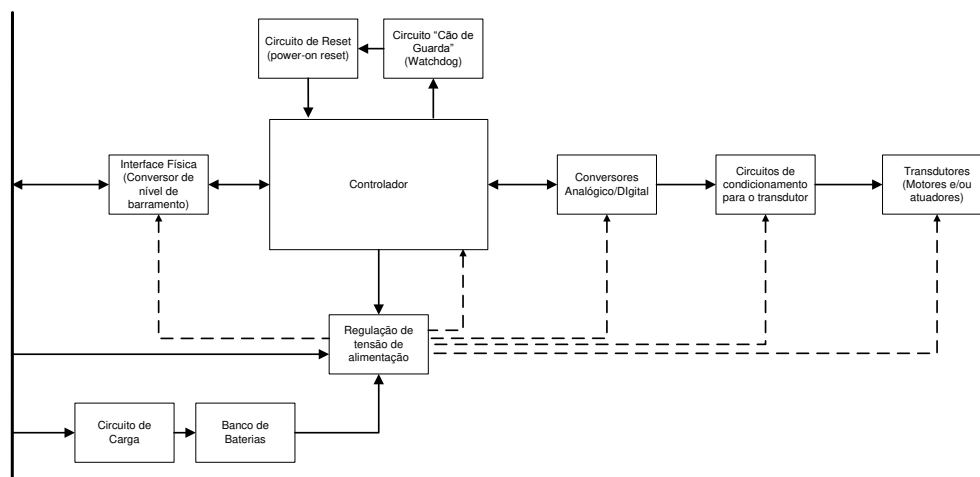


Figura 53 – Diagrama de exemplo do circuito eletrônico do módulo de atuação e controle (MAC).

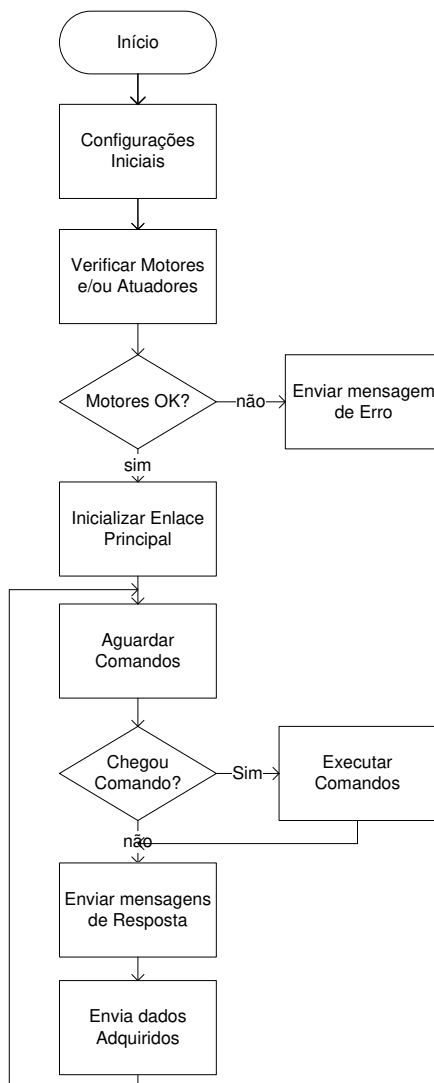


Figura 54 – Fluxograma exemplo do software recomendado para o módulo MAC.

O módulo MAC aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens para o computador de bordo com informações sobre o estado dos atuadores. As mensagens do módulo MAC são sugeridas no Apêndice E.

5.6.10 Módulo de alimentação (MA)

O módulo de alimentação é responsável pelo armazenamento de energia. Ou seja, é neste módulo em que estão as baterias que alimentam o sistema. E é através deste módulo que os sistemas devem obter o suprimento de energia necessário para o seu funcionamento. Em casos de emergência, os módulos CO, MAC, CB e RB possuem uma bateria interna de forma a possibilitar o funcionamento do sistema em caso de falha do módulo MA.

O Módulo de Alimentação deve ter comunicação direta com todos os outros módulos, através do barramento principal e deve monitorar o nível da carga da bateria de forma a prever quanto de carga se tem armazenada na mesma e com isto prever o quanto de energia se tem disponível durante a utilização do veículo robótico móvel.

A Figura 55 mostra um diagrama de blocos proposto para o módulo MA e a Figura 56 mostra um fluxograma simplificado do software proposto para o controlador do módulo MA.

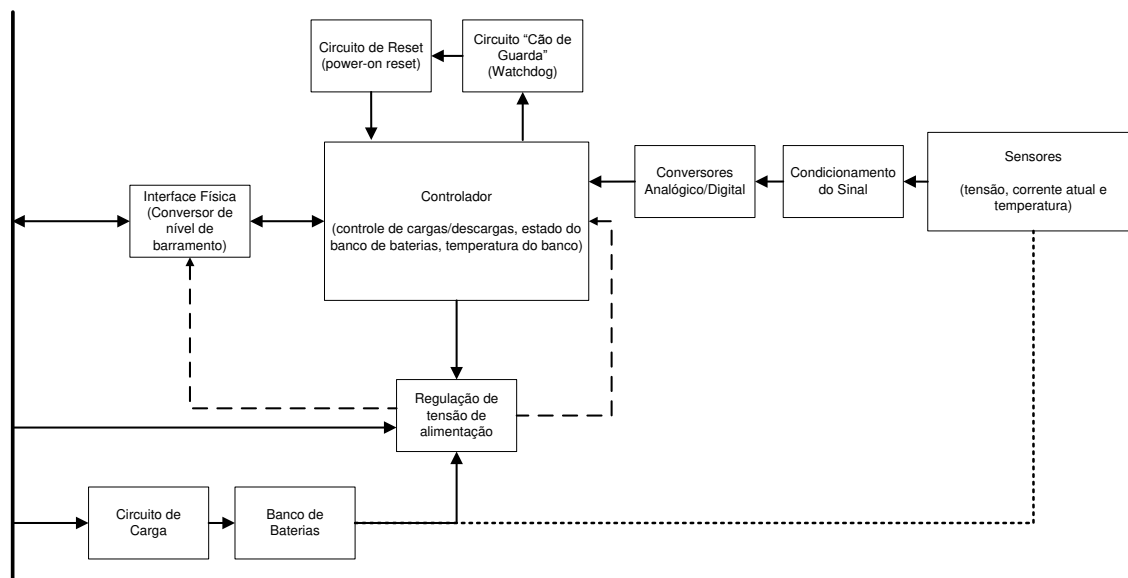


Figura 55 – Diagrama de exemplo do circuito eletrônico do módulo de alimentação (MA).

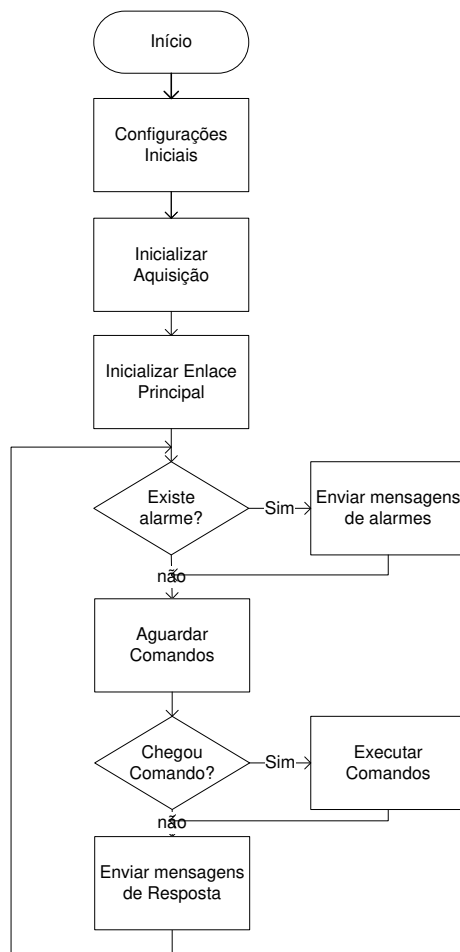


Figura 56 – Fluxograma exemplo do software recomendado para o módulo MA.

O módulo MA aceita somente mensagens diretamente enviadas a ele para a sua operação. E gera mensagens de tensão de alimentação, tempo até a descarga e o número de ciclos de carga/descarga para o computador de bordo. As mensagens do módulo MA são sugeridas no Apêndice E.

5.6.11 Computador de bordo (CB)

O computador de bordo (CB) é o “cérebro” do sistema, responsável por todas as ações tomadas pelo veículo. Este subsistema é responsável pelo sequenciamento dos estados do veículo baseado nas máquinas de estado internas e entradas dos outros módulos. Uma vez que uma fase do sistema tenha sido detectada, o computador de bordo é responsável em tomar alguma ação necessária.

A Figura 57 mostra um diagrama de blocos proposto para o módulo CB e a Figura 58 mostra um diagrama de estados simplificado mostrando os estados de Inicialização, Depuração, Pré-Operação, Operação e Emergência do software proposto para o controlador do módulo CB. A Figura 59 mostra um exemplo de fluxogramas para os estados de INICIALIZAÇÃO e DEPURAÇÃO. A Figura 60 mostra um exemplo de fluxogramas para os estados de PRÉ-OPERAÇÃO, OPERAÇÃO e EMERGÊNCIA.

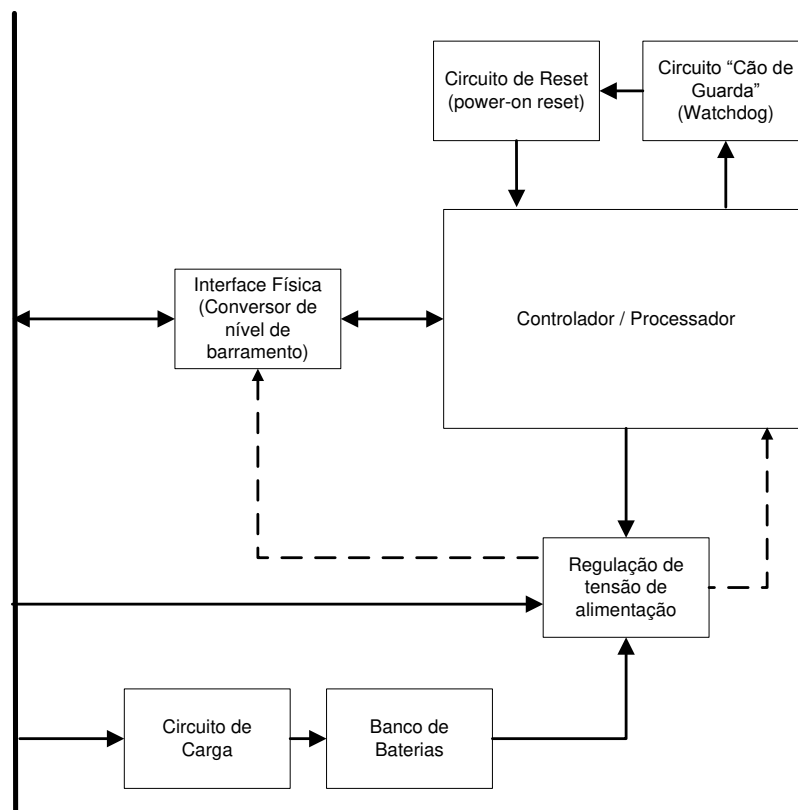


Figura 57 – Diagrama de exemplo do circuito eletrônico do computador de bordo (CB).

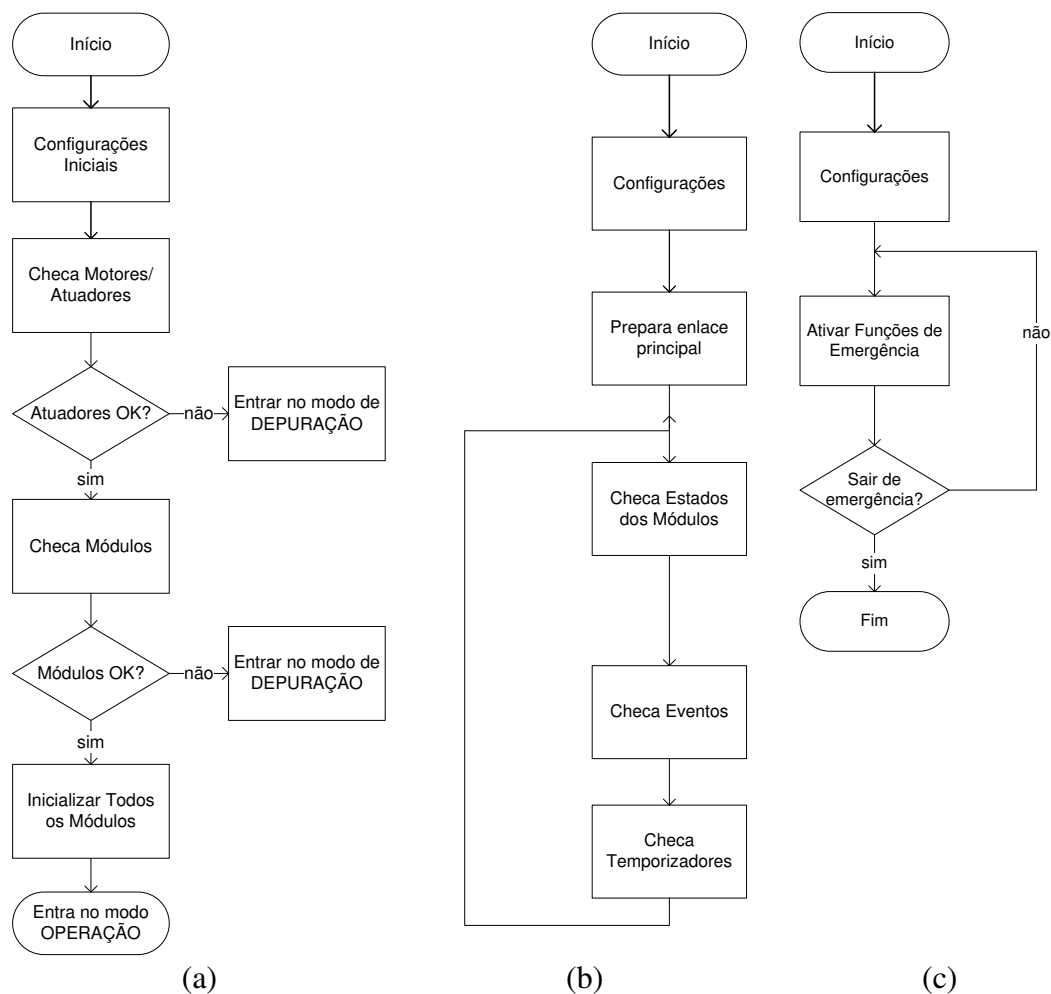


Figura 60 – Fluxogramas exemplo para os estados de PRÉ-OPERAÇÃO (a), OPERAÇÃO (b) e EMERGÊNCIA (c) do módulo CB.

O computador de bordo (CB) deve ser projetado para aceitar todas as mensagens consideradas válidas. Estas mensagens sugeridas são listadas no Apêndice E.

5.6.12 Módulo de comunicação (MC)

O módulo de comunicação (MC) proporciona uma conexão de dados entre o veículo e o mundo externo. O MC é responsável por todas as comunicações com o mundo externo. Este módulo deve coordenar todos os dados entre a plataforma eletrônica, o sistema de comunicação, e a carga paga. Assim, as tarefas deste módulo são coletar e

empacotar dados relevantes para distribuição para a carga paga e comunicação de base. O módulo MC é o módulo responsável pelo envio de dados de telemetria de forma a permitir um acompanhamento do veículo externamente.

A Figura 61 mostra um diagrama de blocos proposto para o módulo MC e a Figura 62 mostra um fluxograma simplificado do software proposto para o controlador do módulo MC.

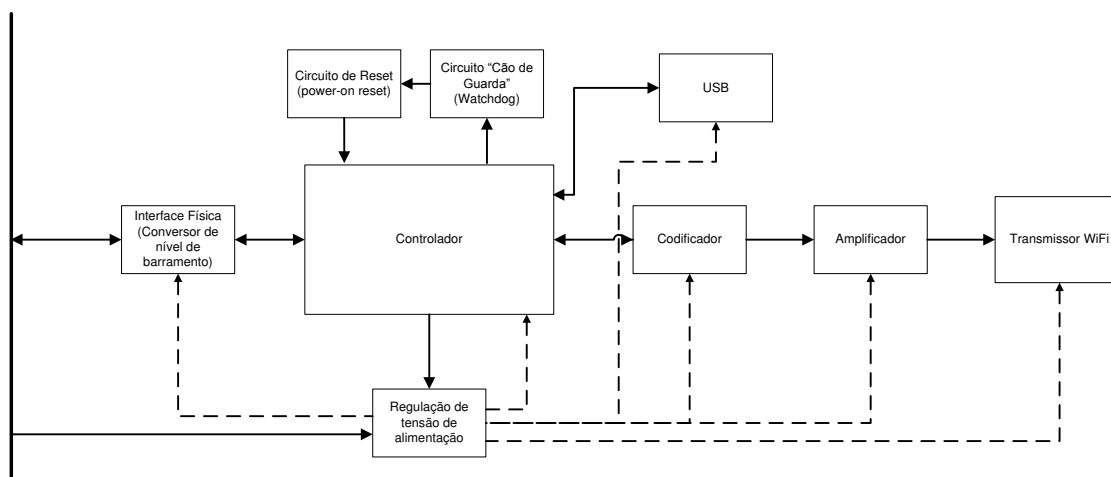


Figura 61 – Diagrama de exemplo do circuito eletrônico do módulo de comunicação (MC).

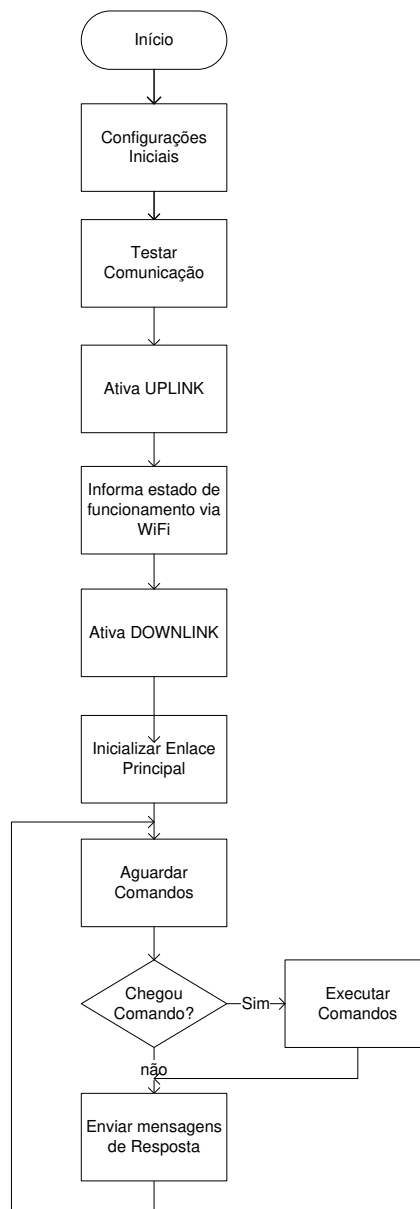


Figura 62 – Fluxograma exemplo do software recomendado para o módulo MC.

O módulo MC deve ser capaz de monitorar todas as mensagens do barramento CAN de forma a executar sua função de comunicação com a estação de solo e com a carga-paga. O MC aceita e processa todas as mensagens de todos os subsistemas.

O módulo MC gera somente uma mensagem, MC.ESTADO, pois seu funcionamento transparente não necessita de outros tipos de mensagens. A mensagem de estado MC.ESTADO é proveniente do módulo MC e mostra o estado atual do MC.

5.6.13 Módulo de emergência (ME)

O módulo de emergência é um módulo redundante. Sua importância só é percebida se considerarmos uma falha geral no sistema. Assim, este módulo deve poder acionar os motores/atuadores através de comandos recebidos pelo barramento e por uma interface própria diferente do barramento de comunicação como, por exemplo, pode-se utilizar uma antena própria operando na frequência de 146MHz e utilizando um rádio amador como transmissor de comandos DTMF. Desta forma, os comandos de emergência enviados para o ME são tons DTMF que são identificados e, por razões de segurança, precisam ser confirmados com a entrada de uma “senha”, também em DTMF, que valida a execução do comando. (SANTOS, 2006; PORTLAND STATE UNIVERSITY, 2008)

A Figura 63 mostra um diagrama de blocos proposto para o módulo ME e a Figura 64 mostra um fluxograma simplificado do software proposto para o controlador do módulo ME.

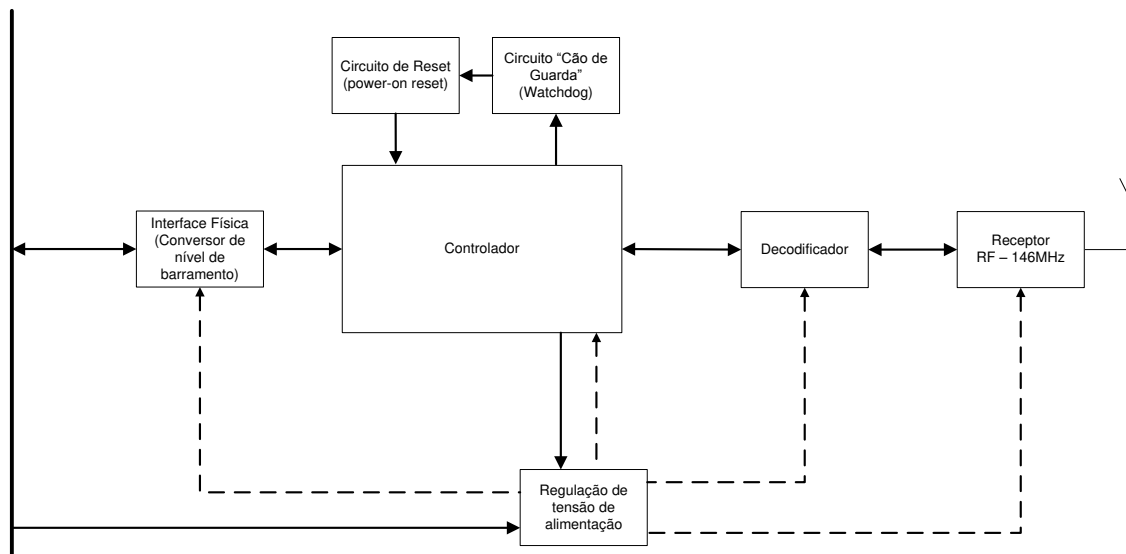


Figura 63 – Diagrama de exemplo do circuito eletrônico do módulo de emergência (ME).

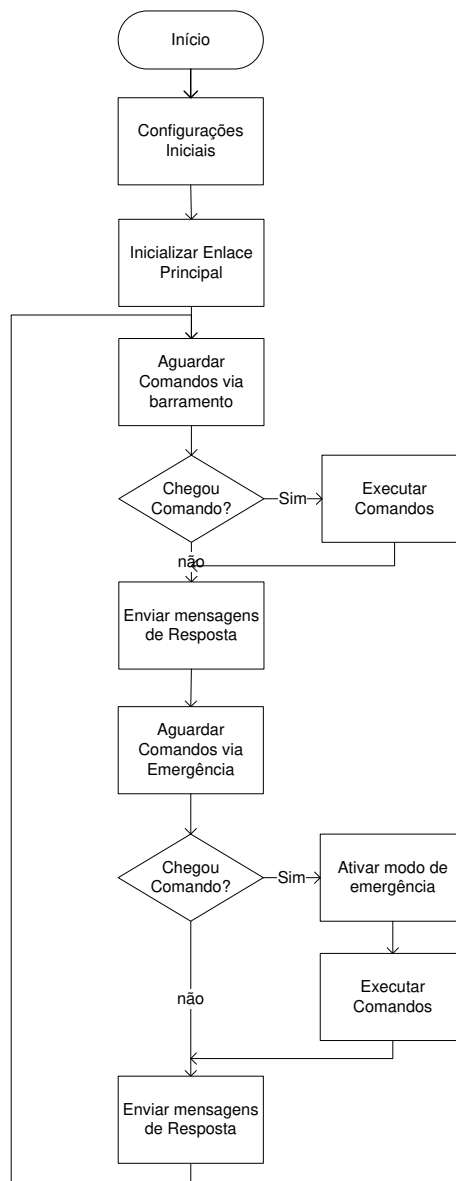


Figura 64 – Fluxograma exemplo do software recomendado para o módulo ME.

O módulo ME aceita somente mensagens diretamente enviadas à ele para a sua operação. E gera mensagens de estado e eventos para o computador de bordo. As mensagens do módulo ME são sugeridas no Apêndice E.

5.6.14 Registrador de bordo (RB)

O registrador de bordo é um dispositivo gravador de dados de estado-sólido. Ele armazena toda a informação enviada a si em uma memória não-volátil. Os dados armazenados dentro do Registrador de Bordo serão usados para análise posterior. O módulo registrador de bordo é projetado para ser uma “caixa preta”, de forma a gravar qualquer dado que trafegar pelo barramento.

Este módulo pode ser composto por somente três partes: o processador, a interface com o barramento CAN, e o banco de memória Flash. Suas funções são de capturar todas as mensagens do barramento CAN, processá-las se necessário, e escrevê-las no banco de memória Flash.

A Figura 65 mostra um diagrama de blocos proposto para o módulo RB e a Figura 66 mostra um fluxograma simplificado do software proposto para o controlador do módulo RB.

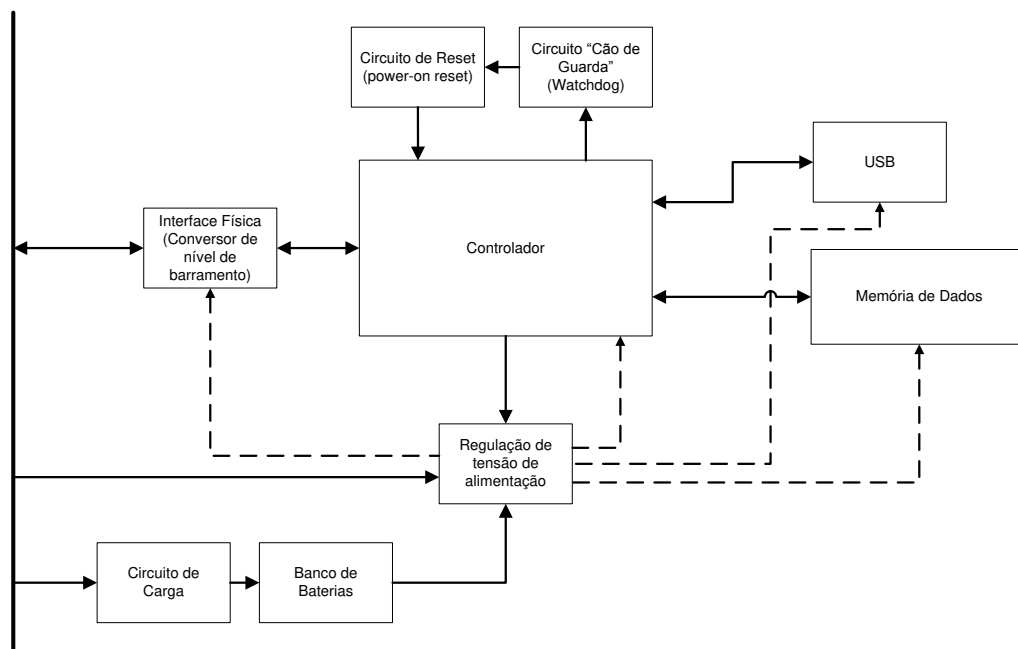


Figura 65 – Diagrama de exemplo do circuito eletrônico do registrador de bordo (RB).

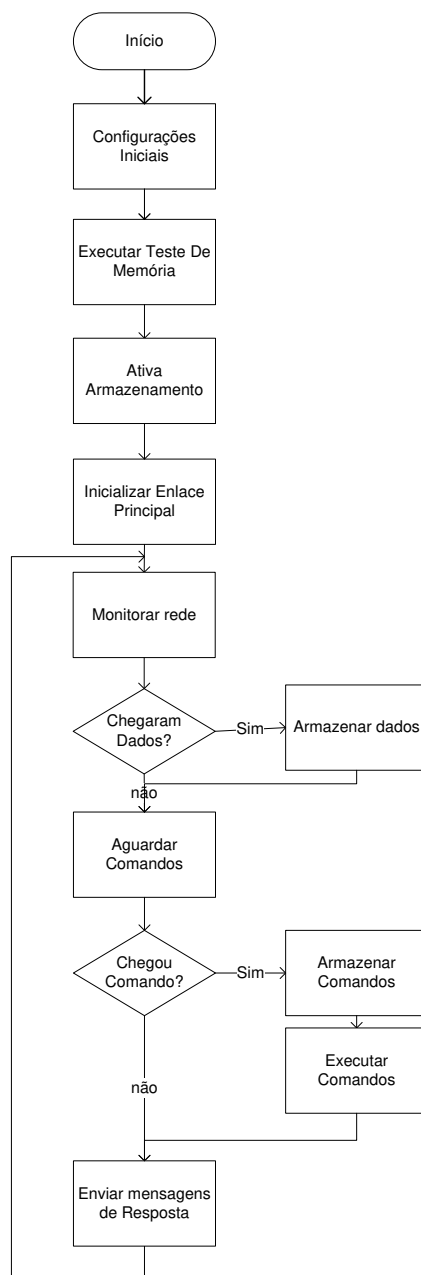


Figura 66 – Fluxograma exemplo do software recomendado para o módulo RB.

O módulo RB deve ser projetado para aceitar e gravar todas as mensagens válidas de forma a executar as funções de gravação de todas as mensagens do barramento.

Além disto, o RB pode receber comandos específicos vindo do CB ou do ME. As mensagens do módulo RB são sugeridas no Apêndice E.

5.6.15 Cabo de Depuração (CD)

O cabo de depuração é apenas um cabo que se conecta ao veículo utilizando o mesmo barramento da rede interna do sistema. Assim, ele pode acessar todos os módulos individualmente, inclusive o Computador de Bordo de forma a coletar os dados recebidos, verificar o funcionamento e até emular o funcionamento do sistema através de “mensagens teste”, ou seja, mensagens simulando o funcionamento ou mal funcionamento das partes do sistema (PORTLAND STATE UNIVERSITY, 2008). Tal cabo deve possuir os mesmos contatos do barramento de comunicação e os contatos do barramento elétrico tomando-se especial a potência necessária para carregar os módulos do veículo. Um exemplo deste tipo de cabo pode ser encontrado no projeto de Santos (2006), conforme mostra a Figura 67.



DB-9 (Macho) D-Sub

Pino	Sinais	Descrição
1	GND	Terra Adicional
2	CAN_L	Nível Baixo dominante
3	CAN_GND	Terra
4	V+	Alimentação Adicional
5	CAN_SHLD	Blindagem
6	GND	Terra
7	CAN_H	Nível Alto dominante
8	GND	Terra Adicional
9	CAN_V+	Alimentação

Figura 67 – Diagrama da montagem do cabo de depuração para o projeto do foguete Sonda-L.

Fonte: (SANTOS, 2006)

5.7 Discussão / Comentários

O projeto do sistema autônomo completo pode ser executado através de diversas formas. Essencialmente a execução se refere aos desenvolvimentos dos módulos referentes às partes do veículo.

Nossa proposta de desenvolvimento a seguir se refere ao uso da arquitetura modular de modo efetivamente incremental durante o desenvolvimento do projeto.

Assim, para o desenvolvimento do sistema autônomo completo recomenda-se seguir os seguintes passos:

1. Escolher qual plataforma mecânica deverá ser utilizada;
2. Escolher funcionalidades e especificar os módulos que serão desenvolvidos;
3. Escrever as mensagens que serão trocadas pelo barramento central;
4. Implementar cada módulo referente as funcionalidades desejadas;
5. Integrar os módulos no sistema.

O desenvolvimento de cada módulo no passo 4 deve ser interpretado como o único módulo que o computador de bordo teria para uso. Desta forma, o grau de dependência requisitado pelo projetista será tratado somente em nível de decisão no software do computador de bordo. O desenvolvimento destes módulos no passo 4 pode ser tratado com 8 fases.

5.7.1 Fase 1 – Robô Comandado

O fluxo de desenvolvimento começa com a Fase 1 com o desenvolvimento da mecânica do veículo e de seu modelo cinemático. Nesta fase deve ser preparado o módulo de atuação e controle (MAC) de forma a funcionar sem o computador de bordo.

Além do MAC, nesta fase se desenvolve o primeiro módulo de alimentação (MA) que será a fonte para o MAC e para outros módulos subsequentes. Como o MA será conectado no barramento elétrico de alimentação, pode-se implementar um módulo de fácil

replicação e que ao passo que o projeto for crescendo outros módulos de alimentação vão sendo adicionados de forma a aumentar a tempo de operação do sistema.

Desta forma, teremos somente um robô que pode ser comandado externamente.

5.7.2 Fase 2 – Robô Autônomo Simples

Com a fase 2 serão adicionados os módulos de sensores de proximidade (SP) e o módulo de cálculo odométrico (CO). Estes dois módulos são de extrema importância para um primeiro trabalho com conceitos de autonomia. O módulo CO deve conter todas as análises cinemáticas dos dados dos sensores de forma a possibilitar a estimativa de posicionamento do veículo.

Desta forma, nesta fase temos um veículo com a mecânica definida e com o controle eletrônico dela. Ainda, teremos sensores de proximidade (SP) e de deslocamento (CO) que se forem interligados diretamente ao MAC pode permitir o estudo dos casos mais simples de autonomia estudados como “veículos de Braitenberg”. Ou seja, um robô autônomo simples.

5.7.3 Fase 3 – Robô Autônomo Simples

Nesta fase deve-se adicionar o computador de bordo (CB) ao sistema. Na inserção do CB, os módulos previamente preparados (MAC, SP, MA e CO) podem necessitar de ajustes para funcionarem corretamente.

Porém, nesta fase teremos a possibilidade de implementação de algoritmos decisórios. Ou seja, estudos de um robô autônomo simples.

5.7.4 Fase 4 – Robô Autônomo com Redução de Erros de Estimativa

Nesta fase serão adicionados os módulos de referência absoluta externa. Ou seja, serão adicionados os módulos de triangulação/trilateração (TT) e o módulo de localização por marcas (LM).

A adição destes módulos possibilitará uma correção dos erros a cada referência encontrada no sistema. E desta forma se torna possível a redução de erros de estimativa.

5.7.5 Fase 5 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias

Nesta fase será adicionado o módulo de registro de mapas (RM) que tornará possível o melhor estudo das trajetórias em ambientes previamente conhecidos ou previamente percorridos.

5.7.6 Fase 6 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas

Nesta fase será inserido o módulo de visão computacional (VC) onde se torna possível um melhor estudo de trajetórias. Com a adição do módulo VC é possível a utilização de algoritmos de trajetórias para ambientes desconhecidos.

Junto com o módulo RM, o módulo VC torna o robô desenvolvido em um robô autônomo com redução de erros de estimativa e otimização de trajetórias complexas.

5.7.7 Fase 7 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas em Ambientes Externos

Nesta fase serão adicionados os módulos de dados GPS (MDGPS) e a unidade de medição inercial (UMI). Juntos, estes dois módulos potencializam a aplicação do robô autônomo possibilitando o deslocamento em ambientes externos. Para isto, novos algoritmos devem ser implementados no computador de bordo de forma a mudar as prioridades de acesso das informações.

5.7.8 Fase 8 – Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas em Ambientes Externos com Acompanhamento Externo e Controle de Emergências

Esta fase pode ser tida como opcional. Nela podem-se inserir os módulos: de comunicação (MC) que possibilitará um acompanhamento externo do tráfego no

barramento; de emergência (ME) que funciona como um computador de bordo reserva, e deverá atuar somente em caso de falha do CB; de registro de bordo (RB) que registrará todas as mensagens trocadas para análises futuras em caso de falhas; e o cabo de depuração (CD) que possibilitará o acesso ao barramento para programação e/ou carga do(s) módulo(s) de alimentação (MA).

A Figura 68 sintetiza os resultados para cada fase de implementação de um robô móvel. De forma a se atingir um “Robô Autônomo com Redução de Erros de Estimativa e Otimização de Trajetórias Complexas em Ambientes Externos com Acompanhamento Externo e Controle de Emergências”.

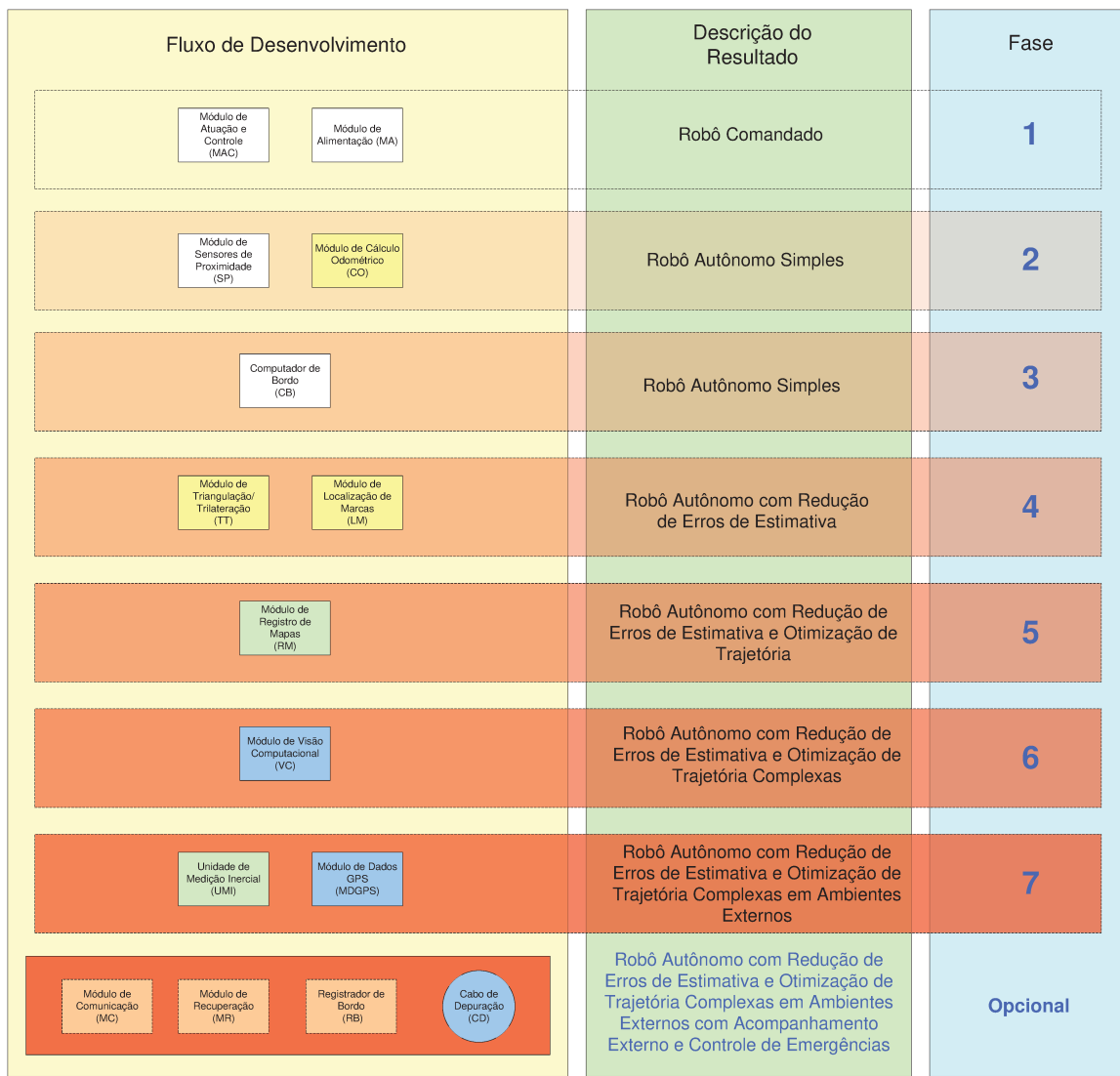


Figura 68 - Fases da implementação do desenvolvimento de um projeto de um veículo autônomo.

6 Estudo de Casos

Para cada sistema robótico possível, são desejados diferentes níveis de autonomia. Para alguns, pode ser necessária apenas um controle melhor no seu posicionamento, para outros, pode ser necessário somente o traçado de uma trajetória ótima, ou ainda um posicionamento com traçado de trajetória. Porém, ao dividirem-se estes níveis de autonomia em classes, tem-se o que se pode chamar de nível mínimo necessário de autonomia para cada uma destas classes. Este capítulo tenta explorar este aspecto dos sistemas autônomos.

Assim, quando se deseja autonomia, é possível pensar em arquiteturas mínimas para cada tipo específico de aplicação. Em teoria, podemos utilizar os mesmos módulos em vários tipos de robôs móveis. Assim, utilizando-se a mesma arquitetura é possível desenvolver diversos experimentos com diversas possibilidades de aplicação.

Para facilitar, foram divididas em duas categorias principais: bidimensional e tridimensional. Porém, esta divisão nada tem a ver com o número de variáveis presentes no modelo cinemático do veículo. Esta classificação se refere ao espaço disponível para movimentação do veículo. Ou seja, se este se movimenta no plano ou no espaço.

6.1 Arquitetura geral mínima bidimensional

Pensando-se somente em deslocamentos no plano, ou seja, em duas dimensões, e através de uma arquitetura geral mínima bidimensional (2D) é possível construir qualquer dispositivo robótico autônomo que se movimenta em duas dimensões. Uma proposta para uma arquitetura geral mínima 2D é mostrada na Figura 69.

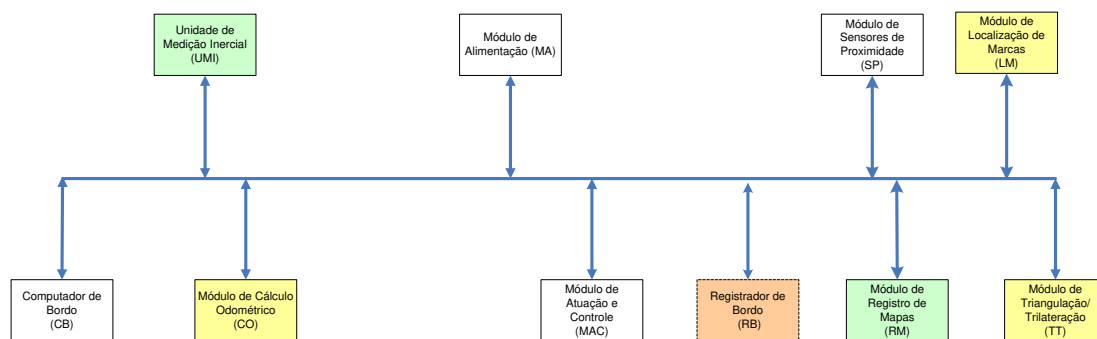


Figura 69 – Arquitetura geral mínima 2D.

Nesta proposta de arquitetura geral mínima bidimensional temos uma simplificação do esquema da Figura 36 utilizando-se somente os módulos essenciais para uma navegação com um veículo robótico móvel em duas dimensões.

6.1.1 Estudo de casos

6.1.1.1 Carro Ackerman com movimentação indoor

A configuração conhecida como Ackerman é a configuração mecânica que vemos nos veículos automotores comuns onde o eixo frontal é utilizado para fazer as curvas enquanto que o eixo traseiro permanece em linha reta. Esta configuração é mostrada na Figura 70.

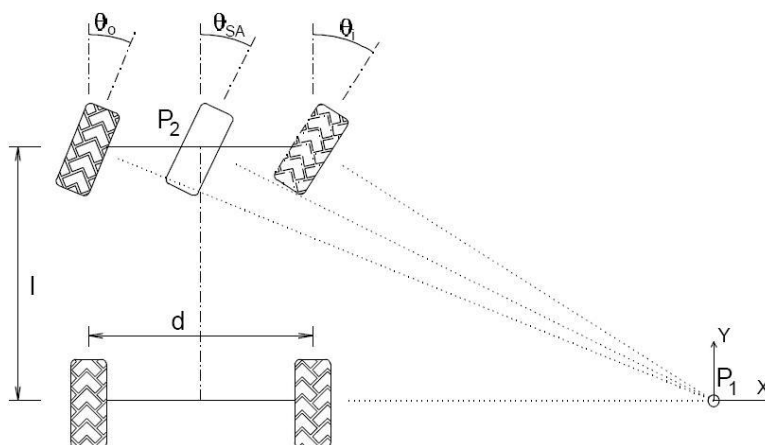


Figura 70 – Configuração Ackerman.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

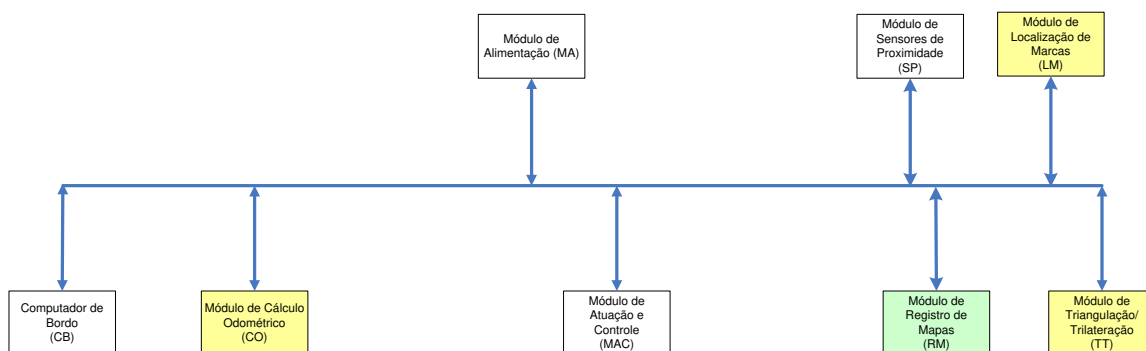


Figura 71 – Diagrama da arquitetura proposta para veículo com configuração Ackerman de rodas para movimentação indoor.

Onde se pode ver na Figura 71 que o módulo de cálculo odométrico (CO) tratará de todos os cálculos referentes ao modelo cinemático do carro Ackermann, estes cálculos incluem cálculos estatísticos de probabilidade, se este modelo for escolhido. O módulo de atuação e controle (MAC) tratará de todos os acionamentos referentes a mecânica do veículo. Ou seja, no caso do carro Ackermann, será necessário o controle da ponte H do acionamento do motor de tração e os dados de entrada do servo-mecanismo que controla as rodas de direção do veículo.

O módulo de sensores de proximidade (SP) atuará verificando possíveis problemas no caminho do veículo. Ou seja, neste módulo podem estar conectados os sensores de impacto (bumpers) e/ou sensores de distância de ultrassom/laser e todo o processamento destes dados. Assim, caso o módulo SP verifique algum objeto próximo, este deverá informar ao computador de bordo através das mensagens específicas para que este comande uma mudança de trajetória.

O módulo de localização por marcas (LM) se fará necessário com o uso de marcas naturais e atuará em conjunto com o módulo de triangulação/trilateração (TT) trocando mensagens para a geração de mapas com o módulo de registro de mapas (RM).

Todos estes módulos serão alimentados pelo módulo de alimentação (MA) e enviarão as informações para o computador de bordo (CB) que possui a função de gerenciar as decisões de forma a atingir o sucesso da missão, qualquer que seja ela.

6.1.1.2 Robô “tanque de guerra” para terrenos off-road

Uma configuração mecânica muito comum em veículos autônomos utilizados na indústria ou para o entretenimento é a configuração com uso de esteiras para movimentação. Esta configuração, frequentemente chamada de “tanque de guerra” devido a sua semelhança mecânica com os veículos blindados utilizados em guerras é bastante útil quando o veículo se destina ao uso off-road. Pois, nesta configuração mecânica é mais difícil ocorrer erros não-sistemáticos, como escorregamento das rodas, devido a sua grande área de contato com o solo. A Figura 72 mostra um veículo com esta configuração mecânica.

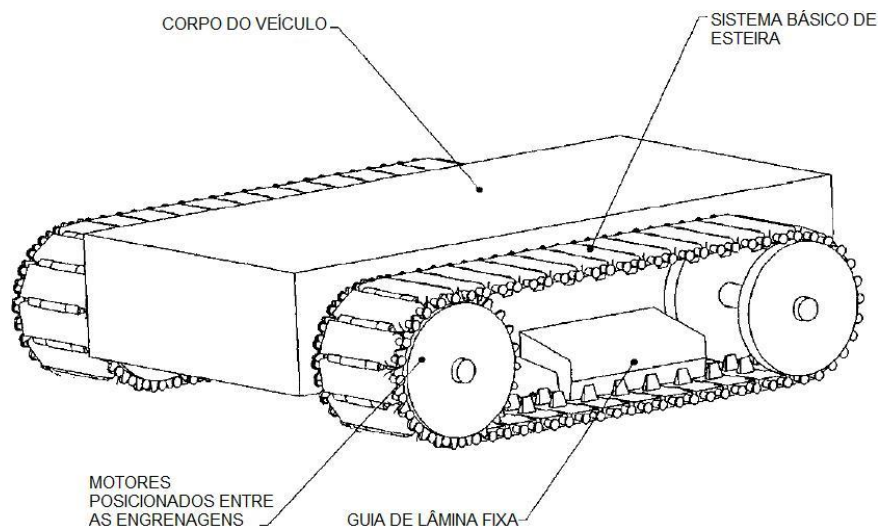


Figura 72 – Configuração mecânica com esteira.

Fonte: (SANDIN, 2003).

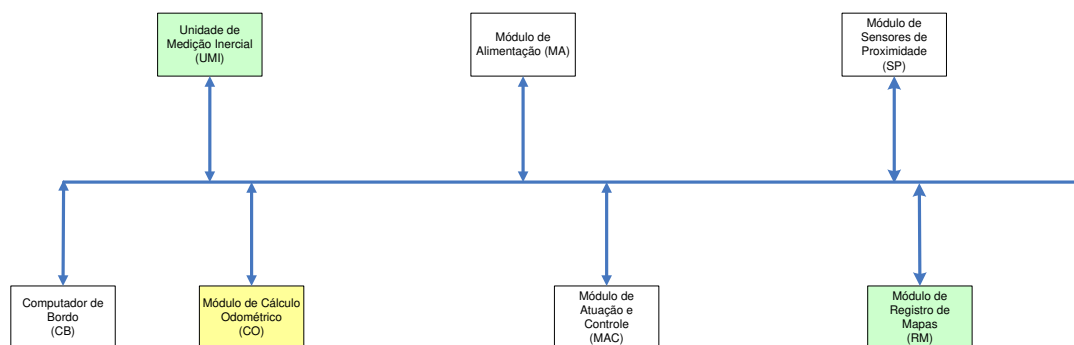


Figura 73 – Diagrama da arquitetura proposta para veículo com configuração mecânica em esteira tipo “tanque de guerra” para terrenos off-road.

Onde podemos ver na Figura 73 que o módulo de cálculo odométrico (CO) tratará de todos os cálculos referentes ao modelo cinemático do robô tanque, estes cálculos incluem cálculos estatísticos de probabilidade, se este modelo for escolhido. O módulo de atuação e controle (MAC) tratará de todos os acionamentos referentes à mecânica do veículo. Ou seja, no caso do robô tanque, será necessário o controle das pontes H dos acionamentos independentes dos motores de tração das esteiras do veículo.

O módulo de sensores de proximidade (SP) atuará verificando possíveis problemas no caminho do veículo. Ou seja, neste módulo podem estar conectados os sensores de impacto (bumpers) e/ou sensores de distância de ultrassom/laser e todo o processamento destes dados da mesma forma do carro Ackermann do exemplo anterior. Assim, caso o módulo SP verifique algum objeto próximo, este deverá informar ao computador de bordo através das mensagens específicas para que este comande uma mudança de trajetória.

A unidade de medição inercial (UMI) atuará provendo referências sobre a atitude do veículo para o sistema, trocando mensagens para a geração de mapas com o módulo de registro de mapas (RM).

Todos estes módulos serão alimentados pelo módulo de alimentação (MA) e enviarão as informações para o computador de bordo (CB) que possui a função de gerenciar as decisões de forma a atingir o sucesso da missão, qualquer que seja ela.

6.2 Arquitetura geral mínima tridimensional

Semelhantemente a arquitetura geral mínima bidimensional, a arquitetura geral mínima tridimensional (3D) representa uma arquitetura onde é possível construir um dispositivo robótico autônomo que se movimenta nos três eixos. Uma proposta de arquitetura geral mínima 3D é mostrada na Figura 74.

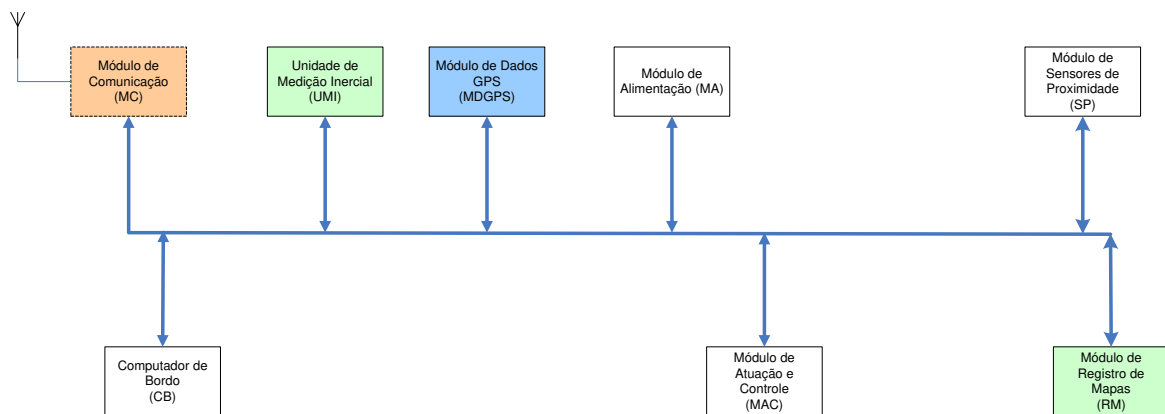


Figura 74 – Arquitetura geral mínima 3D.

Nesta proposta de arquitetura geral mínima tridimensional temos uma simplificação do esquema da Figura 36 utilizando-se somente os módulos essenciais para uma navegação com um veículo robótico móvel em três dimensões.

6.2.1 Estudo de casos

6.2.1.1 Dirigível de hélio para ambientes indoor

Mais leve que o ar, este veículo é frequentemente utilizado para entretenimento carregando câmeras ou mostrando mensagens. Seu controle é relativamente fácil quando se trata de movimentos dentro de ambientes fechados, ou seja, sem a interferência do vento ou de outros objetos voadores. A Figura 75 mostra algumas fotos deste tipo de veículo.



Figura 75 – Configuração mecânica de um dirigível de hélio para ambientes indoor.

Fonte: (DRAGANFLY INNOVATIONS INC., 2012)

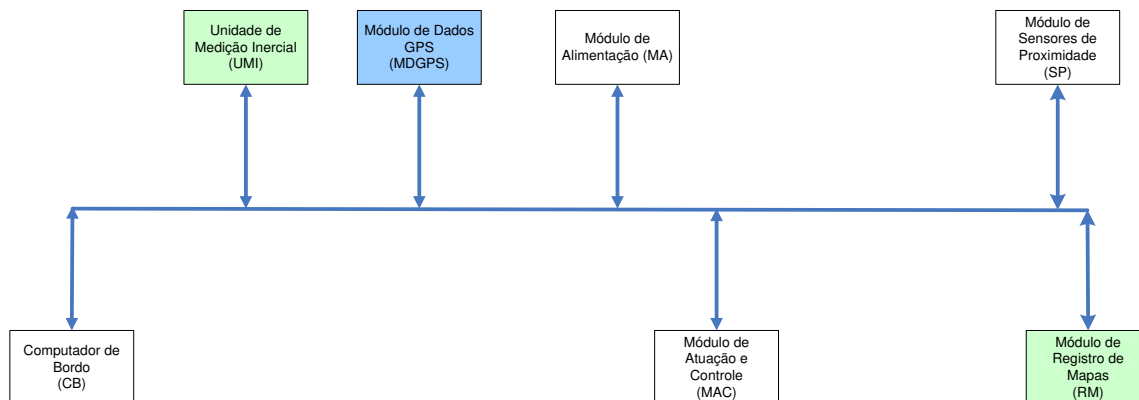


Figura 76 – Diagrama da arquitetura proposta para um dirigível de hélio autônomo se movendo no espaço livre.

Onde podemos ver na Figura 76 que o módulo de atuação e controle (MAC) é responsável pelos acionamentos mecânicos do veículo. Assim, este módulo é responsável pelo controle dos três circuitos de ponte H para acionamento dos motores propulsores do dirigível.

O módulo de sensores de proximidade (SP) atuará verificando possíveis problemas no caminho do veículo. Ou seja, neste módulo podem estar conectados sensores de distância de ultrassom/laser e todo o processamento destes dados. Assim, caso o módulo SP verifique algum objeto próximo, este deverá informar ao computador de bordo através das mensagens específicas para que este comande uma mudança de trajetória.

A unidade de medição inercial (UMI) atuará provendo referências sobre a atitude do veículo para o sistema, trocando mensagens para a geração de mapas com o módulo de registro de mapas (RM). A UMI, como o módulo de cálculo odométrico necessita de processamento próprio enviando somente mensagens já processadas para o computador de bordo. O módulo dados GPS (MDGPS) atua como referência absoluta do módulo UMI.

Todos estes módulos serão alimentados pelo módulo de alimentação (MA) e enviarão as informações para o computador de bordo (CB) que possui a função de gerenciar as decisões de forma a atingir o sucesso da missão, qualquer que seja ela.

6.2.1.2 Aeronave não tripulada

Extremamente popular, as aeronaves não tripuladas tem crescido em número e aplicações. Seu uso varia desde monitoramento agrícola até aplicações militares. Seu controle é mais difícil devido a estar em ambientes abertos e sujeitos a todas e quaisquer interferências naturais. A Figura 77 mostra duas configurações mecânicas deste tipo de veículo.

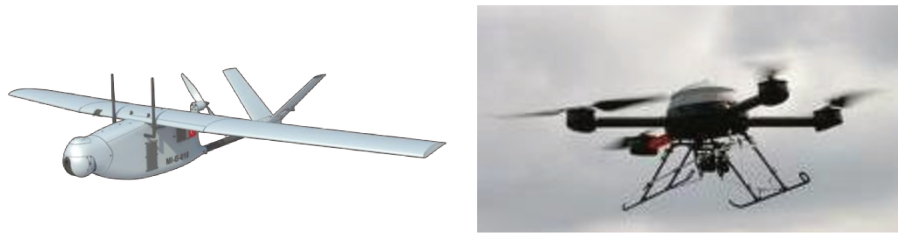


Figura 77 – Propostas de configurações mecânicas de aeronaves não tripuladas existentes no mercado.

Fonte: (BAYKAR MAKINA, 2013; SEA HAWK MARITIME LTD., 2013)

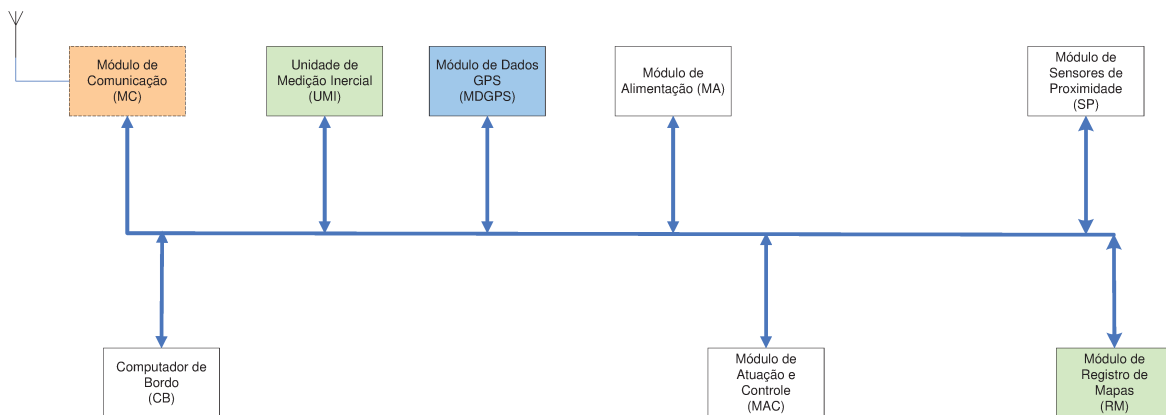


Figura 78 – Diagrama da arquitetura proposta para uma aeronave não tripulada autônoma.

Onde podemos ver na Figura 78 que o módulo de atuação e controle (MAC) é responsável pelos acionamentos mecânicos do veículo. A unidade de medição inercial (UMI) atuará provendo referências sobre a atitude do veículo para o sistema e junto com o

módulo de dados GPS (MDGPS) torna possível a localização no espaço do veículo. Estes dois módulos trocam mensagens para a geração de mapas com o módulo de registro de mapas (RM). A UMI, como o módulo de cálculo odométrico necessita de processamento próprio enviando somente mensagens já processadas para o computador de bordo.

O módulo SP inclui todos os sensores gerais para detecção de iluminação, altitude, proximidade dentre outros.

Através do módulo de comunicação (MC) é possível diagnosticar ou inserir informações/missões para o computador de bordo.

Todos estes módulos serão alimentados pelo módulo de alimentação (MA) e enviarão as informações para o computador de bordo (CB) que possui a função de gerenciar as decisões de forma a atingir o sucesso da missão, qualquer que seja ela.

7 Conclusão e Trabalhos Futuros

7.1 Conclusão

Com este trabalho foi possível estudar conceitos de autonomia, métodos de localização, guiagem e arquiteturas modulares existentes no mercado industrial e automotivo para uso em robótica móvel. Com isto foi possível obter uma arquitetura de hardware escalável para veículos móveis de forma a se atingir comportamento autônomo. Ou seja, como uma plataforma de estudo de conceitos de autonomia.

Baseado nas faculdades mentais como modelo foi proposto um “sistema autônomo completo” que suporta diversos tipos de aplicações mecânicas para a mesma plataforma eletrônica. Ainda, foram propostas arquiteturas de hardware simplificadas baseadas no “sistema autônomo completo”. E por fim um método de como alcançar o desenvolvimento das arquiteturas de hardware propostas.

7.2 Trabalhos Futuros

Durante as fases de pesquisa deste trabalho, o autor percebeu algumas possibilidades de trabalhos futuros que são comentadas a seguir:

1. Módulos complementares: Com uma arquitetura modular, alguns módulos podem ser integrados de forma complementar. Tais módulos atuando-se em conjunto já possibilitam características de autonomia no sistema.
2. Arco-Reflexo: Assim como o “arco-reflexo” do corpo humano, é possível estudarmos este conceito no campo da robótica. Este conceito pode ser implementado como uma mensagem de alta prioridade no barramento, ou mesmo uma conexão direta entre dois módulos de forma a permitir reações mais rápidas do veículo por razões de segurança.

3. Estudos de competência: Contudo, um dos principais objetivos deste trabalho foi o de iniciar uma série de estudos com desenvolvimento de competências dentro da universidade. Assim, cada módulo citado nos capítulos anteriores pode ser desenvolvido por pessoas diferentes em diferentes projetos de forma a agregar diferentes experiências com veículos móveis e por fim estudar conceitos de autonomia.

Explorando um pouco mais a terceira sugestão foi sintetizada uma proposta de possíveis trabalhos a serem executados na Tabela 10 onde são sugeridos o tipo de trabalho (iniciação científica, mestrado ou doutorado) de acordo com a complexidade e a duração estimada do projeto.

Tabela 10 – Sugestão de trabalhos científicos para desenvolvimento de conceitos de autonomia.

Fase	Módulo	Complexidade	Tempo de Desenvolvimento	Tipo de Trabalho
1	Módulo de Atuação e Controle	Média	1 ano	Mestrado
	Módulo de Alimentação	Baixa	6 meses	Iniciação Científica
2	Módulo de Sensores de Proximidade	Baixa	6 meses	Iniciação Científica
	Módulo de Cálculo Odométrico	Média	1 ano e 6 meses	Mestrado
3	Computador de Bordo	Alta	2 anos	Mestrado/Doutorado
4	Módulo de Triangulação/Trilateração	Média	1 ano	Mestrado
	Módulo de Localização de Marcas	Média	1 ano	Mestrado

Fase	Módulo	Complexidade	Tempo de Desenvolvimento	Tipo de Trabalho
5	Módulo de Registro de Mapas	Alta	2 anos	Mestrado/Doutorado
6	Módulo de Visão Computacional	Muito Alta	3 anos	Doutorado
7	Unidade de Medição Inercial	Alta	2 anos	Mestrado/Doutorado
	Módulo de Dados GPS	Alta	1 ano e 6 meses	Mestrado
8	Módulo de Comunicação	Média	1 ano	Mestrado
	Módulo de Recuperação	Média	1 ano	Mestrado
	Registrador de Bordo	Baixa	6 meses	Iniciação Científica
	Cordão Umbilical	Baixa	6 meses	Iniciação Científica

Por fim, caso seja feito este trabalho futuro, ao concluir esta sequência de projetos será possível vislumbrar diferentes camadas do desenvolvimento de sistemas autônomos com possibilidades de pesquisa em diversas áreas da engenharia.

Referências

1911 ENCYCLOPAEDIA BRITANNICA. Autonomy. **Wikisource**, 2012. Disponível em: <http://en.wikisource.org/wiki/1911_Encyclop%C3%A6dia_Britannica/Autonomy>. Acesso em: 04 fev. 2013.

AGUIAR, L. M. F.; VASSALLO, R. F. **Navegação de um Robô a Partir de um Mapa do Ambiente Utilizando Visão Omnidirecional**. X SBAI - Simpósio Brasileiro de Automação Inteligente. São João del Rei - MG - Brasil: [s.n.]. 2011. p. 689-694.

ALMONFREY, D. et al. **Uso da Constância de Cor na Robótica Móvel**. X SBAI - Simpósio Brasileiro de Automação Inteligente. São João del Rei, MG, Brasil: [s.n.]. 2011. p. 1256-1261.

ARKIN, R. C. **Behavior-Based Robotics**. Cambridge, Massachussets: The MIT Press, 1998.

ARRUDA, T. A. **Arquitetura de Hardware e Software para Supervisão e Controle de um Carro Autônomo**. Universidade Federal de Minas Gerais. Departamento de Engenharia Elétrica. Laboratório de Sistemas Computacionais e Robótica. Grupo de Pesquisa e Desenvolvimento de Veículos Autônomos. Belo Horizonte. 2012.

AUGUSTO, S. R. **Uma plataforma móvel para estudos de autonomia**. Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Telecomunicações e Controle. São Paulo. 2007.

BAYKAR MAKINA. Bayraktar Mini UAV. **Bayraktar**, 2013. Disponível em: <<http://www.baykarmakina.com/en/MiniUAV>>. Acesso em: 09 mar. 2013.

BECKE, G. et al. Comparing Bus Solutions. **High-Performance Linear/Interface. Application Report. Texas Instruments Inc.**, 2000.

BEKIR, E. **Introduction to Modern Navigation Systems**. 1st. ed. Singapore: World Scientific Publishing Co. Pte. Ltd., 2007.

BORENSTEIN, J.; EVERETT, H. R.; FENG, L. **Where am I? Sensors and Methods for Mobile Robot Positioning**. [S.l.]: University of Michigan, 1996. Oak Ridge National Lab (ORNL). D&D Program. The United States Department of Energy's Robotics Technology Development Program.

BOSCH. **CAN Specification version 2.0**. Robert Bosch GmbH. Stuttgart. 1991.

BRAITENBERG, V. **Vehicles: Experiments in Synthetic Psychology**. [S.l.]: Bradford, 1986.

CAN IN AUTOMATION GMBH. **CANopen - CiA DS-102 v.2.0 - CAN Physical Layer for Industrial Applications**. CAN in Automation GmbH. Erlangen, Germany. 1994.

CAN IN AUTOMATION GMBH. **CiA - CAN Specification 2.0, part A**. CAN in Automation GmbH. Erlangen, Germany. 1998.

CAN IN AUTOMATION GMBH. **CiA - CAN Specification 2.0, part B**. CAN in Automation GmbH. Erlangen, Germany. 1998.

CANopen - CiA DS-301 v.4.02 - Application Layer and Communication Profile. CAN in Automation GmbH. [S.l.]. 2002.

CANOPEN - the standardized embedded network. **CAN in Automation GmbH**. Disponível em: <<http://www.can-cia.org>>.

COHEN, C.; KOSS, F. V. **A Comprehensive Study of Three Object Triangulation**. 1993 SPIE Conference on Mobile Robots. Boston, MA: [s.n.]. 1992. p. 95-106.

COMPARISON of WLAN, WPAN and WMAN Technologies. **Network Dictionary**. Disponível em: <<http://www.networkdictionary.com/Wireless/Comparison-of-WLAN.php>>.

CUVILLIER, A. **ABC da Psicologia**. São Paulo: Companhia Editora Nacional, 1937.

DALGALARRONDO, P. **Psicopatologia e Semiologia dos Transtornos Mentais**. [S.l.]: Editora Artes Médicas, 2000.

DLR GERMANY AND AREXX - THE NETHERLANDS. ASURO ROBOT KIT. **AREXX ENGINEERING**, 2006. Disponível em: <http://www.arexx.com/arexx.php?cmd=goto&cparam=p_asuro>. Acesso em: 04 fev. 2013.

DOUGHERTY, E. R.; GIARDINA, C. R. **Mathematical Methods for Artificial Intelligence and Autonomous Systems**. [S.l.]: Prentice-Hall International Inc., 1988.

DRAGANFLY INNOVATIONS INC. RC Electric Model Blimps. **Draganfly - RCtoys**, 2012. Disponível em: <<http://www.rc toys.com/>>. Acesso em: 09 mar. 2013.

EMC Design Guide for Printed Circuit Boards. Engineering Specification ES-3U5T-1B257-AA. FORD Motor Company. [S.l.]. 2002.

FARREL, J.; BARTH, M. **The Global Positioning System and Inertial Navigation**. [S.l.]: McGraw-Hill Professional, 1999.

GREWAL, M. S.; WEILL, L. R.; ANDREWS, A. P. **Global Positioning Systems, Inertial Navigation, and Integration**. Second Edition. ed. [S.l.]: John Wiley & Sons Inc., 2007.

HELLES, G. ABC of Rechargeable Batteries. **Elektor**, 2004.

HELLES, G. O ABC das Pilhas Recarregáveis. **Elektor eletrônica e microinformática. Edição brasileira.**, p. 20, 42 2004. ISSN 1676-790X.

HOLLAND, J. M. **Designing Autonomous Mobile Robots: Inside the Mind of an Intelligent Machine.** [S.l.]: Newnes. Elsevier Inc., 2004.

HOLT, J. E. **How not to design an avionics system.** Published by AIAA - 3rd Responsive Space Conference. [S.l.]: [s.n.]. 2005.

INTERFACEBUS.COM. **Interface Bus.** Disponível em: <<http://www.interfacebus.com>>.

JOHNSON, K. et al. **Choosing the best real-time scheduling method for your embedded device.** Embedded.com. [S.l.]. 2006.

KENNAN, M.; MCFADDEN, M. **PSAS Avionics System Design Project.** Portland State Aerospace Society. Portland State AESS. [S.l.]. 1999.

KLEEMAN, L. **Optimal Estimation of Position and Heading for Mobile Robots Using Ultrasonic Beacons and Dead-reckoning.** Proceedings of IEEE International Conference on Robotics and Automation. Nice, France: [s.n.]. 1992. p. pp. 2582-2587. apud.

KOOKER, J. **Bluetooth, Zigbee, and Wibree: A comparison of WPAN technologies.** UCSDCSE. [S.l.]. 2008.

LAWRENCE, A. **Modern Inertial Technology: Navigation, Guidance, and Control.** [S.l.]: Springer, 1998. ISBN 0387985077.

LEONARD, J.; DURRANT-WHITE, H. F. Mobile Robot Localization by Tracking Geometric Beacons. **IEEE Transactions on Robotics and Automation**, v. 7, n. 3, p. 376-382, 1991.

LIGGINS, M. E.; HALL, D. L.; LLINAS, J. **Handbook of Multisensor Data Fusion. Theory and Practice.** Second Edition. ed. [S.l.]: CRC Press, 2009.

LUETHI, P.; MOSER, T. **Low Cost Inertial Navigation System**, 2000. Disponível em: <<http://www.trash.net/~luethi/study/ins/ins.html>>. Students project at the Electronics Laboratory of the Swiss Federal Institute of Technology. Zurich, Switzerland.

MACHADO, R. F.; ROSA, P. F. F.; DE ALMEIDA, D. F. **Um Sistema de Acompanhamento Visual para Robôs Móveis Semi-Autônomos em Ambientes Semi-Estruturados.** X SBAI - Simpósio Brasileiro de Automação Inteligente. São João del Rei - MG - Brasil: [s.n.]. 2011. p. 677-682.

MARTINS, R. J. et al. **Localização em Robótica Terrestre: Fusão entre Odometria por Múltiplos Encoders e GPS**. X SBAI - Simpósio Brasileiro de Automação Inteligente. São João del Rei - MG - Brasil: [s.n.]. 2011. p. 1268-1273.

MOREIRA, M. A. G. et al. Localização, Modelagem e Controle de um Mini-Helicóptero em Ambientes Internos. **Revista Controle & Automação / Vol.22 no.3**, Belo Horizonte, MG, Junho 2011. 238-255.

NILLSON, N. L. **Principles of Artificial Intelligence**. [S.l.]: Morgan Kaufmann, 1986.

OLIVEIRA, A. A. G. **Estudo estatístico dos processos envolvidos em uma plataforma de atitude solidária**. Instituto Nacional de Pesquisas Espaciais (INPE). [S.l.]. 2000. (INPE-7990-TDI/745). Dissertação de Mestrado em Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle.

OPERATING system - from Wikipedia, the free encyclopedia. **Wikipedia**. Disponível em: <http://en.wikipedia.org/wiki/Operating_system>.

PAIVA, B. M.; NETTO, A. F. M.; PAIVA, H. M. **Algoritmo Completo para Integração em Sistemas Inerciais Solidários (Strapdown)**. Instituto Nacional de Pesquisas Espaciais (INPE). [S.l.]. 2005. (INPE-12977-PRE/8255). Relatório final de projeto de iniciação científica (PIBIC/CNPq/INPE).

PARALLAX INC. Eddie Robot Platform. **Parallax Inc.**, 2013. Disponível em: <<http://www.parallax.com/eddie>>. Acesso em: 07 fev. 2013.

PASSERN, U. Baterias de Lítio. **Elektor eletrônica e microinformática. Edição brasileira.**, p. 32, 48 2005. ISSN 1676-790X.

PASSERN, U. Super Lithium Batteries. **Elektor**, 2005.

PATRICK, J. Serial Protocols Compared. **Embedded.com**, 2002.

PAZUL, K. **Controller Area Network (CAN) Basics. AN713**. Microchip Technology Inc. [S.l.]. 1999.

PENG, F. The myth of the "ideal bus". **Embedded.com**, 2005.

PERKINS, J. et al. **Free Software and High-Power Rocketry**: The Portland State Aerospace Society. 2003 USENIX Annual Technical Conference (USENIX'03). [S.l.]: [s.n.]. 2003. Portland State Aerospace Society. Computer Science Department. Portland State University.

PORTLAND STATE UNIVERSITY. PSAS - Portland State Aerospace Society. **Portland State University**, 2008. Disponível em: <<http://psas.pdx.edu>>. Acesso em: 06 fev. 2013.

REAL-TIME operating system - from Wikipedia, the free encyclopedia. **Wikipedia**. Disponível em: <<http://en.wikipedia.org/wiki/RTOS>>.

RICHARDS, P. A **CAN Physical Layer Discussion**. AN228. Microship Technology Inc. [S.l.]. 2002.

RICHARDS, P.; STONEKING, R. Build A Simple And Inexpensive Controller Area Network Node. **Ideas for Design. Electronic Design.**, 2001.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence a modern approach**. 3rd. ed. New Jersey: Prentice Hall, 2010.

SANDIN, P. E. **Robot Mechanisms and Mechanical Devices Illustrated**. [S.l.]: McGrall-Hill, 2003.

SANTANA, D. D. S.; FURUKAWA, C. M. **Estimação de trajetórias terrestres utilizando unidade de medição inercial de baixo custo e fusão sensorial**. Escola Politécnica da Universidade de Sao Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos. [S.l.]. 2005. Dissertação (Mestrado).

SANTOS, M. F. S. **Descrição Técnica dos Sistemas Aviônicos do Projeto SONDA-L**. Instituto de Aeronautica e Espaço. Comando da Aeronautica. Sao Jose dos Campos, SP. 2006.

SEA HAWK MARITIME LTD. Sea Hawk Maritime Security and Risk Management - UAV. **Sea Hawk Maritime**, 2013. Disponível em: <<http://www.seahawk.org.uk/uav/index.php>>. Acesso em: 09 mar. 2013.

SEALE, E. The EncycloBEAMia : BEAM. **BEAM**, 2003. Disponível em: <<http://encyclobeamia.solarbotics.net/articles/beam.html>>. Acesso em: 30 jun. 2013.

SENA ESTEVES, J.; CARVALHO, A.; COUTO, C. **Generalized Geometric Triangulation Algorithm for Mobile Robot Absolute Self-Localization**. ISIE'03. IEEE INTERNATIONAL SYMPOSIUM ON INDUSTRIAL ELECTRONICS. Rio de Janeiro: IEEE. 2003.

SIEGWART, R.; NOURBAKHSI, I. R. **Introduction to Autonomous Mobile Robots**. Cambridge: The MIT Press, 2004. A Bradford Book.

SIEGWART, R.; NOURBAKHSI, I. R.; SCARAMUZZA, D. **Introduction to Autonomous Mobile Robots**. 2nd. ed. Cambridge, Massachusetts: The MIT Press, 2011.

SILVA, D. H. D. J. **Implementação de uma rede CANopen para controle de veículos autônomos**. Universidade Federal de Minas Gerais. Escola de Engenharia. Monografia de Graduação. Minas Gerais. 2011.

THE Teardown. **Engineering & Technology**, v. 6, n. 3, p. 94-95, 2011.

THRUN, S. Probabilistic Algorithms in Robotics, Pittsburgh, PA, April 2000. CMU-CS-00-126. Carnegie Mellon University. School of Computer Science.

THRUN, S.; BURGARD, W.; FOX, D. **Probabilistic Robotics**. Cambridge, Massachussets: The MIT Press, 2006.

TORRES, J. M. C.; HEMERLY, E. M. **Caracterização de Sensor Inercial e Aplicação em Barco Autônomo**. Instituto Tecnológico de Aeronautica (ITA). [S.l.]. 2002. Relatório final de projeto de iniciação científica (PIBIC/ITA).

VEX ROBOTICS INC. Sensor Accessories. **VEX Robotics**, 2012-2013. Disponível em: <<http://www.vexrobotics.com/vex/products/accessories/sensors>>. Acesso em: 06 mar. 2013.

WIKIPEDIA. Angulos de Euler. **Wikipedia**, 2011. Disponível em: <http://es.wikipedia.org/wiki/%C3%81ngulos_de_Euler>. Acesso em: 05 fev. 2013.

WIKIPEDIA. Autonomous Robot. **Wikipedia**, 2012. Disponível em: <http://en.wikipedia.org/wiki/Autonomous_robot>. Acesso em: 04 fev. 2013.

WIKIPEDIA. Autonomy. **Wikipedia**, 2012. Disponível em: <<http://en.wikipedia.org/wiki/autonomy>>. Acesso em: 04 fev. 2013.

WIKIPEDIA. Soporte Cardan. **Wikipedia**, 2012. Disponível em: <<http://es.wikipedia.org/wiki/Gimbal>>. Acesso em: 07 fev. 2013.

WOLF, W. H. **Computers as Components: Principles of Embedded Computing System Design**. [S.l.]: Morgan Kaufmann Publishers, 2001.

YONEYAMA, T.; NASCIMENTO JR., C. L. **Inteligência Artificial em Controle e Automação**. [S.l.]: Editora Edgard Blucher, 2000.

ZELWANGER, H. **CANopen: The Higher Layer Protocol for Embedded Networks**. CAN in Automation (CiA). [S.l.].

ZHANG, Z. Microsoft Kinect Sensor and Its Effect. **Multimedia, IEEE**, v. 19, n. 2, p. 4-10, Feb. 2012. ISSN 1070-986X.

Apêndices

Apêndice A – Erros na navegação odométrica

A.1 Erros

Na navegação odométrica temos um intenso relacionamento do movimento da roda do robô com o processo de determinação de posição. Assim, dependendo da geometria do robô, poderemos ter erros relacionados a diversas características mecânicas ou ambientais do robô.

Por exemplo, em um robô utilizando a topologia *Ackerman* (a mesma arquitetura dos automóveis comerciais) com *encoder* óptico ligado a uma de suas rodas traseiras, pode apresentar erro sistemático devido a diferenças milimétricas no diâmetro de suas rodas ou mesmo devido a deformidades milimétricas no chassi do robô causando pequenos ângulos entre os eixos das rodas; e erros não-sistemáticos ocasionados esporadicamente devido a um terreno escorregadio ou ondulado. Ambos os erros são problemas para o projetista, o erro sistemático é um problema porque sempre está adicionando erro no sistema e o erro não-sistemático devido a sua imprevisibilidade (BORENSTEIN, EVERETT e FENG, 1996, p. 130-131).

Visto isto, é razoável que muitos pesquisadores gastem muito tempo desenvolvendo algoritmos para estimar a incerteza da posição. Com esta estimativa, cada robô possui não somente uma posição, mas, uma região onde sua localização é mais provável. Assim, a posição do veículo móvel é dada por uma elipse característica (conforme mostra a Figura 79) do seu erro de posição, indicando a região de incerteza da sua atual posição.

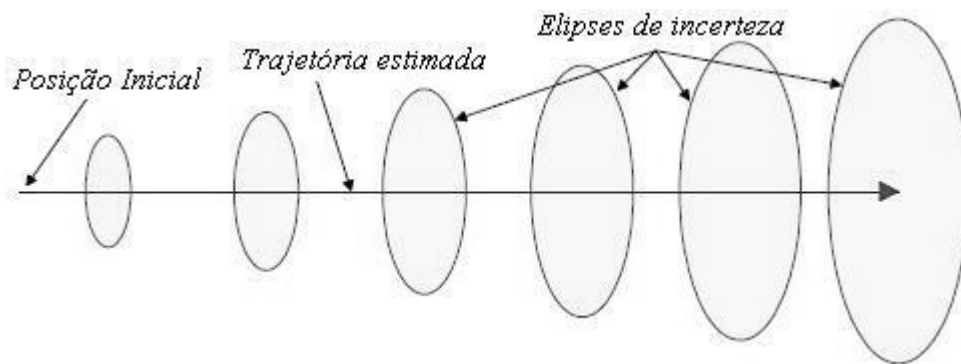


Figura 79 – Crescimento das “elipses de erros” indicando o crescimento da incerteza de posição com a odometria.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

Tipicamente, estas elipses crescem com o aumento das distâncias percorridas devido à propagação dos erros. Desta forma, devem ser combinados com o posicionamento por marcas ou mapas. Ou seja, a navegação odométrica/mapas processos de localização/navegação absoluta, tais como marcas no ambiente, de modo a reiniciar o processo de cálculo de incerteza. Quando um sistema de robótica móvel é instalado de forma híbrida com navegação odométrica e por marcas, a frequência das marcas no solo/teto é determinada empiricamente e é baseada no pior caso dos erros sistemáticos. Porém, tais sistemas não estão livres de falhas caso ocorra um erro não-sistemático maior que o esperado.

Obviamente, estes parâmetros calculados somente levam em consideração os erros sistemáticos, visto que a magnitude dos erros não-sistemáticos é imprevisível.

A.2 Redução de Erros Odométricos

A precisão da odometria depende de alguns graus de liberdade e de dimensões críticas de seu modelo cinemático. Segundo Borenstein, Everett e Feng (1996, p. 137-138), algumas considerações de projeto, obtidas empiricamente, que afetam a precisão o sistema odométrico são listadas abaixo:

- Veículos com pequena distância entre eixos são mais predispostos a erros de orientação que veículos com grande distância entre eixos;
- Veículos com rodas castor (mostrada na Figura 80), que escorregam frequentemente quando se reverte a direção de movimento, comprometem as medidas incrementais;

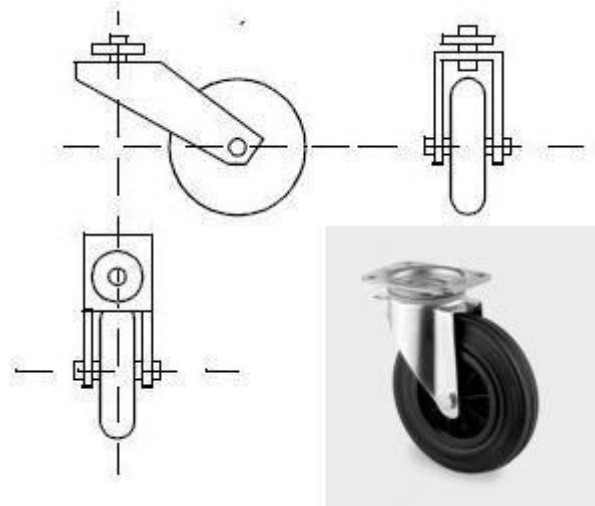


Figura 80 – Rodas castor amplamente utilizada em robótica.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

- Idealmente, as rodas utilizadas em odometria devem ser retas, finas e de material não compressível. A roda ideal deve ser feita de alumínio com uma fina camada de borracha para dar tração;
- Tipicamente, a arquitetura *Synchro-drive* (mostrada na Figura 81) proporciona uma melhor precisão que os veículos diferenciais (mostrados na Figura 82). Isto é especialmente verdadeiro se os veículos estiverem em terrenos irregulares. Mesmo quando apenas duas rodas permanecem em contato com o solo, a arquitetura *Synchro-drive* proporciona mais tração e força que outras arquiteturas. Além de informar a correta distância total percorrida, que será propriamente indicada pelo *encoder*.

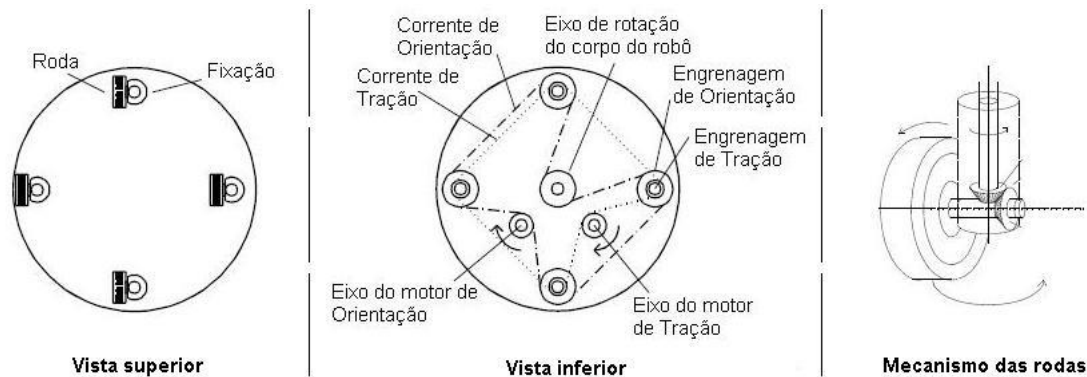


Figura 81 – Arquitetura Synchro-drive, mostrando vista superior, inferior e o mecanismo das rodas que permite o controle de direção de tração por meio de correntes.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

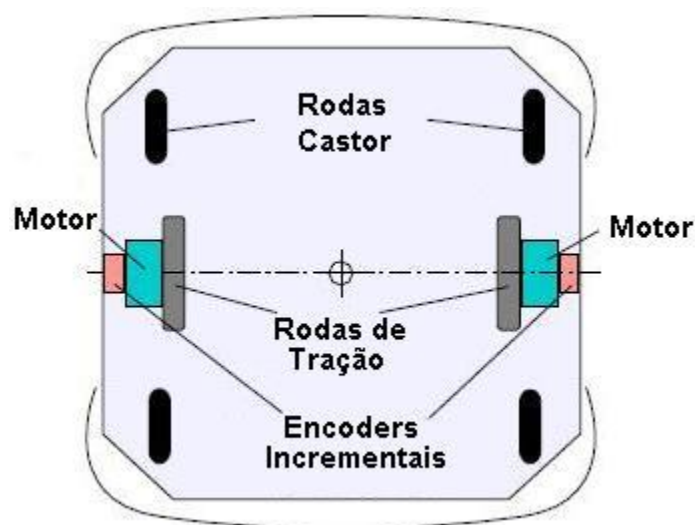


Figura 82 – Arquitetura diferencial (vista inferior).

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

Para redução de erros odométricos sistemáticos existem algumas técnicas que são descritas abaixo:

- Rodas auxiliares – normalmente é possível aumentar a precisão odométrica adicionando um par de rodas retas não tracionadas (veja a Figura 83)

fixadas em um *encoder*. Estas rodas, como não transmitem tração, podem ser finas com uma fina camada de borracha (BORENSTEIN, EVERETT e FENG, 1996).

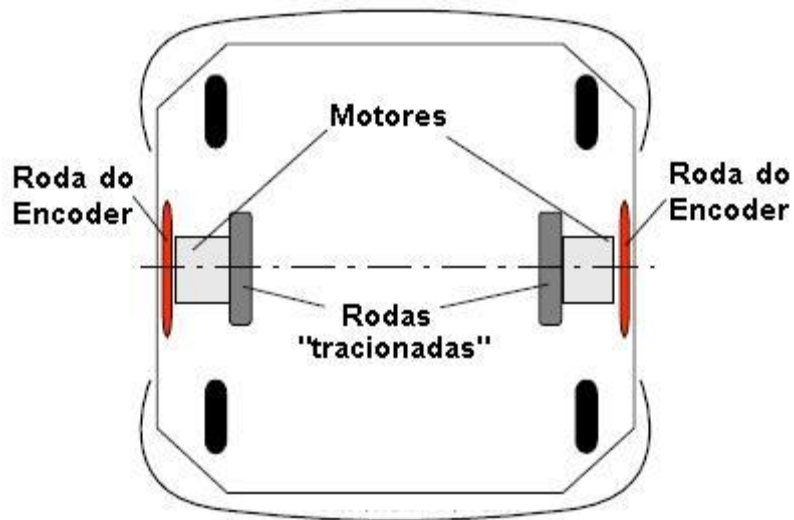


Figura 83 – Desenho conceitual de rodas de encoders para um veículo com tração diferencial.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

- Encoder trailer – uma alternativa às rodas auxiliares é levar um outro carrinho (trailer) com duas rodas ligadas a um encoder, como mostra a Figura 84 (BORENSTEIN, EVERETT e FENG, 1996).

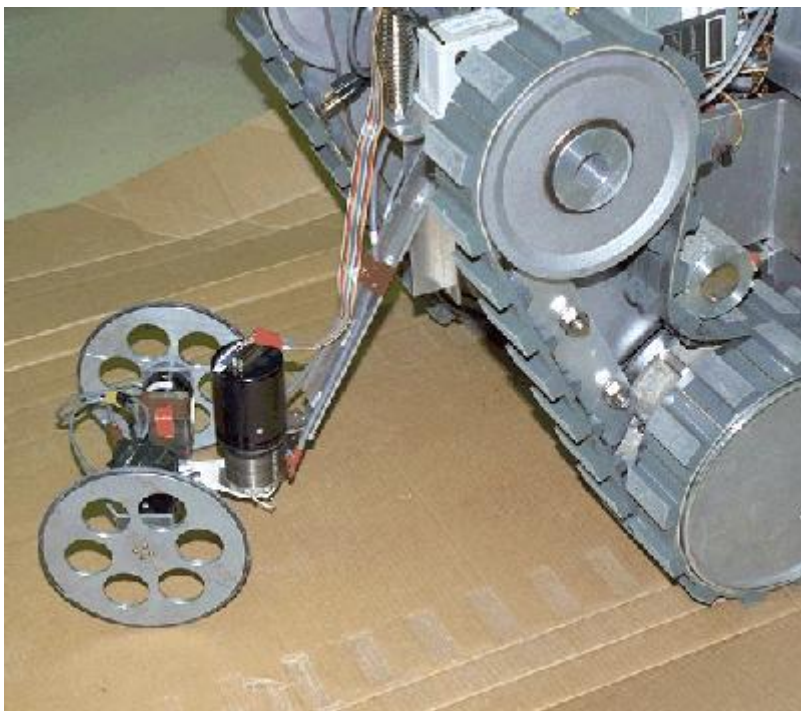


Figura 84 – Encoder trailer. O carrinho (trailer) foi projetado para aumentar a precisão em um robô que utiliza esteiras como tração.

Fonte: (BORENSTEIN, EVERETT e FENG, 1996).

- Calibração sistemática – uma maneira de aumentar a precisão odométrica (sem adicionar dispositivos ou sensores) é calibrar cuidadosamente o robô. Porém, a calibração é difícil porque a cada minuto ocorrem desvios na geometria do veículo ou em suas partes que podem causar substanciais erros odométricos. Assim, uma calibração sistemática freqüente pode reduzir os efeitos das mudanças na geometria do veículo mantendo a navegação odométrica precisa (BORENSTEIN, EVERETT e FENG, 1996).

Para redução de erros odométricos não-sistemáticos existem algumas técnicas que são descritas abaixo:

- Referenciação mútua – este procedimento propõe a utilização de dois robôs. Um utiliza como referência o outro. Desta forma, é possível corrigir

erros não-sistemáticos, mas, limita a eficiência dos robôs (BORENSTEIN, EVERETT e FENG, 1996).

- Correção interna de erro de posição – com este procedimento dois robôs corrigem seus erros mutuamente. Porém, neste método os dois robôs estão em contínuo funcionamento e o erro é medido medindo-se a distância relativa entre os dois. Com este método é possível corrigir erros não-sistemáticos, porém, o modelo cinemático do sistema é bastante complexo (BORENSTEIN, EVERETT e FENG, 1996).

Apêndice B – Sistema de Navegação Inercial

B.1 Dead Reckoning

“Dead Reckoning” (originalmente chamado “ded-reckoning” como abreviação de “deduced reckoning” ou cálculo deduzido/estimado) ou navegação por cálculo estimado é o nome dado ao processo de determinar uma mudança na posição e orientação de um veículo no tempo pela integração de sua velocidade no tempo. Durante muito tempo, este foi o principal método de estimativa de posição nas navegações marítimas. Basicamente, o processo consiste em a cada meia hora armazenar a leitura da bússola, estimativas de força e direção do vento, a forma que as velas foram ajustadas e uma estimativa do sentido das correntes marítimas. Estes dados eram usados para cálculo (reckoning) da nova posição do navio.

Em robótica, “dead reckoning” é o processo de estimativa da posição do robô depois de passado algum tempo pela integração da velocidade no tempo. A Odometria descreve *dead reckoning* onde a posição estimada é obtida pela integração da velocidade ou diretamente de *encoders* montados nas rodas dos robôs.

Assim, podemos dividir a Odometria em dois tipos principais de navegação a navegação Odométrica, onde se utilizam encoders e a navegação inercial, onde são utilizados acelerômetros e giroscópios. A navegação odométrica foi tratada no texto principal, neste apêndice trataremos um pouco mais dos sistemas de navegação inercial.

B.2 Sistemas de Navegação Inercial

Uma montagem dos sensores inerciais em uma superfície rigidamente encaixada a uma base comum de forma a manter as mesmas orientações relativas é

chamada de Montagem de Sensores Inerciais (ISA – *Inertial Sensor Assembly*). Normalmente uma ISA contém três acelerômetros e três giroscópios ou uma configuração equivalente com o uso de sensores de múltiplos eixos (GREWAL, WEILL e ANDREWS, 2007).

O conjunto de uma ISA com a eletrônica de suporte associada para calibração e controle é chamado de Unidade de Medição Inercial (IMU – *Inertial Measurement Unit*). A eletrônica de suporte que está dentro da IMU também inclui controle ou compensação térmica, condicionamento de sinais, e controle das entradas e saídas. A IMU pode também incluir um processador, e eletrônica de controle mecânico (quando forem necessários) (GREWAL, WEILL e ANDREWS, 2007).

Um sistema de navegação inercial (INS – *Inertial Navigation System*) consiste em uma IMU mais o seguinte (GREWAL, WEILL e ANDREWS, 2007):

- Computadores de navegação: um ou mais computadores para cálculo da aceleração gravitacional (não medida pelos acelerômetros) e processar as medidas dos acelerômetros e giroscópios obtidas da IMU para manter uma estimativa de posição da IMU. Resultados intermediários do método de implementação normalmente inclui estimativas de velocidade, atitude, e variação de atitude da IMU.
- Interface com o usuário: tais como *displays* para operadores humanos e interfaces de dados analógicas e/ou digitais para guiagem do veículo e funções de controle.
- Fontes de alimentação: Alimentação e/ou armazenamento de energia para todo o sistema INS.

Engenheiros projetaram muitos diferentes sistemas de navegação inercial com características de longa distância melhores em desempenho. Eles geralmente recaem em duas categorias (GREWAL, WEILL e ANDREWS, 2007):

- *Gimbaled* ou “Suporte Cardã” (também chamado de Sistemas Isolados de Rotação (SANTANA e FURUKAWA, 2005), Plataforma convencional (OLIVEIRA, 2000) ou Plataforma mecânica (TORRES e HEMERLY,

2002)): é aquele em que a ISA é isolada de rotações do veículo, como mostra as figuras Figura 85 e Figura 86.

- *Strapdown* ou “Solidários” (PAIVA, NETTO e PAIVA, 2005) (também chamado de Plataforma Analítica (OLIVEIRA, 2000)): Neste caso a ISA não é isolada das rotações do veículo, mas é montada de forma “quase rigidamente” (*quasirigidly*) na estrutura do veículo. A plataforma solidária é mostrada na Figura 87. O termo “Quase-rígido” é aplicado a montagens de IMU com algum isolamento capaz de isolar a IMU de choques e vibrações transmitidas através da estrutura do veículo (GREWAL, WEILL e ANDREWS, 2007). O jargão *strapdown* foi criado a partir da expressão “*strapped down*” que literalmente significa “amarrado” se referindo a fixação deste tipo de INS no veículo (LAWRENCE, 1998).

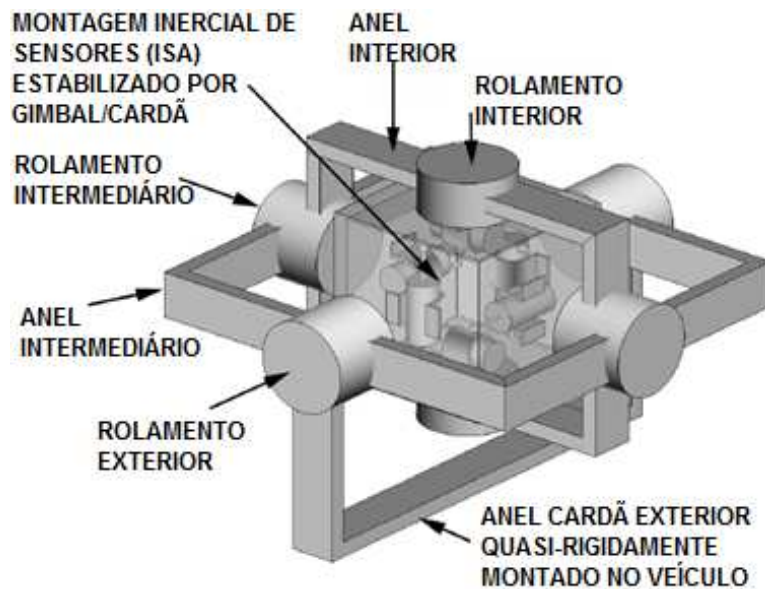


Figura 85 – Unidade de medição inercial montada sob plataforma mecânica de suporte cardã (Gimbaled IMU).

Fonte: (GREWAL, WEILL e ANDREWS, 2007).

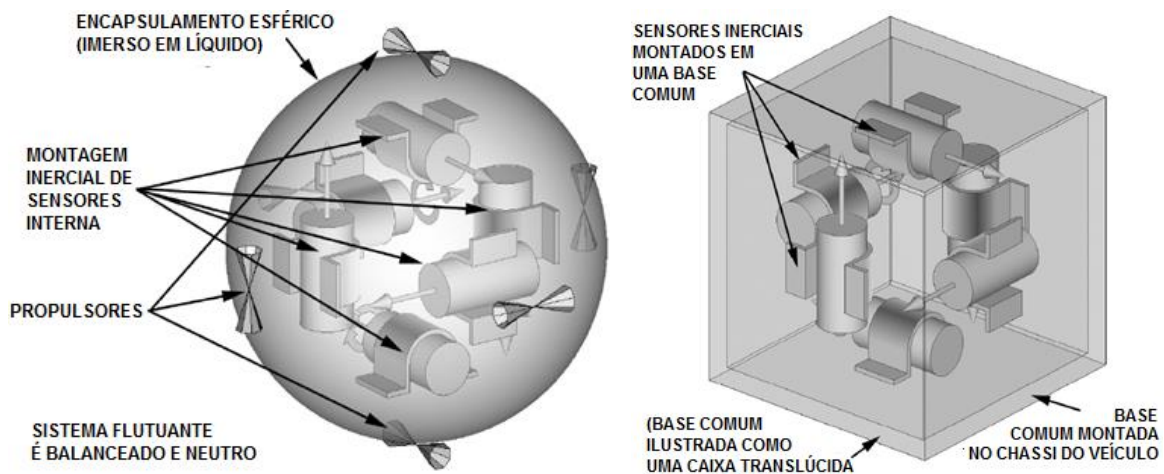


Figura 86 – Unidade de medição inercial montada sob plataforma mecânica flutuante (Floatated IMU) e uma unidade de medição inercial solidária (Strapdown ISA).

Fonte: (GREWAL, WEILL e ANDREWS, 2007).

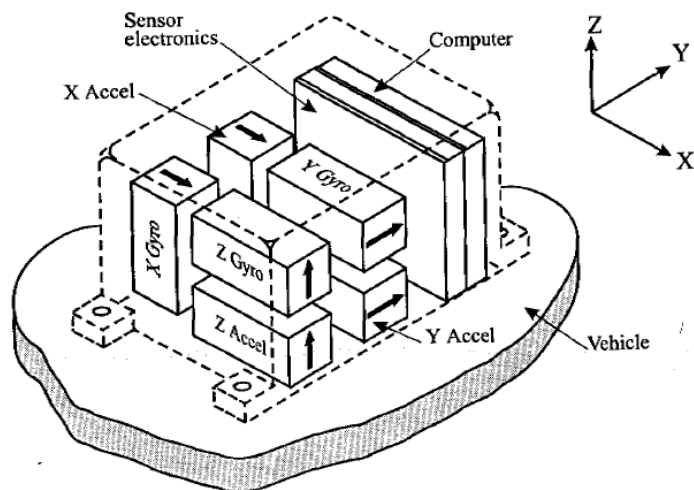


Figura 87 – Esquema de montagem de um sistema de navegação inercial solidário (Strapdown INS).

Fonte: (LAWRENCE, 1998).

A Tabela 11 faz um comparativo entre as montagens de UMI através de suas vantagens e desvantagens.

Tabela 11 – Vantagens e desvantagens das plataformas inerciais mecânicas e analíticas.

	<i>Vantagens</i>	<i>Desvantagens</i>
<u>Plataforma mecânica (gimbaled)</u>	• isolamento dos sensores inerciais das altas variações angulares, que eliminam muitos erros de sensores relacionados a variação e geralmente permite sensores de maior precisão. • auto-calibráveis orientando a ISA com a gravidade (para calibração dos acelerômetros) e com respeito ao eixo de rotação da terra (para calibração dos giroscópios), e utilizando um autocolimador óptico externo com espelhos na ISA para medir independentemente sua orientação em relação ao ambiente (GREWAL, WEILL e ANDREWS, 2007).	• custo • peso • volume • curvatura do cardã (importante em ambientes com altas acelerações) • caminhos elétricos são necessários através da estrutura do suporte cardã de forma a fornecer energia para a IMU, e para os sensores. Isto requer “<i>Slip Rings</i>”, que podem introduzir ruído ou “<i>cable wraps</i>” nos rolamentos do suporte cardã. Embora, projetos mais novos tenham utilizado métodos sem fio para transmissão de sinal e potência (GREWAL, WEILL e ANDREWS, 2007).
<u>Plataforma Mecânica (floated)</u>	• Isolação dos sensores inerciais das altas variações angulares, que eliminam muitos erros de sensores relacionados a variação e geralmente permite sensores de maior precisão. • Vantagem sobre sistemas cardã em que não existem estruturas cardã que funcionem em ambientes de forte carregamento de	• Dificuldade de acessar a ISA para testes, diagnóstico, manutenção ou reparos; • Necessita de meios para determinar a atitude da ISA relativamente ao veículo. • Necessita de métodos sem fio para prover energia e dados (comandos) para a ISA

	<i>Vantagens</i>	<i>Desvantagens</i>
	aceleração. • A habilidade de orientar a esfera permite auto-calibração, mas o problema da autocolimação do elemento estável é mais difícil que nos sistemas com eixos cardã.	através do fluido.
<u>Plataforma Analítica (strapdown)</u>	• Custo • Provêm dados de variação de rotação mais precisos;	• Alto custo dos sensores quando o sistema opera em altas variações. • Necessário grande range dinâmico das entradas • Necessita maior estabilidade de fator de escala para alcançar o mesmo desempenho da plataforma mecânica. • Geralmente necessita de intervalos de integração menores. (aumento de custo de processadores). • Custo da calibração e teste do giroscópio que necessita de uma tabela de taxas de precisão.

Fonte: (GREWAL, WEILL e ANDREWS, 2007, p. 24-27)

Na Figura 88, é mostrado um diagrama da implementação em software dos cálculos referentes a plataforma cardã. E na Figura 89 é mostrado o diagrama de implementação da plataforma solidária.

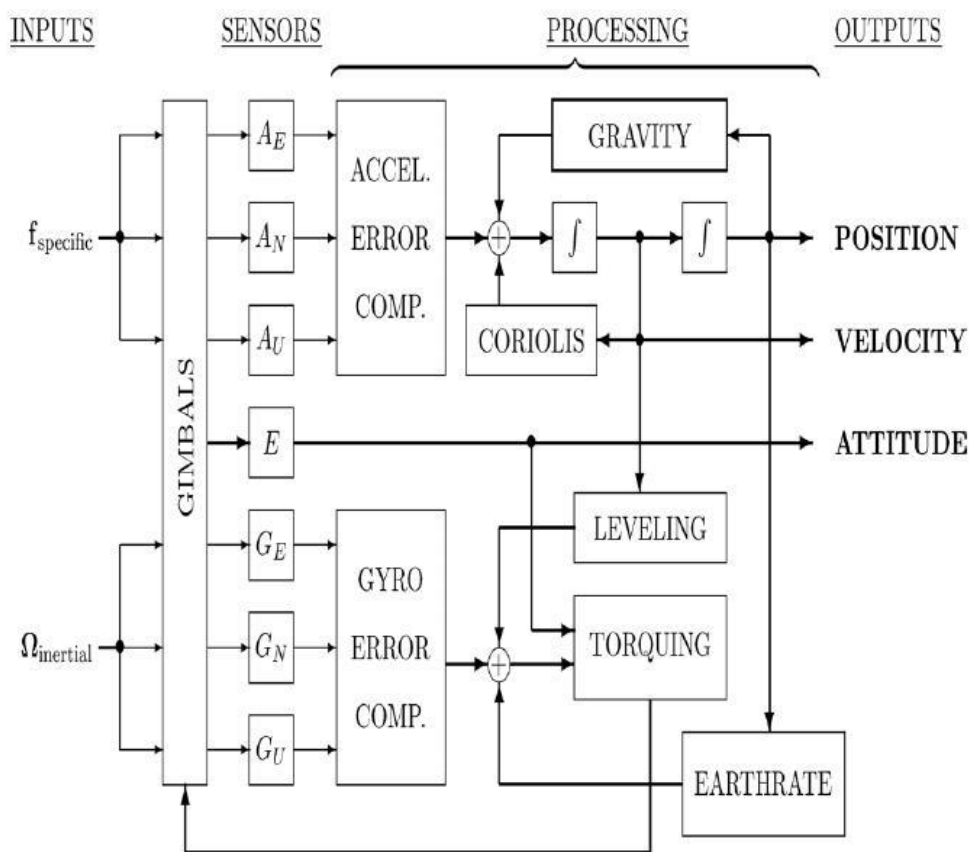


Figura 88 – Diagrama de processamento de sinais de navegação utilizando unidade de medição montada sob plataforma mecânica de suporte cardã (Gimbaled IMU).

Fonte: (GREWAL, WEILL e ANDREWS, 2007)

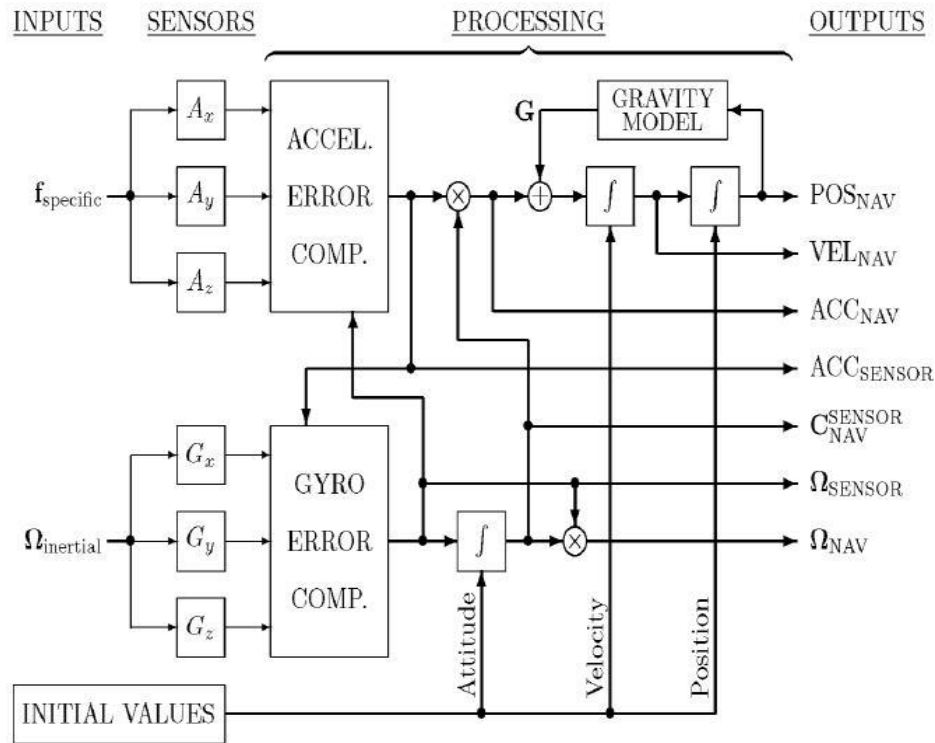


Figura 89 – Diagrama de processamento de sinais de navegação utilizando unidade de medição solidária (Strapdown ISA) e sistema de coordenadas de Euler.

Fonte: (GREWAL, WEILL e ANDREWS, 2007)

Onde:

$f_{specific}$ são as forças que geram as acelerações medidas pelos acelerômetros;

$\Omega_{inertial}$ são as velocidades angulares medidas pelo giroscópio;

POS_{NAV} é a posição do veículo robótico nas coordenadas de navegação;

VEL_{NAV} é a velocidade do veículo robótico nas coordenadas de navegação;

ACC_{NAV} é a aceleração do veículo robótico nas coordenadas de navegação;

ACC_{SENSOR} é a aceleração do veículo robótico nas coordenadas do acelerômetro fixo.

Apêndice C – Barramento CAN - CANBus

O barramento CAN (*Controller Area Network*) é um padrão ISO (*International Standards Organization*) homologado para sistemas que necessitam de um barramento de baixo custo, tolerante a falhas e um protocolo de comunicação robusto em tempo real. Inicialmente, em 1984, o *CANBus* foi desenvolvido pela empresa Robert Bosch para transferências de dados em veículos automotivos (BOSCH, 1991; CAN IN AUTOMATION GMBH, 1998; CAN IN AUTOMATION GMBH, 1998). Porém, os circuitos integrados dedicados se tornaram disponíveis somente em 1987 quando então o barramento CAN foi utilizado pela primeira vez em automóveis, isto só ocorreu em 1992. Uma versão preliminar da padronização internacional foi introduzida em 1991 e se tornou um padrão ISO (ISO-11898) em 1994.

A estrutura de barramento serial proporciona duas vantagens sobre os barramentos paralelos: aumento da confiabilidade em transferências mesmo sobre grandes distâncias e custos mais favoráveis. As especificações do barramento CAN definem um barramento baseado em prioridades que possui um acesso múltiplo sensível à uma portadora com detecção de colisões e uma arbitração não destrutiva (CSMA/CD+NDA). Este esquema de arbitração baseada em *hardware* proporciona uma maior consistência nos dados transmitidos no sistema (BECKE, BORGERT, *et al.*, 2000; interfacebus.com; PENG, 2005).

O barramento CAN permite recepção multiplexada com sincronização, detecção e sinalização de erro. Mensagens corrompidas são automaticamente retransmitidas e nós (unidades transmissoras/receptoras presentes no barramento) defeituosos são automaticamente removidos do circuito. Além disto, os nós podem fazer pedidos de dados remotamente. A sinalização utiliza codificação NRZ (*non-return to zero*) e permite a completa isolamento dos fios de interconexão (RICHARDS, 2002; PAZUL, 1999).

C.1 Funcionamento

A informação transmitida no CANbus é organizada em pequenos pacotes chamados “Frames”. Um Frame consiste em algumas informações de identificação de prioridade, até oito bytes de dados, e um bloco de checagem de erro, conforme mostra a Figura 90.

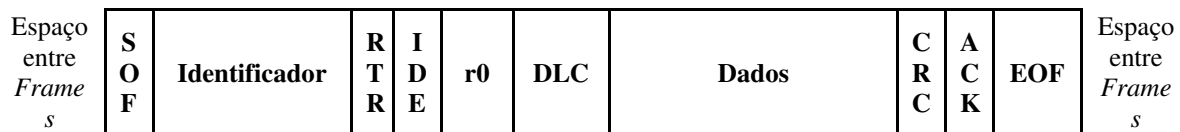


Figura 90 - Frames do barramento CAN.

Fonte: (BOSCH, 1991; RICHARDS e STONEKING, 2001)

Pela Figura 90 temos que um *frame* possui um bit dominante (nível lógico ‘0’) como marcador de início de um *frame* (SOF – *Start-of-Frame*), 11 bits para o identificador da mensagem, 1 bit para indicar se o nó remoto está pedindo para transmitir (RTR – *Remote Transmission Request*), 1 bit dominante (nível lógico ‘0’) indicando se o formato da mensagem é estendido ou não (IDE – *Identifier Extension*), 1 bit dominante (nível lógico ‘0’) reservado (r0), 4 bits indicando quantos bytes de dados se seguem (DLC – *Data Length Code*), até 8 bytes de dados (conforme número indicado em DLC), 2 bytes de checagem de erro (CRC – *Cyclical Redundancy Check*), 2 bits de confirmação de recebimento de mensagem (ACK – *Acknowledge*) e 7 bits recessivos (nível lógico ‘1’) como indicação de fim de mensagem (EOF – *End-of-Frame*) (BOSCH, 1991; CAN IN AUTOMATION GMBH, 1998; CAN IN AUTOMATION GMBH, 1998).

Quando um *frame* é transmitido, cada nó receptor deve responder se o *frame* foi recebido corretamente inserindo um bit de confirmação no espaço ACK deixado no *frame* durante a transmissão. Desta forma, o nó transmissor pode determinar que pelo menos um nó recebeu corretamente o *frame*.

O protocolo *CANbus* é endereçado por dados/mensagens, ao invés de endereçado por nó. Ou seja, o conteúdo da mensagem é marcado com um identificador que é único na rede. Todos os outros nós da rede recebem a mensagem e cada um executa um

teste de compatibilidade com o identificador para determinar se a mensagem, e seu conteúdo, é relevante para o nó. O bloco “Identificador” do *frame CANbus* é usado tanto para identificar uma mensagem como para indicar a sua prioridade (BOSCH, 1991).

A arbitragem do barramento é feita pelo CSMA/CD+NDA (*Carrier Sense Multiply Access/Collision Detect + Non-Destructive Arbitration*). Quando um nó está pronto pra transmitir uma mensagem pela rede, ele primeiro precisa verificar se o barramento está no estado inativo pela existência ou não da portadora (*Carrier Sense*). Se o barramento está inativo, o nó se torna mestre do barramento e transmite sua mensagem. Todos os outros nós detectam o início de um *frame*, trocam para o modo de recepção, e enviam uma confirmação (ACK) depois da correta recepção da mensagem (BOSCH, 1991).

Se mais de um nó começar a transmitir ao mesmo tempo (*Multiply Access*), a colisão das mensagens é evitada por um esquema de arbitragem bit a bit. Como cada nó transmissor envia bits do identificador de sua mensagem, este monitora os bits do barramento. O nó que envia um bit recessivo (nível lógico ‘1’) e detecta um bit dominante (nível lógico ‘0’) e então muda para o modo de recepção (*Collision Detect/Non-Destructive Arbitration*). Isto resulta em um esquema de priorização em que mensagens de maior prioridade recebem um “Identificador” com valor numérico menor. Nós cujas mensagens são interrompidas por mensagens de maior prioridade tentarão reenviar a mensagem quando o barramento se tornar inativo novamente.

A Figura 91 demonstra o processo de arbitragem em que um nó perde o estado de mestre do barramento. O sinal superior mostra a saída do nó em questão, enquanto que o sinal inferior mostra os bits atualmente presentes no barramento.

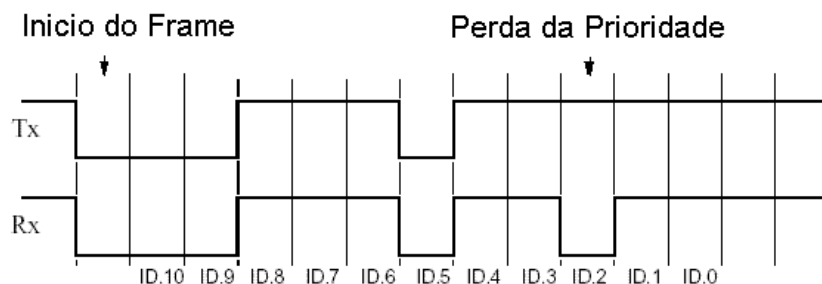


Figura 91 – Esquema de arbitragem do barramento CAN.

Na Figura 91, quando o nono bit identificador (ID.2) é transmitido, o nó que está transmitindo (Tx) observa que o nível de sinal no barramento é dominante, enquanto que a saída do nó (Rx) é recessiva. Logicamente, isto representa um caso em que o nó em questão deseja transmitir uma mensagem com prioridade “0011101111” (lógico) enquanto que um nó remoto no barramento deseja enviar uma mensagem com prioridade “00111011011”. Desta forma, o nó que está transmitindo perde a arbitração desde que o nó remoto que deseja transmitir uma mensagem de maior prioridade. Quando o nó que está transmitindo detecta que perdeu o estado de mestre do barramento (quando recebe ID.2, neste caso), ele muda para o modo de recepção (CAN IN AUTOMATION GMBH, 1998).

C.2 Identificadores

Conforme já foi dito, o identificador possui as funções de identificação e determinação da prioridade da mensagem. Assim, quanto menor o valor numérico do identificador maior é a prioridade da mensagem. O acesso ao barramento é garantido à mensagem de maior prioridade assim que a mensagem é transmitida. Mensagens de prioridades mais baixas são automaticamente retransmitidas no próximo ciclo do barramento, ou em um ciclo de barramento subsequente se existirem outras mensagens de maiores prioridades aguardando ser enviadas.

C.3 Endereçamento e arbitração

Existem dois estados elétricos no barramento: dominante, que é representado ‘0’ ou ‘ativo’, e recessivo, que é representado ‘1’ ou ‘inativo’. Se algum nó transmite um bit dominante, então o barramento irá estar no estado dominante. Porém, todos os nós devem transmitir um bit recessivo para o barramento ser recessivo (BOSCH, 1991; CAN IN AUTOMATION GMBH, 1998).

Um identificador de 11-bits permite até $2^{11}=2048$ mensagens, porém, o *CANbus* permite apenas 2032. As 16 mensagens faltantes correspondem ao caso em que o sétimo bit mais significativo do identificador é recessivo; tais casos não são permitidos, pois o delimitador (*end-of-frame*) do pacote de dados consiste em 7 bits recessivos consecutivos.

Assim, cada nó monitora o barramento, e não inicializa a transmissão se um *frame* está sendo transmitido por outro nó. Se dois nós tentam transmitir exatamente ao mesmo tempo, os identificadores (que são transmitidos primeiro) irão sofrer uma intersecção. Como o protocolo proíbe dois nós transmitirem o mesmo identificador, o estado de saída dos dois devem ser diferentes em algum momento durante o identificador. Neste caso, o nó que transmite o bit recessivo irá ver um bit dominante no barramento e cessar a sua transmissão. Desde que o bit mais significativo do identificador é transmitido primeiro, o menor valor binário do identificador sempre “ganha”, pois tem sempre prioridade maior (KENNAN e MCFADDEN, 1999; BOSCH, 1991).

C.4 Checagem de Erro

O protocolo CAN tem cinco métodos de checagem de erro, três em nível de mensagem e dois no nível de bit. Se uma mensagem falha, em qualquer um destes métodos de detecção, ela não será aceita e um frame de erro será gerado que fará que todos os outros nós ignorem a mensagem defeituosa. Isto faz com que o nó transmissor reenvie a mensagem.

No nível de mensagem, uma checagem de redundância cíclica (CRC - *Cyclical-Redundancy Check*), um campo de confirmação (ACK) embutido na mensagem e uma checagem de forma são usados para detectar erros. O CRC de 15-bits é calculado sobre o campo identificador e os bytes de dados, e então comparado ao campo CRC colocado na mensagem. Se os valores diferem, um erro é gerado. O campo de confirmação (ACK) é de dois bits que consistem em um bit ACK e um bit delimitador. O transmissor irá colocar um bit recessivo no bit de confirmação (ACK). Todo nó que receber a mensagem corretamente escreverá um bit dominante no campo de confirmação. Se o transmissor não detectar um bit

dominante no campo de confirmação da mensagem, este gerará um *frame* de erro e retransmitirá a mensagem. O terceiro método de checagem de erro em nível de mensagem é uma checagem de forma que verifica se certos campos na mensagem contêm somente bits recessivos. Se um bit dominante é detectado, um erro é gerado. Os bits checados são “*start-of-frame*”, “*end-of-frame*”, “*acknowledge delimiter*” e “*CRC delimiter*”.

No nível de bits, cada mensagem é monitorada pelo transmissor. Se um bit é escrito no barramento e seu complemento é lido, um erro é gerado. As exceções a esta regra são os bits do campo identificador, que utiliza contenção de barramento em um esquema de arbitração não destrutiva (NDA – *Nondesctructive Arbitration*), e o *slot* de confirmação (ACK), que requer que um bit recessivo seja sobrescrito por um dominante quando a mensagem é corretamente recebida por um nó remoto.

Um método adicional de detecção de erro utiliza uma regra de checagem de bits, onde um nó, que transmite uma mensagem que contém cinco bits consecutivos de um mesmo nível lógico, deve então transmitir um único bit do nível complementar para prevenir a presença de um nível DC no barramento. Se o próximo bit não é um complemento depois de cinco bits consecutivos de um mesmo nível lógico, então um erro é gerado. Este método é utilizado para os bits entre o “*start-of-frame*” e o “*CRC delimiter*”.

C.5 Realização

O *CANbus* é um protocolo multicamada. Como implementado aqui, existem três camadas principais:

- Camada de aplicação
- Camada de protocolo
- Camada física

A camada de aplicação é o dispositivo que utiliza o *CANbus*. Neste projeto, a camada de aplicação é realizada por um microcontrolador que interpreta e utiliza os dados trafegados no barramento. Assim, cada subsistema (módulo), que requer acesso ao

barramento contém um microcontrolador, que irá visualizar o barramento *CANbus* como uma simples porta de entrada/saída (IO) mapeada em memória.

A camada de protocolo mantém a coerência e gerencia todas as atividades do barramento CAN (CANopen - CiA DS-301 v.4.02 - Application Layer and Communication Profile, 2002; CANopen - the standardized embedded network; ZELWANGER). São suas responsabilidades:

- Filtragem e validação das mensagens
- Confinamento de falhas
- Detecção de erro e sinalização
- Arbitração
- Conformação de mensagens (*Message Framing*)
- Taxas de transferência e gerenciamento de tempos (*Transfer rate and timing management*)

Tipicamente, a camada de protocolo é implementada utilizando um circuito integrado controlador CAN disponível no mercado.

A camada física define como os sinais são transmitidos, executando a representação em nível de sinais dos dados. Esta camada conecta o controlador CAN ao meio de transmissão, que é frequentemente um cabo blindado ou um par trancado para reduzir a interferência eletromagnética (EMI – *Electromagnetic Interference*).

Apêndice D – Baterias Recarregáveis

Uma bateria é composta, basicamente, de células menores de características peculiares que determinam as características da bateria. A corrente que uma célula pode fornecer depende de vários fatores: tipo de reação química, velocidade da reação e desenho da célula (sobretudo, a dimensão da área onde a reação ocorre). O efeito destes fatores químicos e mecânicos pode ser expresso por meio de uma quantidade físico-matemática denominada “resistência interna”. A diminuição do valor da resistência interna permite obter correntes mais elevadas. Alguns tipos de células são especialmente recomendadas para fornecer correntes intensas. Um curto-circuito numa pilha de níquel-cádmio pode produzir uma corrente suficientemente elevada para derreter metais e iniciar incêndios. No entanto, existem outros tipos de células que apenas podem gerar poucas intensidades de corrente (HELLES, 2004; HELLES, 2004; PASSERN, 2005; PASSERN, 2005).

Nenhuma pilha tem vida eterna. Inevitavelmente, os elementos químicos vão-se degradando lentamente, provocando a diminuição da capacidade da pilha.

O quociente “energia armazenada/peso” de uma pilha (ou “energia armazenada/volume”) é denominado “densidade de energia”. É desejável obter uma densidade alta para conseguir armazenar mais energia numa pilha com um determinado peso (ou volume).

A Tabela 12 apresenta a tensão nominal e a densidade de energia (expressas em Wh por kg) para os vários tipos de pilhas mais importantes, utilizados em computadores portáteis e telefones móveis. A Tabela 13 apresenta dados que possibilitam a comparação rápida de três tipos de pilhas, para facilitar a escolha da aplicação. Note que na Europa e em outros países, já é proibido fabricar pilhas de NiCd, em breve estas pilhas deverão ser banidas definitivamente.

Tabela 12 – Densidade de energia em baterias.

Tipo de célula	Tensão Nominal (V)	Densidade de energia típica (Wh/kg)
Chumbo	2.1	30
Níquel Cádmio	1.2	40 a 60
Níquel hidreto metálico	1.2	60 a 80
Íons de Lítio, circular	3.6	90 a 100
Íons de Lítio, prismática	3.6	100 a 110
Íons de Lítio, polímero	3.6	130 a 150

Tabela 13 – Comparativo entre baterias.

Parâmetro	Níquel Cádmio	Níquel hidreto metálico	Íons de Lítio
Densidade de energia	Baixa	Média	Alta
Armazenamento de energia	Baixa	Média	Média
Tempo de vida	Alta	Alta	Alta
Custo	Baixa	Média	Alta
Segurança	Alta	Alta	Média
Ambiente	Baixa	Média	Média

Apêndice E – Mensagens CAN

Neste apêndice foram adicionadas as tabelas de sugestão para cada módulo listado no Sistema Autônomo Completo. As tabelas foram preparadas pensando no sistema autônomo completo como um todo e devem ser adaptadas para cada sistema de aplicação.

Tabela 14 – Sugestão de mensagens CAN para o módulo SP.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARSP CB.COMANDO.DESLIGARSP	CB	SP	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do SP. Mensagens de eventos continuam sendo geradas como resposta a eventos críticos, mas, SP.TEMP e SP.PRESS não são enviadas pelo barramento.
ME.COMANDO.LIGARSP ME.COMANDO.DESLIGARSP	ME	SP	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
SP.TEMP (dados de temperatura)	SP	CB, RB, ME	Mensagem proveniente do SP contendo as leituras de temperaturas do veículo.
SP.PRESS (dados de pressão)	SP	CB, RB, ME	Mensagem proveniente do SP contendo as leituras de pressão do veículo.
SP.EVENTO	SP	CB, RB, ME	Mensagem proveniente do SP que marca a ocorrência de evento ou um reinício do sistema

Mensagem	Origem	Destino	Descrição
SP.ESTADO	SP	CB, RB, ME	pelo circuito “cão de guarda”. Mensagem proveniente do SP contendo informações sobre os estados internos do SP, a voltagem (estado da bateria) do sistema, e as condições dos sensores que o SP monitora. Esta mensagem é gerada quando o SP recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 15 – Sugestão de mensagens CAN para o módulo CO.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARCO CB.COMANDO.DESLIGARCO	CB	CO	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do CO.
ME.COMANDO.LIGARCO ME.COMANDO.DESLIGARCO	ME	CO	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
CO.DPOS	CO	CB, RB, ME	Mensagem proveniente do CO, contendo a variação de posição do veículo.
CO.VMED	CO	CB, RB, ME	Mensagem proveniente do CO, contendo a velocidade média do veículo.
CO.EVENTO	CO	CB, RB, ME	Mensagem proveniente do CO que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
CO.ESTADO	CO	CB, RB, ME	Mensagem proveniente do CO contendo informações sobre os estados internos do CO, a voltagem (estado da bateria) do sistema, as condições dos sensores que o CO monitora e os últimos dados de posição e velocidade média do veículo . Esta mensagem é gerada quando o CO recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 16 – Sugestão de mensagens CAN para o módulo UMI.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARUMI CB.COMANDO.DESLIGARUMI	CB	UMI	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do UMI.
ME.COMANDO.LIGARUMI ME.COMANDO.DESLIGARUMI	ME	UMI	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
UMI.XYZ (dados dos acelerômetros dos eixos X, Y e Z)	UMI	CB, RB, ME	Mensagem proveniente da UMI com os pacotes de dados dos sensores de navegação inercial. Estes dados serão armazenados pelo RV e enviado para a equipe de solo através do MR.
UMI.ABC (dados dos giroscópios dos eixos a, b e g)	UMI	CB, RB, ME	Cada mensagem contém valores de 3x12-bit correspondendo às aquisições dos eixos X, Y e Z ou a, b e g.
UMI.ACEL.n (dados de aceleração do eixo n, n={X,Y,Z}) UMI.VEL.n (dados de velocidade do eixo n, n={X,Y,Z}) UMI.POS.n (dados de posição do eixo n, n={X,Y,Z}) UMI.ORI.n (dados de orientação do eixo n, n={a,b,g})	UMI	CB, RB, ME	Mensagem proveniente da UMI com dados de aceleração, velocidade, posição e orientação. Estes dados devem ser encontrados por processamento interno à UMI.
UMI.EVENTO	UMI	CB, RB, ME	Mensagem proveniente do UMI que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
UMI.ESTADO	UMI	CB, RB, ME	Mensagem proveniente da UMI que mostra o estado atual da UMI, ou seja, se está com a saída de dados habilitada ou desabilitada e em

Mensagem	Origem	Destino	Descrição
			que modo de operação esta trabalhando. Esta mensagem é gerada quando o UMI recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 17 – Sugestão de mensagens CAN para o módulo TT.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARTT CB.COMANDO.DESLIGARTT	CB	TT	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do TT.
ME.COMANDO.LIGARTT ME.COMANDO.DESLIGARTT	ME	TT	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
TT.POS	TT	CB, RB, ME	Mensagem proveniente do TT contendo as coordenadas de posição do veículo com referência as coordenadas do ambiente.
TT.EVENTO	TT	CB, RB, ME	Mensagem proveniente do TT que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
TT.ESTADO	TT	CB, RB, ME	Mensagem proveniente do TT contendo informações sobre os estados internos do módulo TT, a voltagem (estado da bateria) do sistema, as condições dos sensores que o TT monitora e os últimos dados de posição do veículo. Esta mensagem é gerada quando o TT recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 18 – Sugestão de mensagens CAN para o módulo MDGPS.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARMDGPS CB.COMANDO.DESLIGARMDGPS	CB	MDGPS	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do MDGPS.
ME.COMANDO.LIGARMDGPS ME.COMANDO.DESLIGARMDGPS	ME	MDGPS	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
MDGPS.DADOS	MDGPS	CB, RB, ME	Mensagem proveniente do módulo MDGPS com os pacotes de dados de posição do veículo.
MDGPS.EVENTO	MDGPS	CB, RB, ME	Mensagem proveniente do MDGPS que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
MDGPS.ESTADO	MDGPS	CB, RB, ME	Mensagem proveniente do MDGPS que mostra o estado atual do receptor GPS, ou seja, se está com a saída de dados habilitada ou desabilitada e em que modo de operação o módulo esta trabalhando. Esta mensagem é gerada quando o MDGPS recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 19 – Sugestão de mensagens CAN para o módulo RM.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARRM CB.COMANDO.DESLIGARRM	CB	RM	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do RM.
ME.COMANDO.LIGARRM ME.COMANDO.DESLIGARRM	ME	RM	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
RM.DADOS	RM	CB, RB, ME	Mensagem proveniente do módulo RM com os pacotes de dados de posição do veículo.
RM.EVENTO	RM	CB, RB, ME	Mensagem proveniente do RM que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
RM.ESTADO	RM	CB, RB, ME	Mensagem proveniente do RM que mostra o estado atual do módulo e em que modo de operação o módulo esta trabalhando. Esta mensagem é gerada quando o RM recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 20 – Sugestão de mensagens CAN para o módulo LM.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARLM CB.COMANDO.DESLIGARLM	CB	LM	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do LM.
ME.COMANDO.LIGARLM ME.COMANDO.DESLIGARLM	ME	LM	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
LM.POSICAO (dados de posição)	LM	CB, RB, ME	Mensagem proveniente do LM contendo os resultados dos cálculos de posição a partir das marcas encontradas.
LM.EVENTO	LM	CB, RB, ME	Mensagem proveniente do LM que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
LM.ESTADO	LM	CB, RB, ME	Mensagem proveniente do LM contendo informações sobre os estados internos do módulo LM, a voltagem (estado da bateria) do sistema, as condições dos sensores que o LM monitora e os últimos dados de posição do veículo. Esta mensagem é gerada quando o LM recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 21 – Sugestão de mensagens CAN para o módulo VC.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARVC CB.COMANDO.DESLIGARVC	CB	VC	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do VC.
ME.COMANDO.LIGARVC ME.COMANDO.DESLIGARVC	ME	VC	
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
VC.POSICAO.X (dados de posição do objeto/marca X)	VC	CB, RB, ME	Mensagem proveniente do VC contendo os resultados dos cálculos de posição do objeto/marca 'X' segundo o eixo de coordenadas do veículo.
VC.EVENTO	VC	CB, RB, ME	Mensagem proveniente do VC que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
VC.ESTADO	VC	CB, RB, ME	Mensagem proveniente do VC contendo informações sobre os estados internos do módulo VC, a voltagem (estado da bateria) do sistema, as condições das câmeras e os últimos dados de posição do veículo. Esta mensagem é gerada quando o VC recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 22 – Sugestão de mensagens CAN para o módulo MAC.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.ARMAR_n CB.COMANDO.DESARMAR_n CB.COMANDO.ACAO_n	CB	MAC	Mensagem proveniente do CB requisitando armar, desarmar ou acionar uma porta (atuador ou motor).
CB.COMANDO.ICHK	CB	MAC	Mensagem proveniente do CB requisitando a execução teste de validade das suas portas (atuadores e/ou motores).
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
MAC.ICHK.n (verificar porta n, n={0...7})	MAC	CB, RB, ME	Mensagem proveniente do MAC contendo os resultados do teste de validade da porta n.
MAC.EVENTO.n (evento da porta n)	MAC	CB, RB, ME	Mensagem proveniente do MAC que é gerada com os eventos armar, desarmar ou acionar.
MAC.ESTADO.n	MAC	CB, RB, ME	Mensagem proveniente do MAC indicando o estado interno do módulo, incluindo se as portas estão armadas ou desarmadas e se estão no modo de operação de diagnóstico ou normal. Esta mensagem é gerada quando o MAC recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 23 – Sugestão de mensagens CAN para o módulo MA.

Mensagem	Origem	Destino	Descrição
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
MA.VOLTAGE	MA	CB, RB, ME	Mensagem proveniente do MA contendo as informações sobre a tensão do banco de baterias.
MA.TDESCARGA	MA	CB, RB, ME	Mensagem proveniente do MA informando em quantos minutos o veículo ficará sem carga na bateria.
MA.CICLOS	MA	CB, RB, ME	Mensagem proveniente do MA informando em quantos ciclos de carga/descarga as baterias do veículo ficarão sem carga.
MA.EVENTO	MA	CB, RB, ME	Mensagem proveniente do MA que marca a ocorrência de evento de erro ou um reinício do sistema pelo circuito “cão de guarda”.
MA.ESTADO	MA	CB, RB, ME	Mensagem proveniente do MA contendo informações sobre os estados internos do MA, a voltagem (estado da bateria) do sistema, as condições dos sensores que o MA monitora e os últimos dados de posição e velocidade média do veículo . Esta mensagem é gerada quando o MA recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 24 – Sugestão de mensagens CAN para o módulo CB.

Mensagem	Origem	Destino	Descrição
UMI.EVENTO	UMI	CB, RB, ME	Mensagem proveniente da UMI indicando evento ocorrido. Tal evento pode ser detecção de lançamento ou apogeu da trajetória.
UMI.ESTADO	UMI	CB, RB, ME	Mensagem proveniente da UMI mostrando os estados internos da UMI.
SP.EVENTO	SP	CB, RB, ME	Mensagem proveniente do módulo SP indicando evento ocorrido.
SP.ESTADO	SP	CB, RB, ME	Mensagem proveniente do módulo SP mostrando os estados internos do SP.
RB.EVENTO	RB	CB, RB, ME	Mensagem proveniente do RB mostrando o estado da memória Flash.
RB.ESTADO	RB	CB, RB, ME	Mensagem proveniente do RB mostrando os estados internos do RB.
MAC.ICHK.n	MAC	CB, RB, ME	Mensagem proveniente do MAC mostrando o estado da porta (atuador ou ignitor) n.
MAC.EVENTO.n	MAC	CB, RB, ME	Mensagem proveniente do MAC como resposta ao comando de armar/desarmar a porta (atuador ou ignitor) n.
MAC.ESTADO.n	MAC	CB, RB, ME	Mensagem proveniente do MAC mostrando o estado interno da porta (atuador ou ignitor) n, além de informar o modo de operação e se a porta está ativa.
MC.ESTADO	MC	CB, RB, ME	Mensagem proveniente do MC mostrando o estado interno do MC. Esta mensagem também deve retornar um byte com o último estado conhecido do CB.
ME.EVENTO	ME	CB, RB	Mensagem proveniente do ME que é gerada quando a ME recebe comando da equipe de solo determinando a separação do veículo.
ME.ESTADO	ME	CB, RB	Mensagem proveniente do ME que mostra o estado atual do ME, ou seja, em que modo de operação esta trabalhando.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, requisitando que todos os módulos sejam ligados ou desligados.
ME.COMANDO.LIGARCB ME.COMANDO.DESLIGARCB	ME	CB, RB	Mensagem proveniente do solo, através do ME, para ligar ou desligar a saída de mensagens de estado do CB (CB.ESTADO).
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos	Mensagem proveniente do solo, através do ME,

Mensagem	Origem	Destino	Descrição
		os Módulos	comandando todos os sistemas a entrarem no modo normal de operação.
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB para comandar todos os outros módulos a ligar ou desligar. Isto pode ser feito para economia de energia ou mesmo como um reinício forçado.
CB.COMANDO.ICHK	CB	MAC	Mensagem proveniente do CB pedindo ao MAC que verifique a validade de suas portas.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB pedindo o estado de operação interna a todos os módulos.
CB.COMANDO.ARMAR_n CB.COMANDO.DESARMAR_n	CB	MAC	Mensagem proveniente do CB comandando armar ou desarmar a porta (atuador ou motor) n. Recomenda-se enviar um código de acionamento junto com a mensagem de armar porta, tal que a porta somente seja acionada mencionando-se tal código de acionamento. O comando de desarmar apaga o código.
CB.COMANDO.ACAO_n	CB	MAC	Mensagem proveniente do CB acionando a porta (atuador ou motor) n.
CB.COMANDO.LIGARRB CB.COMANDO.DESLIGARRB	CB	RB	Mensagem proveniente do CB para habilitar ou desabilitar a gravação de dados do RB.
CB.COMANDO.LIGARUMI CB.COMANDO.DESLIGARUMI	CB	UMI	Mensagem proveniente do CB para habilitar ou desabilitar a saída de dados da UMI. A UMI deve continuar a amostrar dados, mas não deve colocar os dados no barramento.
CB.COMANDO.LIGARSP CB.COMANDO.DESLIGARSP	CB	SP	Mensagem proveniente do CB para habilitar ou desabilitar a saída de dados do SP. O SP deve continuar a amostrar dados, mas não deve colocar os dados no barramento.
CB.COMANDO.LIGARVC CB.COMANDO.DESLIGARVC	CB	VC	Mensagem proveniente do CB para habilitar ou desabilitar o envio de imagens do VC.
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB para informar ao sistema o estado atual do CB. Esta mensagem deve ser enviada regularmente ao sistema na mesma frequência do enlace principal de controle do CB como pulso de sincronismo. Todos os módulos conectados ao barramento devem aceitar esta mensagem. Esta mensagem funciona como um pulso de sincronismo do sistema, e contém os marcadores de tempo primários do sistema. Uma vez recebida esta mensagem, todos os módulos devem retornar suas próprias mensagens de estado para o CB, resultando num bloco contínuo de dados de todo o sistema sincronizado que é então armazenado no RB e enviado pela telemetria. Os bytes de

Mensagem	Origem	Destino	Descrição
			estado do CB incluem um marcador de tempo, o identificador do estado atual, e o estado do sistema. Se outros módulos conectados ao barramento são reiniciados pelo circuito “cão de guarda”, eles podem utilizar as informações de estado do sistema para ressincronizar os seus estados internos com o CB.

Tabela 25 – Sugestão de mensagens CAN para o módulo ME.

Mensagem	Origem	Destino	Descrição
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	Mensagem proveniente externamente ao veículo, através do ME, requisitando que todos os módulos sejam ligados ou desligados.
ME.COMANDO.LIGARCB ME.COMANDO.DESLIGARCB	ME	CB	Mensagem proveniente externamente ao veículo, através do ME, para ligar ou desligar a saída de mensagem de estado do CB.
ME.COMANDO.LIGARUMI ME.COMANDO.DESLIGARUMI	ME	UMI	Mensagem proveniente externamente ao veículo, através do ME, habilitando/desabilitando a saída de dados da UMI.
ME.COMANDO.LIGARSP ME.COMANDO.DESLIGARSP	ME	SP	Mensagem proveniente externamente ao veículo, através do ME, habilitando/desabilitando a saída de dados do SP.
ME.COMANDO.LIGARMDGPS ME.COMANDO.DESLIGARMDGPS	ME	MDGPS	Mensagem proveniente externamente ao veículo, através do ME, habilitando/desabilitando a saída de dados do MDGPS.
ME.COMANDO.LIGARRB ME.COMANDO.DESLIGARRB	ME	RB	Mensagem proveniente externamente ao veículo, através do ME, habilitando/desabilitando a gravação de dados do RB.
ME.COMANDO.APAGARRB	ME	RB	Mensagem proveniente externamente ao veículo, através do ME, comandando apagar o conteúdo da memória flash interna do RB.
ME.COMANDO.LIGARVC ME.COMANDO.DESLIGARVC	ME	VC	Mensagem proveniente externamente ao veículo, através do ME, habilitando/desabilitando a transmissão de imagens de vídeo do VC.

Mensagem	Origem	Destino	Descrição
ME.EVENTO	ME	CB, RB	Mensagem proveniente do ME que é gerada quando a ME recebe comando externo determinando algum acionamento.
ME.ESTADO	ME	CB, RB	Mensagem proveniente do ME que mostra o estado atual do ME, ou seja, em que modo de operação está trabalhando. Esta mensagem é gerada quando o ME recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

Tabela 26 – Sugestão de mensagens CAN para o módulo RB.

Mensagem	Origem	Destino	Descrição
CB.COMANDO.LIGAR CB.COMANDO.DESLIGAR	CB	Todos os Módulos	Mensagem proveniente do CB ou do solo, através do ME, requisitando ligar/desligar todos os módulos.
ME.COMANDO.LIGAR ME.COMANDO.DESLIGAR	ME	Todos os Módulos	
CB.COMANDO.LIGARRB CB.COMANDO.DESLIGARRB	CB	RB	Mensagem proveniente do CB ou do solo, através do ME, habilitando/desabilitando a saída de dados do RB.
ME.COMANDO.LIGARRB ME.COMANDO.DESLIGARRB	ME	RB	
ME.COMANDO.APAGARRB	ME	RB	Mensagem proveniente do solo, através do ME, comandando apagar o conteúdo da memória flash interna do RB.
CB.ESTADO	CB	Todos os Módulos	Mensagem proveniente do CB mostrando tanto os estados internos do CB bem como os bits de estado do restante do sistema.
CB.COMANDO.SCHK	CB	Todos os Módulos	Mensagem proveniente do CB requisitando o estado de todos os módulos.
ME.COMANDO.MDIAG	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo de diagnóstico.
ME.COMANDO.MNORMAL	ME	Todos os Módulos	Mensagem proveniente do solo, através do ME, comandando todos os sistemas a entrarem no modo normal de operação.
RB.EVENTO	RB	CB, ME	Mensagem proveniente do RB que é gerada em resposta a erros internos ao RB e comandos provenientes do CB e do ME ou um reinício do sistema pelo circuito “cão de guarda”.
RB.ESTADO	RB	CB, ME	Mensagem proveniente do RB que mostra o estado atual do RB, ou seja, se está com a gravação de dados habilitada ou desabilitada e em que modo de operação esta trabalhando. Esta mensagem é gerada quando o RB recebe uma mensagem CB.ESTADO ou CB.COMANDO.SCHK.

