

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

LUIS GUILHERME MACHADO CAMARGO
PEDRO ALBERTO DE BORBA
RICARDO FARAH
STEFAN CAMPANA FUCHS
TELMO FRIESEN

MAPEAMENTO DE AMBIENTES COM O ROBÔ BELLATOR

RELATÓRIO TÉCNICO

CURITIBA

2013

LUIS GUILHERME MACHADO CAMARGO
PEDRO ALBERTO DE BORBA
RICARDO FARAH
STEFAN CAMPANA FUCHS
TELMO FRIESEN

MAPEAMENTO DE AMBIENTES COM O ROBÔ BELLATOR

Relatório técnico apresentado à Unidade Curricular
de Oficina de Integração 3 do Curso de Engenharia
de Computação da Universidade Tecnológica Federal
do Paraná como requisito parcial para aprovação.

CURITIBA

2013

SUMÁRIO

1 INTRODUÇÃO	3
1.1 MOTIVAÇÃO	3
1.2 TRABALHOS CORRELATOS	4
1.2.1 PatrolBot	4
1.2.2 Sistema de mapeamento robótico bidimensional por infravermelho	5
1.3 OBJETIVOS	5
2 PLANEJAMENTO	7
2.1 CRONOGRAMA	7
2.2 ORÇAMENTO	7
2.3 DELIVERABLES	7
3 EXECUÇÃO	10
3.1 ARQUITETURA GERAL	10
3.1.1 Software da estação-base e do placa TS-7260	10
3.1.2 Hardware	13
3.2 ESTAÇÃO-BASE	15
3.2.1 Diagrama de classes da estação base	15
3.2.1.1 Descrição das classes da estação base	15
3.2.1.2 Pacote <i>visual</i>	16
3.2.1.3 Pacote <i>dados</i>	16
3.2.1.4 Pacote <i>comunicacao</i>	17
3.2.1.5 Pacote <i>gui</i>	18
3.3 SISTEMA DE COMUNICAÇÃO	19
3.3.1 Codificação das mensagens	19
3.4 SISTEMA EMBARCADO	23
3.4.1 Diagram de classes do sistema embarcado	23
3.5 CRONOGRAMA ORIGINAL VERSUS EXECUTADO	25
3.6 ORÇAMENTO ORIGINAL VERSUS EXECUTADO	26
4 CONCLUSÕES	27
4.1 ANÁLISE DO DESENVOLVIMENTO	27
4.2 INTEGRAÇÃO	28
4.3 TRABALHOS FUTUROS	28
Apêndice A – PLANEJAMENTO DE RISCOS	30
Apêndice B – MEDIDAS DO ROBÔ	39
Apêndice C – DIAGRAMAS DETALHADOS DE HARDWARE	40
C.1 DIAGRAMA ELÉTRICO/ELETRÔNICO	40
C.2 PLACA DE CIRCUITO IMPRESSO	42
Apêndice D – CRONOGRAMA	45
REFERÊNCIAS	46

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

O projeto apresentado neste documento trata-se do “Mapeamento de Ambientes com o robô Bellator” e é uma extensão do projeto “Bellator”. Ele teve sua última alteração em 2012 quando foi utilizado por Alexandre Jacques Marin, Júlio Cesar Nardelli Borges e Yuri Antin Wergrzn como plataforma de experimentos para o projeto final de conclusão de curso. O projeto para a disciplina de Oficina de Integração 3 foi desenvolvido com base nesse robô. Na versão anterior dele, estava presente um conjunto de circuitos (com um microcontrolador) que gerenciava as operações de baixo nível. Além disso, estava presente um PC embarcado (executando o sistema Linux), que efetuava as operações de alto nível.

A equipe deste projeto propôs modificar o robô Bellator para efetuar o mapeamento 2D de ambientes controlados como, por exemplo, labirintos construídos para fins de teste do robô. Posteriormente, em trabalhos futuros, ajustes finos poderão ser feitos para o uso em ambientes diversos, como escritórios, salas e quartos.

Na versão anterior do Bellator, estavam sendo utilizadas duas placas de circuito impresso – uma integrada com o microcontrolador e uma para a interface com os sensores – ambas ligadas por cabos entre si. Ao invés de produzir uma terceira placa para sensores adicionais (aspecto explicado mais à frente), o que aumentaria a quantidade de cabos, foi proposto o desenvolvimento de uma nova placa que realizasse a função de interface com todos os sensores e que fosse acoplada ao microcontrolador. Este microcontrolador pode ser usado diretamente na forma encapsulada de circuito integrado (soldado diretamente na nova placa), ou integrado a um kit de desenvolvimento (acoplado como *shield* na nova placa).

O sistema embarcado do robô desenvolvido pela equipe consiste na placa de interface de sensores acoplada ao microcontrolador. Esse sistema realiza as funções de baixo nível, ou seja, leitura de sensores e controle do PWM dos motores. A estação base é um computador, provido de um software que efetua comunicação bidirecional com o robô. A estação é capaz de enviar comandos de movimentação (especificados manualmente pelo teclado) a ele, além

de receber imagens da câmera e leituras dos sensores. No software, a partir das leituras dos sensores, é produzido um mapa em 2D simplificado do ambiente, com os obstáculos que forem detectados à medida que o robô anda, além do caminho estimado percorrido por ele. Protocolos de comunicação são utilizados entre: circuito de baixo nível e o PC embarcado (através de porta serial), e entre PC embarcado e estação base (através de conexão WI-FI). A placa com sistema linux embarcado foi mantida. Ela serve de interface entre a estação-base e a placa de controle embarcada e possui duas entradas USB. Em uma delas foi colocado um dispositivo de comunicação WI-FI, uma vez que a conexão entre a estação base e o robô têm um alcance de até 20 metros e a tecnologia WI-FI se mostra adequada para a comunicação dentro desse limite de distância. À outra porta foi acoplada uma webcam de forma a permitir que o usuário na estação base acompanhe a movimentação do robô à distância.

1.2 TRABALHOS CORRELATOS

O mapeamento de ambientes realizado por robôs visa ao desenvolvimento de software e hardware que permitam a construção de um mapa a partir de dados captados por um ou mais sensores. Há diversas tecnologias que podem ser empregadas para alcançar tal objetivo, como o processamento de digital de imagens captadas de uma câmera ou a utilização de sensores de proximidade tais como sensores de ultrassom ou sensores de ondas eletromagnéticas.

Esta última opção mostra-se bastante adequada para a maioria dos projetos, pois garante uma medição satisfatória da distância de objetos próximos ao robô a um custo não muito elevado. Um dos sensores mais populares deste tipo é o sensor de proximidade de infravermelho. Quando integrado ao robô permite a obtenção várias medidas discretas da distância do robô a objetos, um dos elementos básicos que permitem a geração do mapa do ambiente.

1.2.1 PatrolBot

O PatrolBot (BOT, 2013) é um robô configurável desenvolvido com interesses comerciais. Ele pode criar uma planta do interior de construções. Utilizando a tecnologia WI-FI ele pode ser controlado remotamente ou se movimentar de forma autônoma, sendo apenas monitorado pela estação base. Tal comunicação é feita por Wi-Fi.

Ele permite a inclusão de acessórios adicionais tais como uma câmera e microfones que permitirão ver e ouvir o que se passa no ambiente que está sendo mapeado. Há diversos outros periféricos que podem ser incluídos no robô, que também oferece a opção de ser programado, por meio de um kit de desenvolvimento próprio.

1.2.2 Sistema de mapeamento robótico bidimensional por infravermelho

Nesta implementação (SATO, 2013), um telêmetro infravermelho é utilizado para obter a distância de objetos próximos. A câmera infravermelha do Nintendo Wii é utilizada juntamente com sinalizadores (LEDs) para triangular a posição do robô e sua direção.

Como hardware, foram utilizados: um Arduino Mini Pro, um cartão microSD, telêmetro infravermelho, câmera do Wii. Não há comunicação em tempo real com a estação base. Isto significa que os dados são obtidos e armazenados no cartão microSD. Para leitura, o cartão deve ser inserido em um computador e, então, carregado na estação base. Os resultados são visualizados em um arquivo textual simples, no qual a letra "O" simboliza a posição do robô e a letra "X" os obstáculos detectados.

1.3 OBJETIVOS

- Implementar um software para comunicação de uma estação base (computador) com o robô, de forma que ela possa enviar comandos de movimentação ao robô, além de receber imagens da câmera e leituras dos sensores. Os comandos de movimentação (mover para frente, para trás, girar para esquerda/direita, parar) serão especificados por um utilizador humano através do teclado da estação base.
- O meio de comunicação entre a estação base e o robô deverá ter alcance máximo de 20 m (se não houverem paredes ou obstáculos entre a estação base e o robô). Para isso a tecnologia WI-FI mostra-se adequada e, portanto, ela será utilizada.
- Inserir uma *webcam* USB no robô, de modo que imagens do ambiente possam ser transmitidas à estação base. O propósito das imagens será unicamente permitir a visualização (pelo usuário da estação base, em tempo real) do ambiente no qual o robô está localizado. A câmera será conectada na porta USB do computador embarcado, e a transmissão de imagens será feita pelo canal Wi-Fi entre a estação base e o robô (o mesmo canal utilizado para a transmissão de dados dos sensores e comandos de movimentação).
- Implementar, no software utilizado na estação base, a geração de uma mapa em 2D com o caminho estimado percorrido pelo robô e os obstáculos detectados pelo mesmo. Os obstáculos serão representados a partir dos pontos em que houve detecção pelos sensores.
- Instalar novos sensores (acelerômetro e giroscópio) para efetuar as medições de velocidade e posicionamento do robô com maior exatidão do que pode ser feito atualmente

com os *encoders*. Ambos os sensores serão posicionados na carcaça do robô. Caso discrepâncias de medição entre os *encoders*, acelerômetro e giroscópio sejam detectadas (por exemplo, em caso de escorregamento de rodas), atenuações de erros poderão ser feitas no *software* da estação base.

- Desenvolver uma placa de circuito impresso que realize a função de interface com os sensores e que seja acoplada ao microcontrolador. Este microcontrolador pode ser usado diretamente na forma encapsulada de circuito integrado (sendo soldado diretamente na nova placa) ou integrado a um kit de desenvolvimento (acoplado como *shield* na nova placa).
- Em caso de falha de comunicação entre o robô e a estação base, o robô deverá permanecer parado e aguardando a conexão ser reestabelecida.

2 PLANEJAMENTO

2.1 CRONOGRAMA

O cronograma pretendido para o projeto está presente no Anexo D.

2.2 ORÇAMENTO

Cada membro propôs gastar até R\$150 com recursos materiais. Portanto, o orçamento máximo do projeto será de R\$750. O orçamento inicial do projeto será de R\$600 reais, dando a possibilidade de uma margem de erro de R\$150 caso exista necessidade, totalizando um custo total de R\$ 750 para o projeto.

2.3 DELIVERABLES

Na Tabela 1 estão expostos os *deliverables* previstos ao longo do projeto.

Tabela 1: Relação dos entregáveis com seus respectivos responsáveis e prazos

Dia	Auxiliar de Gerenciamento	Deliverables
13/03/2013	Stefan Campana Fuchs	<ol style="list-style-type: none"> 1. Versões iniciais dos diagramas de casos de uso e de classes (estação base). 2. Versão inicial do diagrama em blocos (hardware). 3. Explicação inicial de cada bloco (hardware).
27/03/2013	Telmo Friesen	<ol style="list-style-type: none"> 1. Versão inicial do diagrama de casos de uso (software embarcado). 2. Versão inicial do diagrama de fluxo de dados (software embarcado). 3. Versão inicial dos diagramas de estados (sistema de comunicação). 4. Versão inicial da descrição das mensagens e codificações dos comandos (sistema de comunicação). 5. Versão inicial do diagrama elétrico/eletrônico (hardware).

10/04/2013	Ricardo Farah	<ol style="list-style-type: none">1. Diagrama de casos de uso e de classes (estação base).2. Diagrama de casos de uso (software embarcado).3. Diagrama em blocos (hardware).4. Explicação detalhada de cada bloco (hardware).5. Diagrama elétrico/eletrônico (hardware).
24/04/2013	Stefan Campana Fuchs	<ol style="list-style-type: none">1. Projeto da PCB (Printed Circuit Board) (hardware).2. Lista de componentes para confecção da PCB completa (hardware).3. Diagramas de estados (sistema de comunicação).4. Descrição das mensagens e codificações dos comandos (sistema de comunicação).5. Descrição do uso do Software (Manual do Usuário)6. Mapa de conexão da PCB com a alimentação e outros elementos (hardware).7. Guia de montagem (hardware).

3 EXECUÇÃO

3.1 ARQUITETURA GERAL

A arquitetura geral do software foi projetada com base nos objetivos expostos no início do documento. Inicialmente foi feito o desenvolvimento dos casos de uso e do diagrama de blocos, de forma a visualizar o funcionamento do sistema como um todo. O resultado está descrito nas seções a seguir.

3.1.1 Software da estação-base e do placa TS-7260

1. Visualização de mapa 2D na interface gráfica segundo os dados lidos dos sensores do robô. Representado pelo caso de uso: **“Mostrar mapa - UC1”**.
2. Gravação do mapa em um arquivo no disco rígido. Representado pelo caso de uso: **“Salvar mapa - UC2”**.
3. Leitura do mapa de um arquivo do disco rígido. Representado pelo caso de uso: **“Carregar mapa - UC3”**.
4. Leitura de informações dos sensores do robô. Representado pelo caso de uso: **“Ler amostras dos sensores - UC4”**.
5. Visualização de imagens da *webcam* do robô. Representado pelo caso de uso: **“Visualizar imagens da câmera - UC5”**.
6. Alteração pelo usuário da velocidade das rodas do robô. Representado pelo caso de uso: **“Alterar velocidade das rodas - UC6”**.
7. Solicitação de estabelecimento de conexão com o robô. **“Estabelecer conexão - UC7”**.
8. Consulta à documentação do robô pelo usuário. Representado pelo caso de uso: **“Consultar documentação do robô - UC8”**.

Foram produzidos três Diagramas de Casos de Uso (Figuras 1, 2 e 3) com base nos casos de uso apresentados. O primeiro diagrama representa o *software* da estação base, e o

segundo e o terceiro representam o sistema embarcado (TS-7260 e placa de baixo nível, respectivamente).

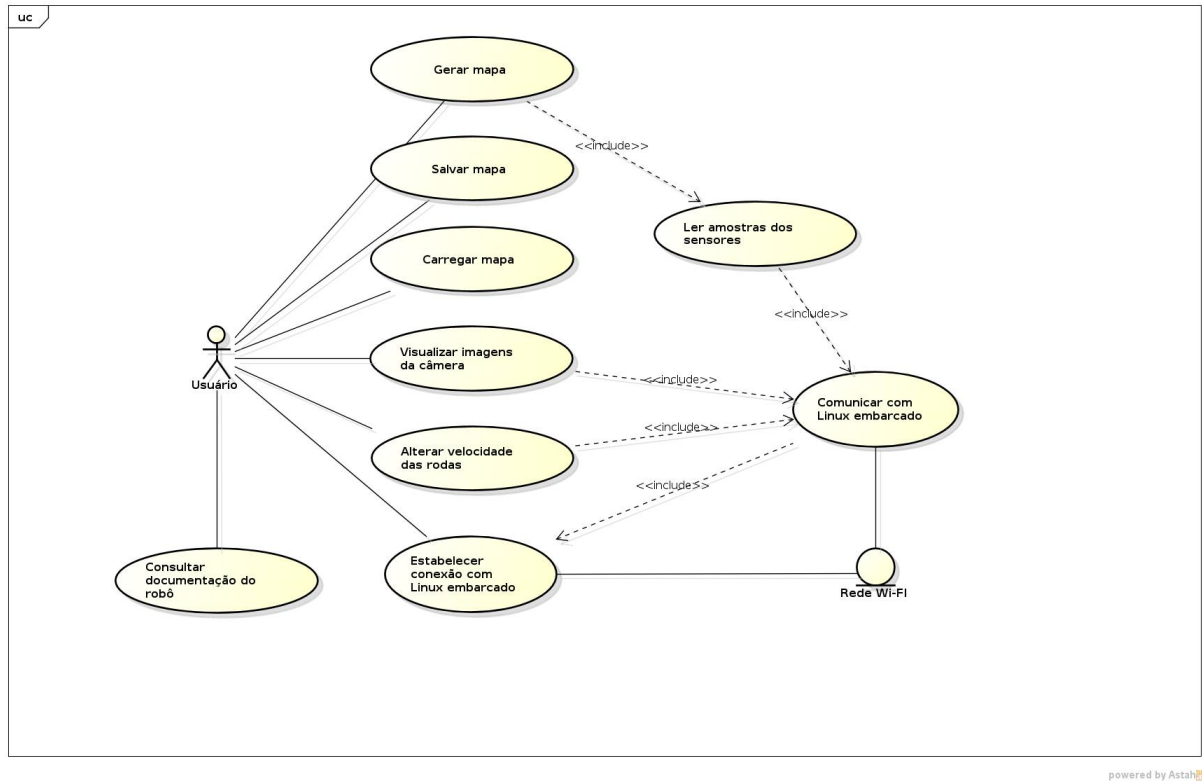


Figura 1: Diagrama de casos de uso do *software* da estação base.

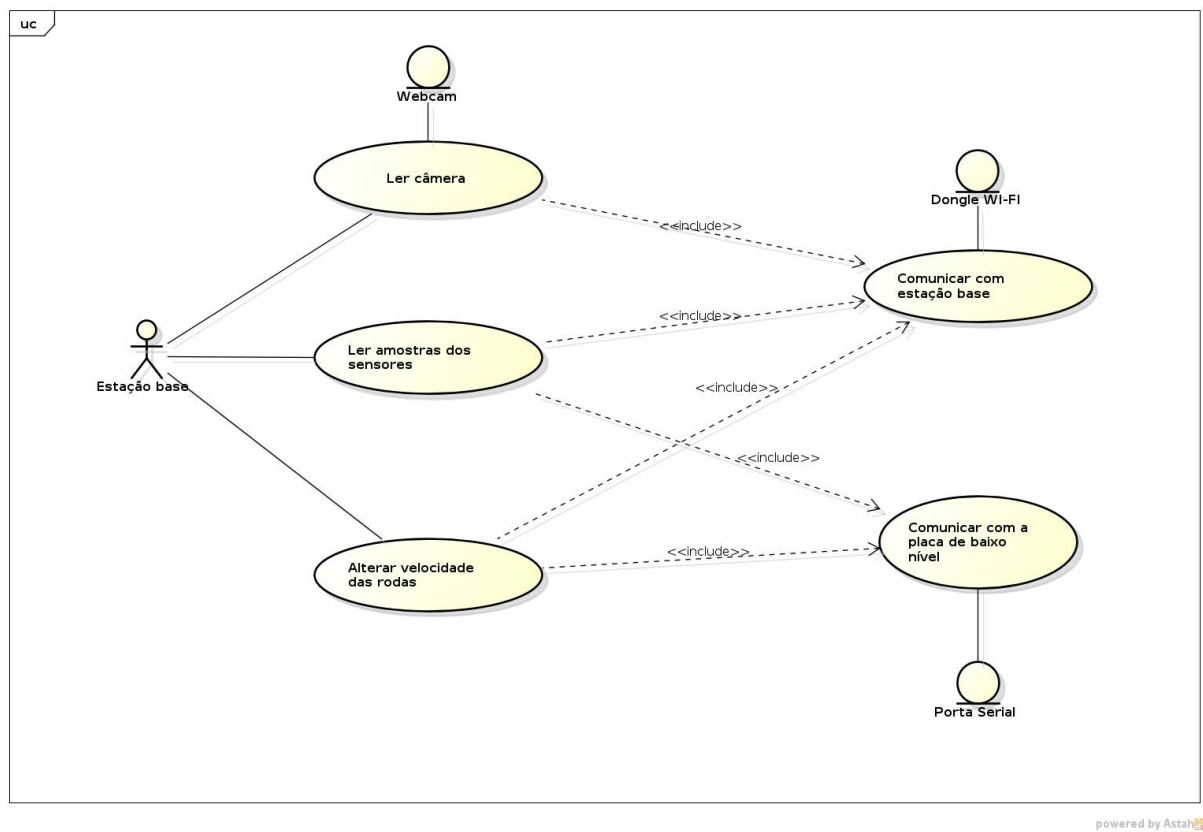


Figura 2: Diagrama de casos de uso do *software* para a placa TS-7260.

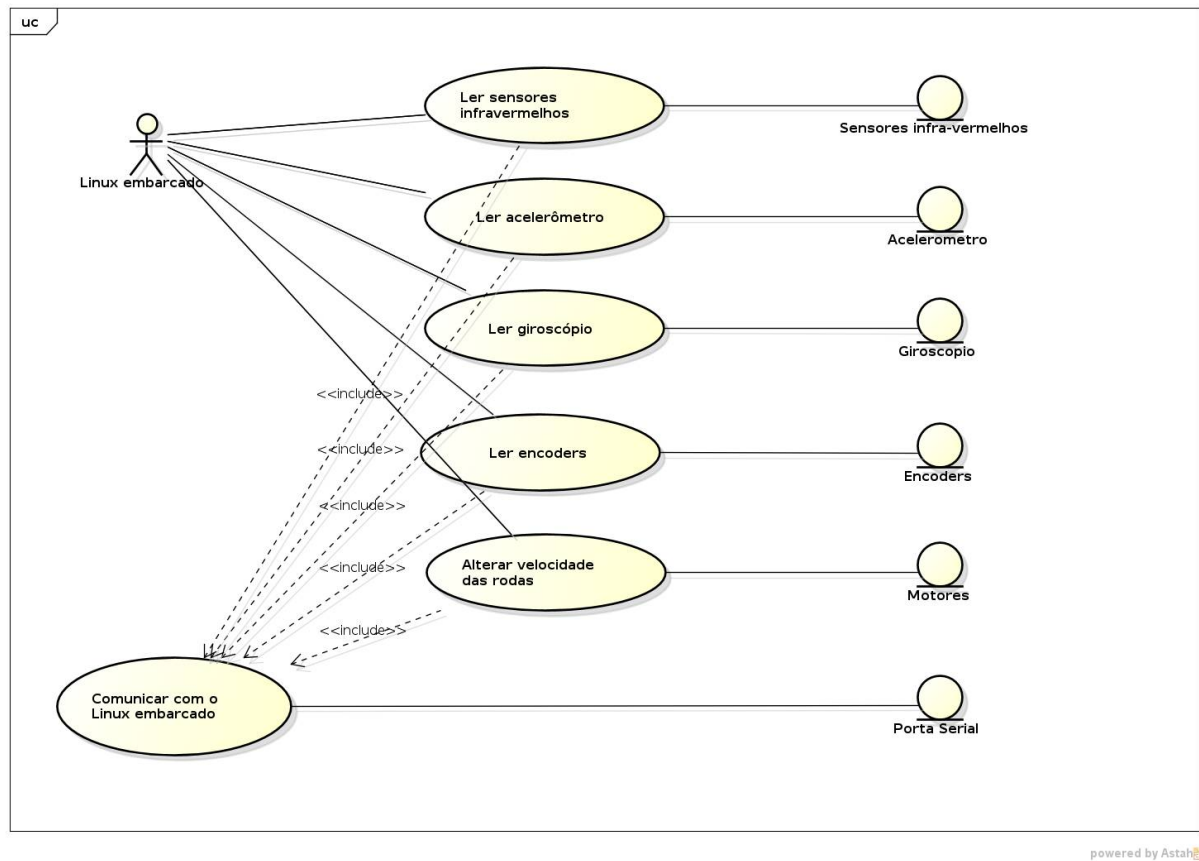


Figura 3: Diagrama de casos de uso do *software* para a placa de baixo nível.

3.1.2 Hardware

Na figura 4 mostra-se o diagrama de blocos do sistema embarcado e suas conexões com o restante do robô. A seguir está também uma descrição para cada um dos blocos da placa de circuito impresso do sistema embarcado.

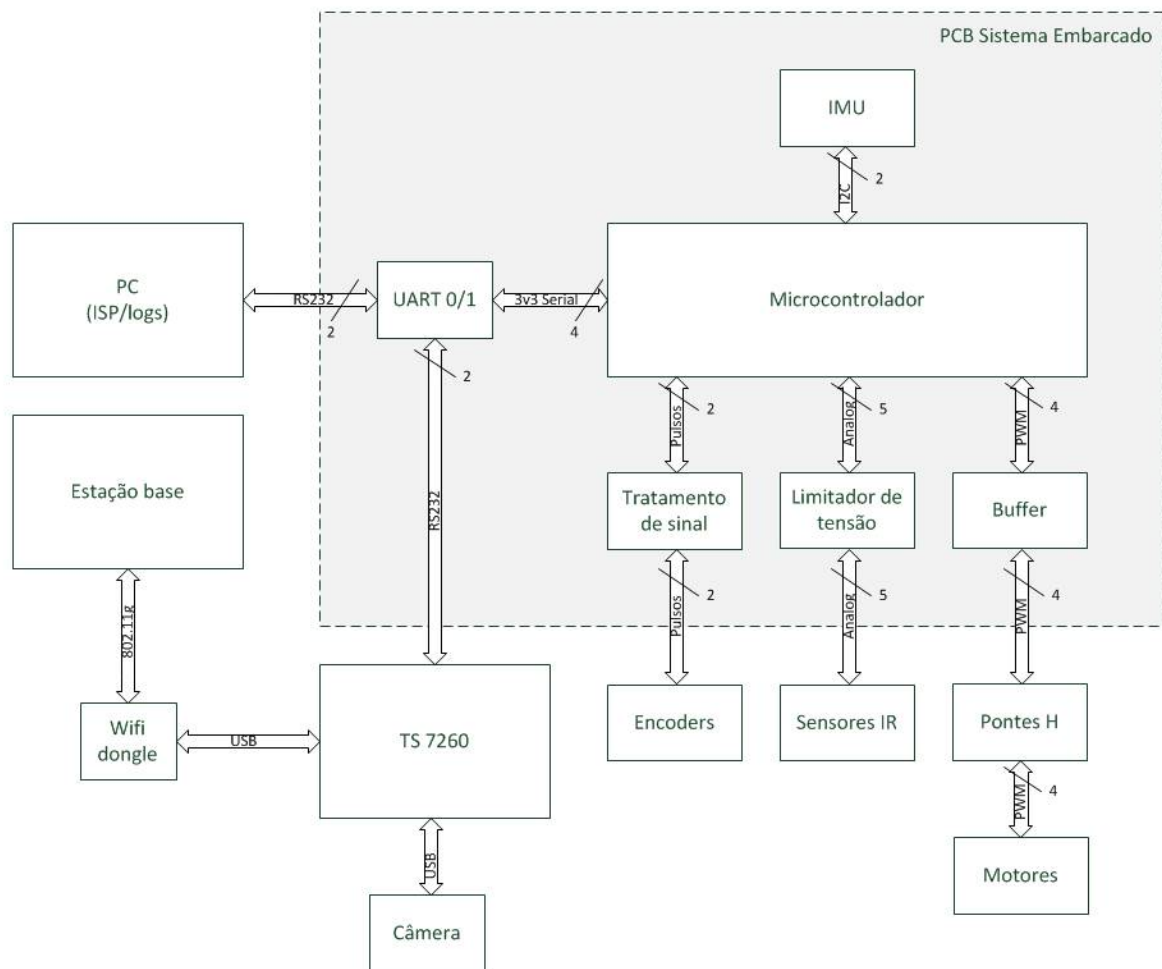


Figura 4: Diagrama de blocos do hardware

1. **Microcontrolador:** Este bloco fará a leitura dos sensores: encoders, infra-vermelhos, acelerômetro e giroscópios. Além disso possui a implementação do protocolo de comunicação para interação com o linux embarcado da placa TS-7260.
2. **UART 0/1:** Responsável por ajustar os níveis de tensão para comunicação serial no padrão RS-232 com a placa TS-7260.
3. **Buffer:** Responsável por fornecer corrente e elevar os níveis de tensão de saída do microcontrolador de 3,3V para 5,0V. Esse buffer é conectado às pontes H já existentes no robô.
4. **IMU (Inertial Measurement Unit):** possui o acelerômetro e o giroscópio e se comunicará com o microcontrolador por meio do protocolo I2C.
5. **Limitador de tensão:** Necessário pois os sinais de saída dos sensores de infravermelho que já existem no robô não estão limitados em 5V, podendo a saída ultrapassar 5,0V e danificar o microcontrolador.

6. Tratamento de sinal: Composto por um filtro RC passa baixas e um Schmitt trigger para remover qualquer falha que possa ocorrer na geração dos pulsos no encoder.

3.2 ESTAÇÃO-BASE

3.2.1 Diagrama de classes da estação base

A Figura 5 mostra o diagrama de classes da estação base.

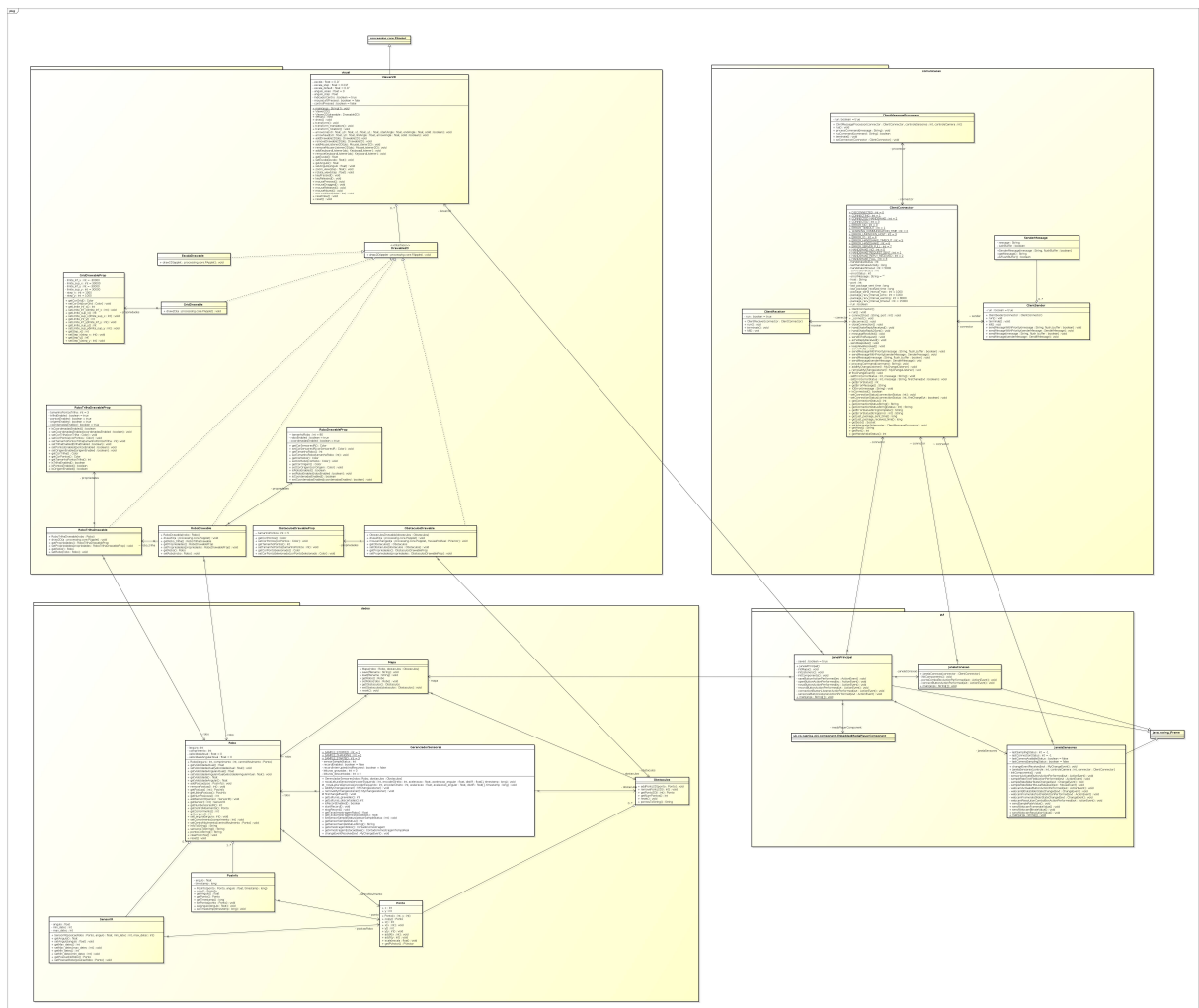


Figura 5: Diagrama de classes da estação base

3.2.1.1 Descrição das classes da estação base

O *software* da estação base do robô foi dividido em cinco pacotes: *visual*, *dados*, *comunicacao*, e *gui*. A seguir há uma descrição de cada pacote e das suas respectivas classes.

3.2.1.2 Pacote *visual*

Este pacote consiste de toda a parte visual da estação base e conta com as seguintes classes: Viewer2D, Drawable2D, EscalaDrawable, RoboDrawable, RoboTrilhaDrawable, ObstaculosDrawable, EscalaDrawableProp, RoboDrawableProp, RoboTrilhaDrawableProp e ObstaculosDrawableProp. Na Tabela 2 estão descritas as classes deste pacote.

Tabela 2: Pacote *visual*

Classe	Descrição
Viewer2D	Responsável por exibir os objetos Drawable2D. Possui recursos de pan, zoom e rotate.
Drawable2D	Representa genericamente objetos 2D que podem ser desenhados em um Viewer2D.
EscalaDrawable	Responsável por desenhar uma escala gráfica no mapa.
RoboDrawable	Responsável por desenhar o robô no mapa.
RoboTrilhaDrawable	Responsável por desenhar a trilha percorrida pelo robô no mapa.
ObstaculosDrawable	Responsável por desenhar os pontos de cada obstáculo no mapa.
EscalaDrawableProp	Contém as propriedades visuais de desenho da escala.
RoboDrawableProp	Contém as propriedades visuais de desenho do robô
RoboTrilhaDrawableProp	Contém as propriedades visuais de desenho da trilha do robô.
ObstaculosDrawableProp	Contém as propriedades visuais de desenho dos obstáculos.

3.2.1.3 Pacote *dados*

Este pacote consiste de toda a parte da estação base que processa e armazena as informações essenciais do robô e do mapa. Conta com as seguintes classes: Mapa, Obstaculos,

Robo, ControleSensores, Posinfo, SensorIR e Ponto. Na Tabela 3 estão descritas as classes deste pacote.

Tabela 3: Pacote *dados*

Classe	Descrição
Mapa	Responsável por representar o mapa. Armazena as informações essenciais do robô e dos obstáculos detectados.
Obstaculos	Responsável por conter os obstáculos detectados pelo robô.
Robo	Responsável por representar o robô, este contém largura, comprimento e centro de movimento (ponto central entre as duas rodas).
GerenciadorSensores	Responsável por atualizar a posição do robô e dos pontos que representam os obstáculos, de acordo com as leituras feitas pelos sensores.
Posinfo	Responsável por conter as informações de uma posição do robô.
SensorIR	Responsável por representar um sensor IR do robô.
Ponto	Representa um ponto de coordenadas cartesianas (x,y).
GerenciadorCamera	Responsável por gerenciar o status da câmera e o recebimento de imagens.

3.2.1.4 Pacote *comunicacao*

Este pacote consiste em toda a parte de comunicação da estação base com o robô e conta com as seguintes classes: ClientMessageProcessor ClientConnection, ClientReceiver, ClientSender e Message. Na Tabela 4 estão descritas as classes deste pacote.

É importante ressaltar que o protocolo TCP requer obrigatoriamente a especificação de um cliente e de um servidor para estabelecimento de uma conexão. Nas implementações desse protocolo em diversas linguagens (como Java e C++) existem tipos de *socket* distintos

para cliente e servidor. Na criação de um *socket* de servidor, há obrigatoriamente a atribuição de uma porta de escuta, na qual o servidor aguarda que um cliente efetue uma requisição de conexão. Não é possível, ao menos nas implementações atuais do TCP, estabelecer conexão entre dois *sockets* de cliente ou entre dois *sockets* de servidor. Como neste projeto, o robô proverá serviços à estação base (envio de imagens da câmera, envio de leituras de sensores, além de prover a possibilidade de comando dos motores) o robô foi escolhido como servidor e a estação base como cliente. Enfatiza-se que o paradigma cliente-servidor não implica de forma alguma que a comunicação seja unidirecional. Pelo contrário, o envio de pacotes pode ser feito bidirecionalmente após uma conexão TCP ser estabelecida, sem nenhuma restrição quanto a isso.

Tabela 4: Pacote *comunicacao*

Classe	Descrição
ClientMessageProcessor	Thread responsável pelo processamento de mensagens recebidas de um host de conexão.
ClientConnector	Thread responsável por efetuar a gerência da conexão do cliente (estação base) com o servidor (robô).
ClientReceiver	Thread responsável por receber mensagens de um host de uma conexão.
ClientSender	Thread responsável por enviar mensagens ao host de uma conexão.
SenderMessage	Contém uma mensagem que pode ser enviada por um Sender.

3.2.1.5 Pacote *gui*

Este pacote consiste em toda a interface gráfica do sistema e conta com as seguintes classes: *JanelaConexao*, *JanelaPrincipal* e *JanelaSensores*. Na Tabela 5 estão descritas as classes deste pacote.

Tabela 5: Pacote *gui*

Classe	Descrição
JanelaConexao	Janela com as informações e configurações da conexão com o Bellator.
JanelaPrincipal	Janela principal da interface gráfica da estação base.
JanelaSensores	Janela de configuração dos sensores.

3.3 SISTEMA DE COMUNICAÇÃO

3.3.1 Codificação das mensagens

- Mensagens do TS-7260 para o LPC2103 (via porta serial)

- **SYNC (0xA0)**

Quando o microcontrolador LPC2103 recebe esta mensagem, responde com as leituras mais recentes dos encoders, de cada sensor de distância, do acelerômetro e do giroscópio (enviando uma mensagem SENSORS, explicada abaixo).

- **ENGINES (0xB0)**

(byte) vel_roda_esquerda

(byte) vel_roda_direita

(byte) CHECKSUM_H

(byte) CHECKSUM_L

Ao receber este comando, o microcontrolador utiliza os valores para definir o nível de PWM para as rodas do robô. Os valores de velocidade são representados por um byte cada, nos quais o bit mais significativo indica o sentido de rotação da roda (1 para frente e 0 para trás) e os restantes a intensidade do PWM.

Os bytes de checksum são utilizados para verificar se não há dados corrompidos. Os bytes da mensagem são somados (módulo 65536) e o resultado é atribuído aos bytes (high e low) de checksum.

- Mensagens do LPC2103 para a TS-7260 (via porta serial)

- **ENGINES_ACK (0xB1)**

(byte) vel_roda_esquerda

(byte) vel_roda_direita

(byte) CHECKSUM_H

(byte) CHECKSUM_L

Esta mensagem deve ser enviada toda vez que um comando de mudança de velocidade (ENGINES) for recebido na placa de baixo nível. A mensagem é usada no Linux embarcado para verificar se o comando foi corretamente recebido. Caso uma confirmação não seja recebida em certo intervalo de tempo, outro comando ENGINES é enviado para a placa de baixo nível.

O checksum tem função idêntica ao que já foi explicitado na mensagem ENGINES.

– SENSORS (0xC0)

(byte) encoder_esq_H, (byte) encoder_esq_L,

(byte) encoder_dir_H, (byte) encoder_dir_L,

(byte) IR1, (byte) IR2, (byte) IR3, (byte) IR4, (byte) IR5,

(byte) AX_H, (byte) AX_L,

(byte) AY_H, (byte) AY_L,

(byte) AZ_H, (byte) AZ_L,

(byte) GX_H, (byte) GX_L,

(byte) GY_H, (byte) GY_L,

(byte) GZ_H, (byte) GZ_L,

(byte) TIMESTAMP_H, (byte) TIMESTAMP_L

(byte) CHECKSUM_H

(byte) CHECKSUM_L

Representa a leitura de todos os sensores (encoders, infra-vermelhos, acelerômetro e giroscópio).

Os 4 primeiros bytes são os valores das leituras dos encoders esquerdo e direito (cada um com um byte alto e um baixo). Os valores das leituras dos encoders representam a diferença entre a contagem atual a contagem anterior.

Nos próximos 5 bytes, as leituras do sensores ópticos são enviadas em sequência. As distâncias que os sensores ópticos são capazes de mensurar são divididos em valores discretos de 0 a 255 (MARIN et al., 2012).

Após isso, os 12 bytes que se seguem representam as leituras do acelerômetro e do giroscópio. Os bytes que começam com ‘A’ representam a leitura de cada um dos eixos do acelerômetro. Aqueles que começam com ‘G’ representam a leitura de cada um dos eixos do giroscópio.

O timestamp (valor alto e baixo) é um contador de 16 bits que é incrementado entre

cada amostra e zera automaticamente depois que chega ao valor máximo (65535), usado para determinar o instante em que foi feita a leitura dos dados. Como a amostragem dos sensores na placa de baixo nível será efetuada em intervalos fixos, a informação do contador do timestamp pode ser utilizada para obter informações de tempo de cada amostra.

O checksum é tem função idêntica ao que já foi explicitado na mensagem ENGINES.

- Mensagens bidirecionais entre estação base e TS-7260 (via WI-Fi):

- **ECHO_REQUEST (0x01)**

(byte) END_CMD

Requisição de ping.

- **ECHO_REPLY (0x02)**

(byte) END_CMD

Resposta de ping.

- **DISCONNECT (0x0F)**

Solicitação de desconexão.

- Mensagens da estação base para a TS-7260 (via Wi-Fi):

- **HANDSHAKE_REQUEST (0x10)**

Solicitação de handshake.

- **HANDSHAKE_CONFIRMATION (0x12)**

Confirmação de handshake.

- **SENSORS_START (0x20)**

Solicitação de início da amostragem dos sensores.

- **SENSORS_STOP (0x21)**

Solicitação de parada da amostragem dos sensores.

- **SENSORS_RATE (0x22)**

(float) Nova taxa de amostragem (comandos SYNC por segundo)

Solicitação de mudança da taxa de envio de comandos SYNC da TS para a placa de baixo nível.

- **WEBCAM_START (0x30)**

Solicitação de início da amostragem da webcam.

– **WEBCAM_STOP (0x31)**

Solicitação de parada da amostragem da webcam.

– **WEBCAM_RATE (0x32)**

(float) Nova taxa de quadros

Solicitação de mudança da taxa de quadros da webcam.

– **WEBCAM_RESOLUTION (0x33)**

(int) Largura em pixels

(int) Altura em pixels

Solicitação de mudança da resolução da webcam.

– **ENGINES (0xB0)**

(float) vel_roda_esquerda

(float) vel_roda_direita

Solicitação de mudança da velocidade dos motores. Para cada roda há um valor de -1 até 1, sendo que -1 é a máxima velocidade para trás, 1 a máxima velocidade para frente e 0 é parada da roda.

• Mensagens da TS-7260 para a estação base (via Wi-Fi):

– **HANDSHAKE_REPLY (0x11)**

Resposta de handshake.

– **SENSORS (0xC0)**

(byte) encoder1_H, *(byte)* encoder1_L,

(byte) encoder2_H, *(byte)* encoder2_L,

(byte) IR1, *(byte)* IR2, *(byte)* IR3, *(byte)* IR4, *(byte)* IR5,

(byte) AX_H, *(byte)* AX_L,

(byte) AY_H, *(byte)* AY_L,

(byte) AZ_H, *(byte)* AZ_L,

(byte) GX_H, *(byte)* GX_L,

(byte) GY_H, *(byte)* GY_L,

(byte) GZ_H, *(byte)* GZ_L,

(byte) TIMESTAMP_H, *(byte)* TIMESTAMP_L

Possui a mesma funcionalidade e parâmetros que a mensagem SENSORS enviada da LPC2103 para a TS-7260, com exceção do timestamp, que é trocado por um timestamp UNIX em milissegundos (que representa o horário absoluto em que a

amostra foi obtida na placa de baixo nível). Essa informação de tempo é utilizada pela estação base para efetuar os cálculos de posicionamento do robô.

– **SENSORS_STATUS (0xC1)**

(boolean) Status da amostragem [on - off]

(float) Taxa de amostragem

Informações de status dos sensores. Usado na interface gráfica para confirmar o recebimento de comandos de mudança de taxa de amostragem e início/parada da amostragem.

– **WEBCAM_STATUS (0x34)**

(float) Taxa de quadros

(int) Largura em pixels

(int) Altura em pixels

(boolean) Status da stream [on - off]

(int) Porta da stream

Informações de status da webcam. Usado na interface gráfica para confirmar o recebimento de comandos relativos à webcam, e para que a estação base tenha conhecimento do status da stream da webcam.

– **ENGINES_STATUS (0xB1)**

(byte) vel_roda_esquerda

(byte) vel_roda_direita

Informações sobre as velocidades programadas dos motores. Usado na interface gráfica para confirmar o recebimento de comandos de movimentação efetuados pelo usuário.

3.4 SISTEMA EMBARCADO

3.4.1 Diagram de classes do sistema embarcado

A Figura 6 mostra o diagrama de classes do sistema embarcado (placa TS-7260).

Figura 6: Diagrama de classes do sistema embarcado (placa TS-7260).

Tabela 6: Descrição das classes do sistema embarcado (placa TS-7260)

Classe	Descrição
Main	Classe principal do robô.
SensorsSampler	Thread responsável por requisitar amostras dos sensores da placa de baixo nível em intervalos de tempo previamente programados.
ServerMessageProcessor	Thread responsável por realizar o processamento de mensagens recebidas de um host de conexão.
ServerListener	Thread responsável por escutar requisições de conexão.
ServerSender	Thread responsável por enviar mensagens ao host de uma conexão.
SenderMessage	Contém uma mensagem que pode ser enviada por um Sender.
ServerReceiver	Thread responsável por receber mensagens de um host de uma conexão.
SerialCommunicator	Responsável por gerenciar a comunicação via porta serial entre a TS-7260 e a LPC2103.
WebcamManager	Responsável por gerenciar a abertura e o fechamento da stream de imagens da webcam, além de configurar opções como resolução e taxa de frames.

3.5 CRONOGRAMA ORIGINAL VERSUS EXECUTADO

O cronograma original (Anexo D), foi executado conforme o esperado. Devido à antecipação de certas tarefas, eventuais atrasos ficaram dentro do planejado. Os deliverables foram entregues dentro dos prazos estipulados.

3.6 ORÇAMENTO ORIGINAL VERSUS EXECUTADO

Como pode ser visto no orçamento em anexo, o custo total do projeto ficou em 899 reais, o que ultrapassou a inicial máxima em cerca de 18,9%. Isso se deu, principalmente devido à ocorrência do risco de “Taxação dos componentes comprados no exterior”, o que elevou signif

4 CONCLUSÕES

4.1 ANÁLISE DO DESENVOLVIMENTO

Do ponto de vista dos objetivos, percebe-se que a maioria deles foram alcançados. Foi implementado com sucesso um *software* com interface gráfica para geração e visualização de mapas, visualização de imagens da *webcam* do robô e controle do robô por meio do teclado. A comunicação foi implementada satisfatoriamente, tanto entre a estação base e a placa TS-7260 (via Wi-Fi) quanto entre a TS e a placa de baixo nível, via porta serial. Uma *webcam* USB foi instalada e configurada com sucesso no robô para a transmissão de imagens ao usuário. O giroscópio foi utilizado satisfatoriamente na prática para corrigir erros de leituras dos encoders e escorregamento das rodas. Uma placa de circuito impresso, de tamanho reduzido, foi desenvolvida de forma a integrar as funções de baixo nível do robô: interface com os encoders, sensores infra-vermelhos, acelerômetro e giroscópio, e geração de PWM para os motores.

Observou-se que três riscos previstos no planejamento de riscos ocorreram. O primeiro foi a “Não entrega de componentes ou entrega fora do prazo”. Uma unidade MPU-6050 (acelerômetro e giroscópio) encomendada no site eBay não foi entregue da maneira especificada e foi devolvida. Como a encomenda desse item havia sido adiantada, realizou-se uma nova encomenda no site Sparkfun, que chegou no prazo estabelecido não causando atraso no projeto. O segundo risco foi “Taxação dos componentes comprados no exterior”, que gerou um aumento no preço dos componentes, mas não de forma que ultrapassasse o orçamento especificado. O terceiro e último risco foi “Problemas Técnicos com o Hardware”. Como relatado, a imprecisão da leitura dos encoders ópticos interferiu na correta determinação da posição do robô. Como solução, buscou-se aumentar a aderência das rodas e correias do robô a fim de diminuir os escorregamentos e falsas leituras, além de utilizar o giroscópio para correção do posicionamento.

Em termos de déficits de conhecimento, pode-se citar a dificuldade em se utilizar o acelerômetro da placa, devido a seus problemas de leitura. Idealmente, o filtro de Kalman poderia ser utilizado para produzir melhores estimativas do posicionamento do robô, houve

falta de tempo e base acadêmica para implementação desse filtro, que, na prática mostrou-se complexo de complexa interpretação e aplicação.

Por último, devido a um bom planejamento do cronograma, da execução da equipe e da baixa ocorrência de riscos, não houve atrasos dentro do cronograma planejado. As atividades foram concluídas dentro do prazo previsto pela equipe.

4.2 INTEGRAÇÃO

Para o desenvolvimento do projeto, da matéria de Oficinas de Integração 3, foram necessários diversos conhecimentos adquiridos durante o curso. Foram utilizados, principalmente, os conhecimentos nas áreas de Eletrônica Geral, Programação, Engenharia de Software e Comunicação de Dados.

O projeto foi dividido em três seções cada uma com sua gama de conhecimentos específicos:

Planejamento do Projeto: Nesta seção foram utilizados conhecimentos como Gerenciamento de Risco, Custo, Cronograma, Análise Tecnológica, entre outros, adquiridos nas matérias de Análise de Projeto de Sistemas, Engenharia de Software, Oficinas de Integração 1 e 2. Esses conhecimentos foram aplicados para criar o cronograma do projeto considerando a possibilidade de riscos e os custos e trabalho necessário para concluí-lo.

Hardware: Nesta seção foram utilizados conhecimentos como Microcontroladores, Sensores, Filtros, Sinais, Circuitos Digitais, entre outros, adquiridos nas matérias de Eletrônica, Microcontrolados, Controle, Sinais e Sistemas. Esses conhecimentos foram aplicados no desenvolvimento e na confecção da placa de circuito impresso, no uso e controle do robô, etc.

Software: Nesta seção foram utilizados conhecimentos como Programação, Comunicação de Dados, Estrutura de Dados. Esses conhecimentos foram aplicados para criar o programa da estação base, assim como possibilitar toda a estrutura de comunicação entre as partes do projeto.

4.3 TRABALHOS FUTUROS

Há inúmeros modos de se prosseguir com o projeto do robô Bellator, pois muitas melhorias podem ser efetuadas a fim de se atingir resultados mais precisos em termos de mapeamentos de ambientes.

No que diz respeito ao sensoramento, é importante melhorar a confiabilidades dos

dados dos encoders. Uma das formas de se fazer isso é utilizando um *timing belt*, que é uma roda dentada acoplada a um sensor. Dessa forma, problemas de escorregamento da correia em relação à polias do robô seriam eliminados.

Quanto ao uso do acelerômetro, a técnica do filtro de Kalman poderia ser explorada se forma a se realizar a fusão de dados dos três sensores do robô: encoder, acelerômetro e giroscópio. Se desenvolvida corretamente, em tese, a utilização desta técnica permitiria a compensação automática de erros dos sensores o que permitiria uma estimativa muito melhor do posicionamento do robô. As dificuldades de se utilizar esta técnica são: correta determinação da margem de erro de cada sensor, criação de um modelo matemático da movimentação do robô e complexidade do entendimento e implementação do filtro.

Em relação à câmera, há dois aspectos a serem melhorados: a redução do atraso do sinal de vídeo (que é da ordem de segundos) e a interação do usuário com a câmera. O primeiro problema poderia ser resolvido com a substituição da *webcam* atual por outra que possua menos latência, ou pelo desenvolvimentos de software que faça a leitura dos dados brutos da câmera e faça o envio instantâneo desses dados à estação base, procurando reduzir ou eliminar a utilização de *buffers* de vídeo. Em termos de interatividade, pode-se construir um dispositivo de rotação que permita ao usuário girar a câmera 360 graus em torno do eixo vertical, possibilitando maior flexibilidade e autonomia durante a utilização do robô.

APÊNDICE A – PLANEJAMENTO DE RISCOS

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Desistência de Membros da Equipe ou Redução do número de integrantes da equipe por força maior	Nº Identificação 01
Descrição do risco: Possibilidade da desistência de membros da equipe, seja de um único membro até todos os cinco.	

2º Etapa: Avaliação do Risco

Impacto: <input checked="" type="radio"/> 5 (alta) <input type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: Caso o risco venha a ocorrer, o cronograma e a qualidade do projeto serão afetados. Toda a carga de trabalho deverá ser redividida e o cronograma refeito. Dependendo da quantidade de trabalho restante, o escopo do projeto terá que ser reavaliado. Caso muitos membros desistam e a carga de trabalho remanescente seja muito grande para completar o projeto, caberá aos membros documentar o que foi feito e tentar terminar o que for possível para que uma outra equipe de continuidade em outra ocasião.
Probabilidade: <input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: Os integrantes da equipe terão, na medida do possível, suas tarefas e cronogramas definidos de forma que não haja sobrecarga de trabalho (levando em conta, conjuntamente, as outras matérias e/ou compromissos). A motivação da equipe será também levada bastante em consideração.

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade):	
Prevenir: O gerente fará reuniões constantes do grupo para análise da contribuição e desempenho de cada integrante. O gerente também monitorará o andamento do projeto, efetuando ajustes no ritmo de trabalho (caso necessário) para que não haja sobrecarga de trabalho para um único membro da equipe. Todos na equipe deverão se comunicar frequentemente.	
Mitigar: Caso haja desistência de algum membro da equipe, a carga de trabalho deve ser reajustada entre os remanescentes. O cronograma deve ser modificado para que as atividades restantes sejam acomodadas dentro do possível para cada membro. Caso não haja possibilidade de realizar todas as atividades no tempo disponível, deve-se reanalisar o escopo do projeto e direcionar esforços nas áreas mais importantes. Caso muitos membros desistam, consequentemente impossibilitando a completude do trabalho, caberá aos remanescentes documentar o que foi realizado até o momento para que outras equipes, caso venham a dar continuidade ao projeto, possam obter máximo de informações possíveis.	
Impacto reavaliado: 3	Probabilidade reavaliada: 2
Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013
<input type="checkbox"/> Respostas incluídas na WBS/Cronograma	
Registros adicionais: Verso ou Anexos	

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: **Identificação do Risco**

Denominação do risco: Problemas Técnicos com o <i>Hardware</i>	Nº Identificação 02
Descrição do risco: Possibilidade do hardware apresentar problemas	

2º Etapa: **Avaliação do Risco**

Impacto:	<input type="radio"/> 5 (alta) <input checked="" type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: Dependendo do tipo de problema apresentado pelo <i>hardware</i> , reparos poderão ser realizados sem grandes alterações no cronograma, qualidade ou custo do projeto. Caso o problema seja de difícil identificação ou se for requerida reposição de uma peça, o fato deverá ser levado em consideração para que sejam alocados os recursos necessários para a realização do reparo.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: Problemas com peças de <i>hardware</i> são comuns em diversos aparelhos eletrônicos. Como o projeto apresenta vários componentes eletrônicos diversos, há considerável probabilidade de ocorrerem falhas.	

3º Etapa: **Desenvolvimento da Resposta ao Risco**

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade): <p>Prevenir: Manter um estoque de peças reserva para a eventualidade de alguma falhar. Manter contato com professores que possam auxiliar caso o problema seja de difícil identificação ou conserto. Compra de peças de qualidade. Cautela ao manusear o hardware. Manter documentação com alterações e testes no hardware para facilitar identificação caso problemas ocorram. Manter vídeos do hardware funcionando caso ele venha a falhar nas demonstrações do projeto.</p> <p>Mitigar: Caso algum problema técnico venha acontecer, os membros da equipe devem primeiro tentar identificar o problema e consertá-lo. Caso não seja possível identificar ou consertar de maneira rápida, serão alocados os recursos necessários para que os reparos sejam feitos pelos membros da equipe ou por um terceiro especializado.</p>	
Impacto reavaliado: 3	Probabilidade reavaliada: 2
Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013 <input type="checkbox"/> Respostas incluídas na WBS/Cronograma
Registros adicionais: Verso ou Anexos	

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Possibilidade do <i>software</i> apresentar problemas.	Nº Identificação 03
Descrição do risco: Problemas decorrentes da programação ou dificuldades técnicas relacionadas ao programa desenvolvido	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input type="radio"/> 4 (média/alta) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: Dependendo do problema apresentado pelo <i>software</i> , o mesmo poderá ser reparado sem grandes alterações no cronograma, qualidade ou custo do projeto. Caso o problema seja de difícil identificação e conserto, o fato deverá ser informado ao gerente para que sejam alocados os recursos necessários (tempo, dinheiro, etc...) para que o conserto seja realizado.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (médio/baixa) <input type="radio"/> 1 (baixa)
Explique: Problemas com <i>software</i> são relativamente comuns de ocorrerem. Como muitas vezes não é possível realizar testes extensivos para garantir a corretude do programa, há probabilidade mediana de serem encontrados <i>bugs</i> ao ser utilizado o <i>software</i> .	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade):	
Prevenir: Manter um repositório de versões do software, com adições, remoções, testes e correções documentadas. Realizar backup do software entre os membros da equipe e em repositórios online.	
Mitigar: Caso algum problema técnico venha acontecer, os membros da equipe devem primeiro tentar identificar o problema e repará-lo. Caso não seja possível identificar ou consertar de maneira rápida, serão alocados os recursos necessários para que o conserto seja feito pelos membros da equipe ou um terceiro especializado. Em caso de urgência, utilizar a versão funcional mais atual. Ter em vídeo o software detalhado e funcionando também é uma medida pertinente no caso de haver falhas na apresentação do projeto.	
Impacto reavaliado: 2	Probabilidade reavaliada: 2
Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013
<input type="checkbox"/> Respostas incluídas na WBS/Cronograma	
Registros adicionais: Verso ou Anexos	

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Não entrega de componentes ou entrega fora do prazo.	Nº Identificação 04
Descrição do risco: Alguns dos componentes a serem utilizados para o desenvolvimento do projeto deverão ser comprados no exterior por serem inexistentes no Brasil ou muito mais caros de se comprar por aqui. Importar componentes gera incerteza em relação ao prazo de entrega.	

2º Etapa: Avaliação do Risco

Impacto: <input checked="" type="radio"/> 5 (alta) <input type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo) Explique: Sem os componentes não é possível montar o <i>hardware</i> do projeto.
Probabilidade: <input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input type="radio"/> 3 (média) <input checked="" type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa) Explique: Lojas internacionais raramente deixam de enviar o produto comprado. Se houver algum problema de entrega, é mais provável que ocorra devido às transportadoras ou demora na liberação do produto pelas alfândegas brasileiras.

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade): Prevenir: Comprar de lojas confiáveis no exterior, que apresentem boa reputação e de preferência que já tenham sido utilizadas com sucesso por membros da equipe no passado. Comprar componentes com bastante antecedência à sua utilização para evitar que possíveis atrasos na entrega sejam danosos ao projeto. Mitigar: Pesquisar lojas nacionais que ofereçam os mesmos componentes caso haja atrasos ou problemas na entrega, de forma a minimizar os prejuízos ao cronograma do projeto.	
Impacto reavaliado: 4	Probabilidade reavaliada: 2

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013	<input type="checkbox"/> Respostas incluídas na WBS/Cronograma	Registros adicionais: Verso ou Anexos
--	------------------	--	--

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Taxação dos componentes comprados no exterior	Nº Identificação 05
Descrição do risco: Para compras de componentes em sites do exterior existe o risco de, na alfândega brasileira, ocorrer cobrança de imposto de importação e ICMS, o que pode encarecer o custo desses componentes em até 100%. Além disso, podem haver atrasos na entrega por conta da cobrança das taxas.	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input checked="" type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: A compra de componentes no exterior muito provavelmente não ultrapassará o valor limite do orçamento, mesmo com as taxas de importação. Os atrasos por conta de taxaço normalmente não são muito expressivos.	
Probabilidade:	<input checked="" type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input type="radio"/> 3 (média) <input type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: A maioria das encomendas remetidas por empresas internacionais ao Brasil é taxada.	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade):	
Mitigar: Reservar dinheiro para possível pagamento de impostos, e efetuar o pagamento o mais rapidamente possível, caso seja solicitado, de forma que o produto fique retido o menor tempo possível na alfândega. Dessa forma o impacto no cronograma será minimizado.	
Impacto reavaliado: 2	Probabilidade reavaliada: 5

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013	<input type="checkbox"/> Respostas incluídas na WBS/Cronograma	Registros adicionais: Verso ou Anexos
--	------------------	--	--

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Necessidade de mudança de escopo	Nº Identificação 06
Descrição do risco: Caso ocorram muitos problemas, previstos ou não, e isso impossibilite a conclusão total do projeto, o escopo deverá ser diminuído para o que será possível concluir o projeto com utilizando os recursos (humanos, tempo, dinheiro, etc...) disponíveis.	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input type="radio"/> 4 (média/alta) <input checked="" type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: A diminuição de escopo é um risco que pode ocorrer caso os recursos disponíveis não sejam suficientes para concluir o projeto em sua totalidade. O escopo deverá ser refeito considerando as limitações.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input type="radio"/> 3 (média) <input checked="" type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: A diminuição de escopo é um último caso a ser analisado quando os recursos (humanos, tempo, dinheiro, etc...) disponíveis impossibilitem a completude do projeto, ou quando existe recomendação proposta pelos <i>sponsors</i> do projeto.	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade):	
Prevenir: Manter um contato constante com os <i>sponsors</i> do projeto e manter um planejamento eficiente para que os recursos permitam que o projeto seja realizado em sua totalidade.	
Mitigar: Caso o escopo seja reajustado, a equipe deverá reportar o fato aos <i>sponsors</i> do projeto. Deve ser informado o novo escopo e as dificuldades que impediram o escopo anterior de ser realizado.	
Impacto reavaliado: 3	Probabilidade reavaliada: 3

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.

Data: 31/01/2013

☐ Respostas incluídas na WBS/Cronograma

Registros adicionais:
Verso ou Anexos

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Impossibilidade de uso do robô em certos horários.	Nº Identificação 07
Descrição do risco: O robô é patrimônio da universidade, e, portanto o uso dele é restrito aos dias e horários em que há disponibilidade tanto de equipamento quanto de laboratório para a equipe.	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input checked="" type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: O impacto de não ser possível manusear diretamente o robô é expressivo, uma vez que há previsão de que muitos testes tenham que ser feitos, e o tempo gasto com o desenvolvimento do <i>hardware</i> provavelmente será considerável. O acesso ao robô é imprescindível para o bom andamento do projeto.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input type="radio"/> 3 (média) <input checked="" type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: Há modesta probabilidade de não ser possível ter acesso direto ao robô, uma vez que o robô é patrimônio da universidade e a sua utilização deve ser previamente autorizada pelos <i>sponsors</i> .	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade):	
Prevenir: Marcar previamente os horários para ter acesso garantido ao robô. Organizar o cronograma de acordo com os horários de disponibilidade do robô.	
Mitigar: Focar o trabalho da equipe em áreas do projeto que não necessitem acesso direto ao robô (por exemplo, partes específicas do <i>software</i> ou projetos teóricos de circuitos).	
Impacto reavaliado: 4	Probabilidade reavaliada: 1

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.

Data: 31/01/2013

☐ Respostas incluídas na WBS/Cronograma

Registros adicionais:
Verso ou Anexos

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Problemas no planejamento e confecção da placa de circuito impresso.	Nº Identificação 08
Descrição do risco: Problemas na elaboração do modelo do circuito ou dificuldades com os fabricantes da placa.	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input checked="" type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: dificuldades com a placa de circuito impresso tem impacto direto no desenvolvimento do <i>hardware</i> e realização dos testes, uma vez que há dependência de que a placa esteja disponível e funcionando.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: A equipe possui um integrante na equipe que tem considerável experiência com desenvolvimento e confecção de placas de circuito impresso. Portanto, a probabilidade de haverem dificuldades do tipo é média.	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade): Prevenir: Buscar documentação a respeito dos <i>softwares</i> para modelagem de circuitos impressos. Antes de confeccionar a placa de circuito impresso, realizar simulações, quando possível, dos circuitos desenvolvidos. Mitigar: Buscar auxílio com pessoas mais especializadas na área (por exemplo, professores da universidade). Realizar testes e simulações, se possível, do circuito desenvolvido em busca de erros de projeto e fabricação.	
Impacto reavaliado: 4	Probabilidade reavaliada: 1

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.

Data: 31/01/2013

☐ Respostas incluídas na WBS/Cronograma

Registros adicionais:
Verso ou Anexos

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

Planejamento de Riscos

Projeto: Mapeamento de ambientes com o robô Bellator.

1º Etapa: Identificação do Risco

Denominação do risco: Falhas de comunicação entre os membros da equipe.	Nº Identificação 09
Descrição do risco: Por fatores como falta de tempo, problemas pessoais ou falta de comprometimento, as informações importantes a respeito do projeto podem não ser propriamente propagadas entre todos os membros da equipe.	

2º Etapa: Avaliação do Risco

Impacto:	<input type="radio"/> 5 (alta) <input checked="" type="radio"/> 4 (média/alta) <input type="radio"/> 3 (médio) <input type="radio"/> 2 (médio/baixo) <input type="radio"/> 1 (baixo)
Explique: O impacto caso ocorra um problema desse tipo considerável, uma vez que o andamento do projeto pode ser muito prejudicado. Leva-se em conta também que falhas de comunicação podem acarretar na desmotivação da equipe.	
Probabilidade:	<input type="radio"/> 5 (alto) <input type="radio"/> 4 (médio/alto) <input checked="" type="radio"/> 3 (média) <input type="radio"/> 2 (média/baixa) <input type="radio"/> 1 (baixa)
Explique: Tendo em vista a experiência prática de equipes anteriores, há uma probabilidade expressiva de ocorrer falhas de comunicação. Porém, uma vez que há vários canais de comunicação disponíveis e todos os membros possuem acesso à <i>internet</i> , essa probabilidade é reduzida.	

3º Etapa: Desenvolvimento da Resposta ao Risco

Ações, Responsáveis e Datas de Conclusão	
Estratégias e Ações para eliminar ou reduzir este risco (minimizar impacto e/ou probabilidade): Prevenir: Manter vários canais de comunicação (e controle de versões) entre os membros da equipe: <i>e-mail</i> , telefone, <i>github</i> , <i>Dropbox</i> . O gerente irá agendar ao menos uma reunião presencial por semana para troca de informações e motivação da equipe. Mitigar: Agendamento de reuniões presenciais com maior frequência com todos os membros da equipe. Uso mais frequente dos meios de comunicação disponíveis. Estímulo maior ao uso dos sistemas de controle de versões.	
Impacto reavaliado: 3	Probabilidade reavaliada: 1

Elaborado por: Luís, Pedro, Ricardo, Stefan e Telmo.	Data: 31/01/2013	<input type="checkbox"/> Respostas incluídas na WBS/Cronograma	Registros adicionais: Verso ou Anexos
--	------------------	--	--

Formulário sugerido por Gasnier, 2000 Editora IMAN e alterado por Wille

APÊNDICE B – MEDIDAS DO ROBÔ**Tabela 7:** Medidas do robô.

Medida	Valor
Comprimento da carcaça	50 cm
Largura da carcaça	40 cm
Distância entre a parte da frente do robô e os eixos das rodas	14 cm
Largura de cada roda	4 cm
Circunferência de cada roda	64 cm
Circunferência do eixo de cada roda	7,5 cm
Circunferência do eixo de cada encoder	22 cm

APÊNDICE C – DIAGRAMAS DETALHADOS DE HARDWARE

C.1 DIAGRAMA ELÉTRICO/ELETRÔNICO

Na figura 7 mostra-se o diagrama de elétrico eletrônico do sistema embarcado. Cada bloco da figura 4 corresponde a alguns componentes do diagrama elétrico eletrônico. A seguir detalha-se um pouco mais cada bloco.

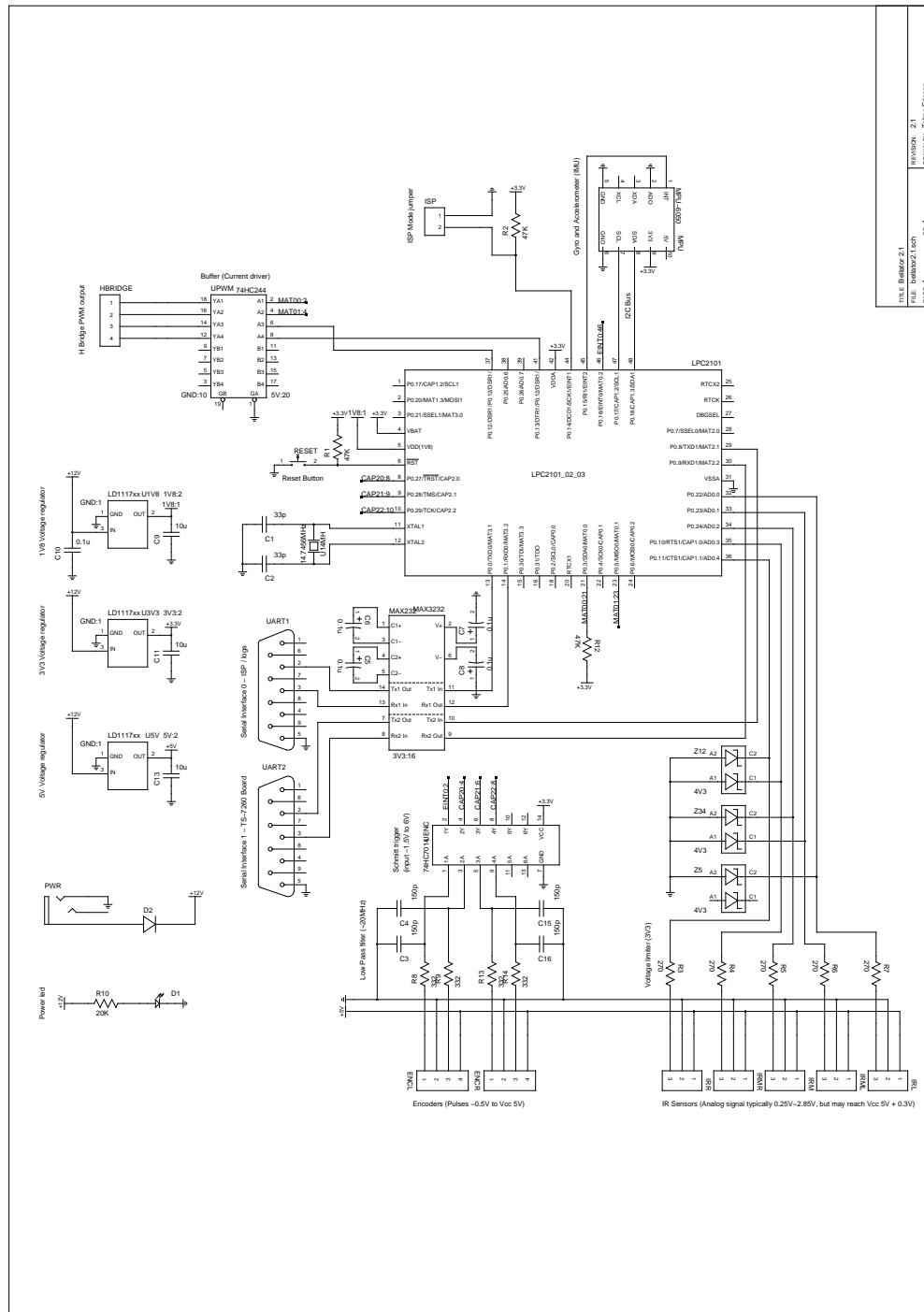


Figura 7: Diagrama elétrico/eletrônico.

1. Microcontrolador: Composto pelo microcontrolador LCP2103 da NXP que possui arquitetura ARM. Ele dispõe de duas interfaces seriais, conversor analógico digital com 8 canais, interface i2c, entradas de captura e interrupção, saídas de PWM entre outras funções que não serão utilizadas nesse projeto.

2. UART 0/1: Constituído por um chip max3232 que opera em níveis de tensão CMOS e que gera os níveis adequados para o padrão RS-232 utilizando alguns capacitores.
3. Buffer: Constituído por um chip 74HC244, é responsável por fornecer corrente e elevar os níveis de tensão de saída do microcontrolador de 3,3V para 5,0V.
4. IMU: Composto pela placa de desenvolvimento MPU-6050, que possui um chip com o mesmo nome, MPU-6050, e circuitos RC auxiliares necessários para o funcionamento do MPU-6050.
5. Limitador de tensão: Constituído de um resistor com baixo valor, 270 ohms, e um diodo Zener polarizado reversamente e com tensão de ruptura de 4.3V. Quando a tensão de entrada ultrapassar 4.3V o diodo passa a conduzir e mantém a tensão de 4.3V no resistor. O datasheet do microcontrolador sugere que se mantenha a impedância da carga menor que 40kohms, logo a adição de um resistor de 270 ohms pode ser desconsiderado com relação ao erro que possa causar na leitura do conversor. Um resistor de 270 ohms leva a uma corrente de 3.7mA quando a saída do sensor for 5.3V, que é o valor máximo previsto no datasheet.
6. Tratamento de sinal: Composto por um filtro RC passa baixas e um chip 74HC7014. As frequências acima de 20MHz são atenuadas no sinal do encoder. Esse valor foi calculado com base na forma de onda da saída especificada no datasheet do encoder. Para tanto utilizam-se resistores de 332 ohms e capacitores de 150pF.

C.2 PLACA DE CIRCUITO IMPRESSO

O projeto da placa de circuito impresso (PCB) do sistema embarcado de baixo nível está explicitado nas Figuras 8 e 9. Na Figura 10 está presente uma foto da placa pronta com os componentes soldados.

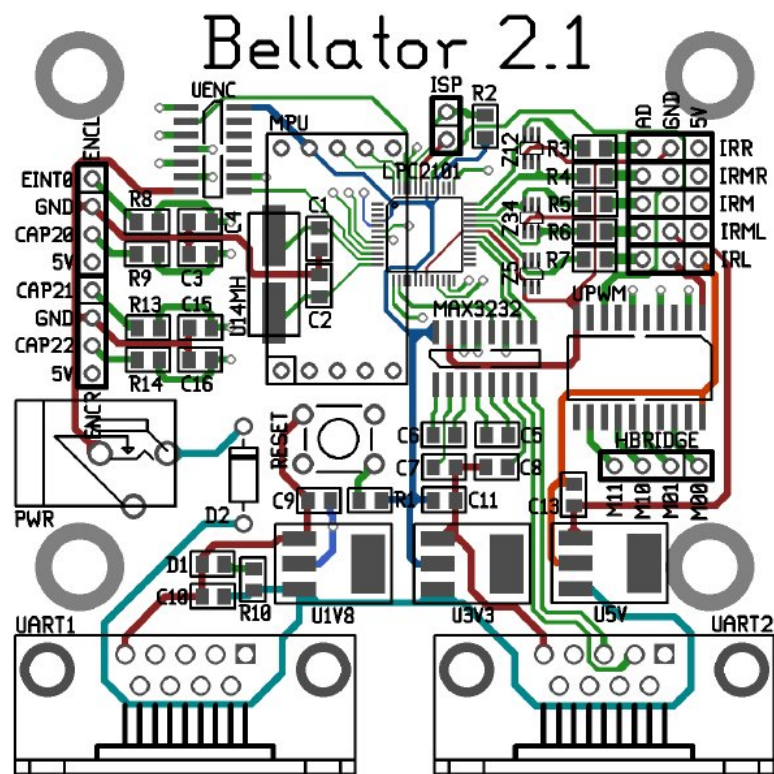


Figura 8: Projeto da PCB – lado de cima.

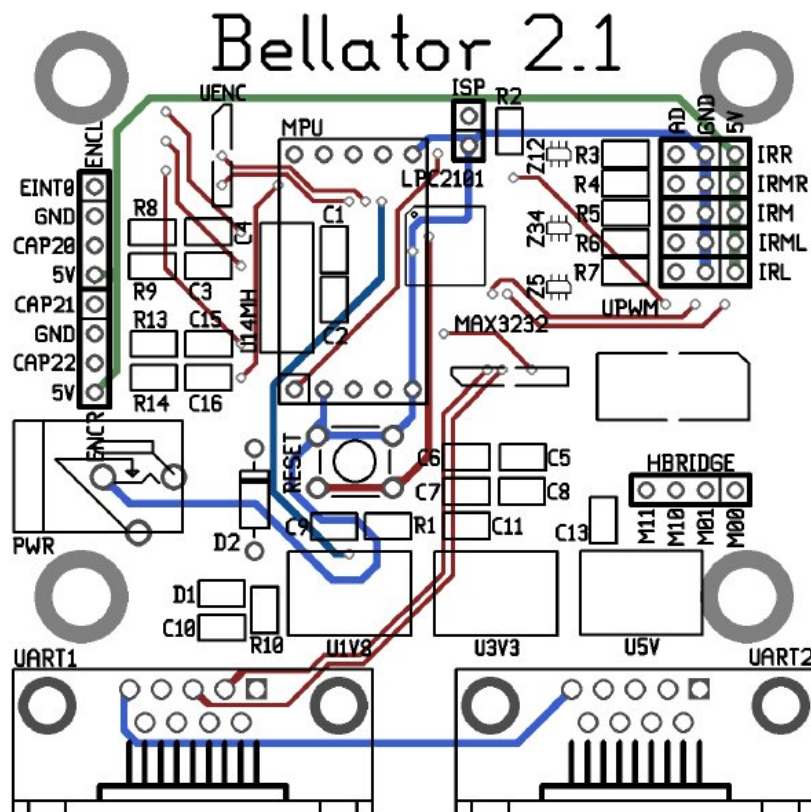


Figura 9: Projeto da PCB – lado de baixo.

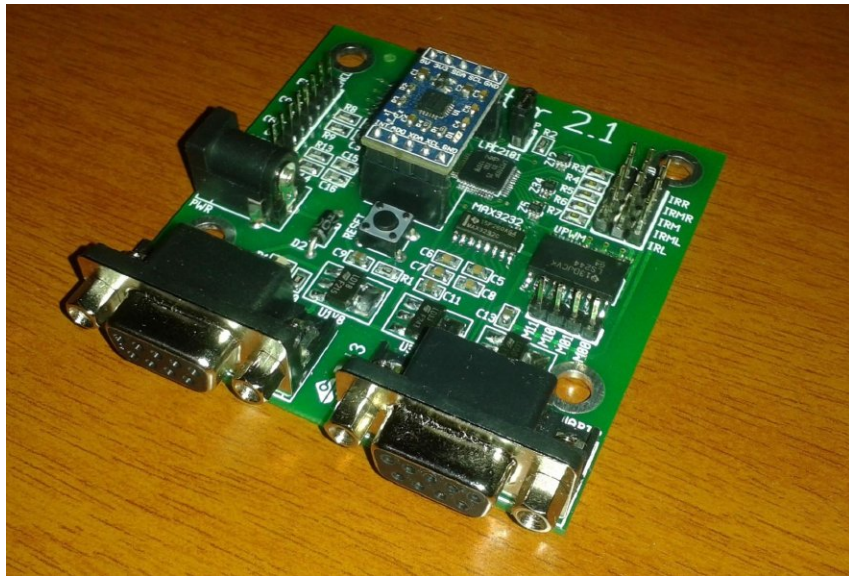













































Figura 10: PCB montada.

APÊNDICE D – CRONOGRAMA

		Name	Duration	Start	Finish
1		<input type="checkbox"/> Desenvolvimento dos entregáveis e Documentação	51 days	2/27/13 8:00 AM	5/8/13 5:00 PM
2		<input type="checkbox"/> Etapa 1	11 days	2/27/13 8:00 AM	3/13/13 5:00 PM
3		Primeira versão do diagrama de casos de uso	11 days	2/27/13 8:00 AM	3/13/13 5:00 PM
4		Primeira versão dos requisitos Funcionais e não-funcionais	11 days	2/27/13 8:00 AM	3/13/13 5:00 PM
5		Primeira versão do diagrama de classes	11 days	2/27/13 8:00 AM	3/13/13 5:00 PM
6		Primeira versão do diagrama de blocos do hardware	11 days	2/27/13 8:00 AM	3/13/13 5:00 PM
7		<input type="checkbox"/> Etapa 2	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
8		<input type="checkbox"/> Atualizar a documentação desenvolvida na Etapa 1	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
9		Segunda versão do diagrama de casos de uso	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
10		Segunda versão dos requisitos Funcionais e não-funcionais	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
11		Segunda versão do diagrama de classes	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
12		Segunda versão do diagrama de fluxo de dados do hardware	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
13		<input type="checkbox"/> Desenvolvimento de documentação adicional necessária	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
14		Primeira versão do diagrama de fluxo de dados do software embarcado	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
15		Primeira versão dos diagramas de estados do sistema de comunicação	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
16		Primeira versão da descrição das mensagens e codificações dos coman...	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
17		Primeira versão do diagrama elétrico/eletrônico do hardware	10 days	3/14/13 8:00 AM	3/27/13 5:00 PM
18		<input type="checkbox"/> Etapa 3	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
19		<input type="checkbox"/> Atualizar a documentação desenvolvida na Etapa 2	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
20		Terceira versão do diagrama de casos de uso	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
21		Terceira versão dos requisitos Funcionais e não-funcionais	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
22		Terceira versão do diagrama de classes	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
23		Terceira versão do diagrama de blocos do hardware	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
24		Segunda versão do diagrama de fluxo de dados do software embarcado	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
25		Segunda versão dos diagramas de estados do sistema de comunicação	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
26		Segunda versão da descrição das mensagens e codificações dos comand...	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
27		<input type="checkbox"/> Etapa 4	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
28		Projeto da Printed Circuit Board	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
29		Lista de componentes para confecção da PCB	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
30		Atualização dos documentos da Etapa 3	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
31		Descrição do uso do Software	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
32		Mapa da conexão da PCB com a alimentação e outros elementos do hardw...	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
33		Guia de montagem	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
34		<input type="checkbox"/> Etapa Final	10 days	4/25/13 8:00 AM	5/8/13 5:00 PM
35		Revisão da Monografia	10 days	4/25/13 8:00 AM	5/8/13 5:00 PM
36		<input type="checkbox"/> Implementação do software da estação base	37 days	3/5/13 8:00 AM	4/24/13 5:00 PM
37		<input type="checkbox"/> Backend	17 days	3/5/13 8:00 AM	3/27/13 5:00 PM
38		Abstração em alto nível do controle dos sensores, webcam, motores e mapa	9 days	3/5/13 8:00 AM	3/15/13 5:00 PM
39		Salvar e carregar mapas	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
40		Desenvolvimento do protocolo de comunicação	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
41		Testes com o software	8 days	3/18/13 8:00 AM	3/27/13 5:00 PM
42		<input type="checkbox"/> Frontend	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
43		Desenho do mapa 2D	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
44		Display de imagens da webcam	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
45		Interface gráfica integrada	5 days	3/11/13 8:00 AM	3/15/13 5:00 PM
46		Implementação do software de alto nível do sistema embarcado	10 days	3/28/13 8:00 AM	4/10/13 5:00 PM
47		Simulações e testes no computador e ajustes	10 days	4/11/13 8:00 AM	4/24/13 5:00 PM
48		<input type="checkbox"/> Implementação do hardware	41 days	2/27/13 8:00 AM	4/24/13 5:00 PM
49		Especificação do diagrama esquemático do hardware	6 days	2/27/13 8:00 AM	3/6/13 5:00 PM
50		Montagem do protótipo em protoboard	8 days	3/7/13 8:00 AM	3/18/13 5:00 PM
51		Testes e ajustes com o protótipo	7 days	3/19/13 8:00 AM	3/27/13 5:00 PM
52		Projeto da PCB, Roteamento da placa e envio para impressão	15 days	3/28/13 8:00 AM	4/17/13 5:00 PM
53		Soldagem dos componentes	5 days	4/18/13 8:00 AM	4/24/13 5:00 PM
54		<input type="checkbox"/> Fase final de desenvolvimento	10 days	4/25/13 8:00 AM	5/8/13 5:00 PM
55		Testes e ajustes finais do robô	10 days	4/25/13 8:00 AM	5/8/13 5:00 PM

REFERÊNCIAS

BOT, P. 2013. Disponível em:

<<http://www.mobilerobots.com/researchrobots/researchpatrolbot.aspx>>.

MARIN, A. J.; BORGES, J. C. N.; WERGRZN, Y. A. **Desenvolvimento de robô móvel e análise qualitativa de algoritmos de navegação fuzzy**. Curitiba, 2012.

SATO, B. N. W. K. 2 dimensional infrared robotic mapping system. 2013. Disponível em:

<<http://www-sens.sys.es.osaka-u.ac.jp/research/thesis/10/tech-rep/brad/thesis.pdf>>.