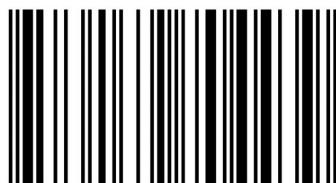


## Desenvolvimento de robô para exploração de ambientes perigosos

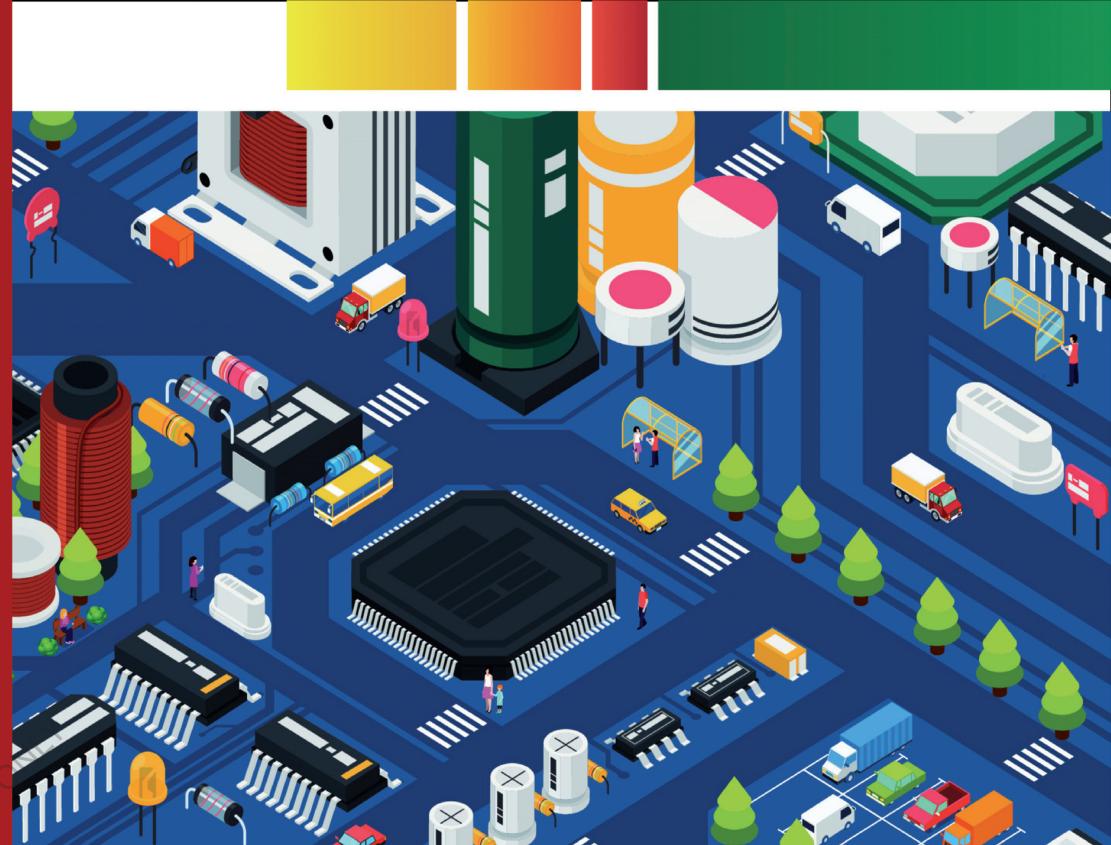
O robô coletor tem por objetivo auxiliar em situações onde existam ameaças para vida, seja pela suspeita de algum artefato explosivo, ambientes onde existam gases inflamáveis ou tóxicos, permitindo realizar um reconhecimento do ambiente por meio de câmeras e em alguns casos coletar o material e levá-lo a um local seguro. Por ser controlado a distância conta com um sistema supervisório, permitindo manipular o material suspeito sem colocar em risco a vida humana. Foram escolhidos os componentes a serem utilizados na fabricação do robô e identificadas quais são suas principais características e limitações, além do levantamento das adaptações necessárias. Em seguida todo o hardware do robô foi montado e suas capacidades construtivas foram avaliadas como: a locomoção, movimentação da câmera e do braço robótico, além da leitura de sensores. Entre as capacidades do robô estão a de identificar a presença de fumaça e alguns gases inflamáveis no ambiente, se o objeto coletado está emitindo calor e por meio da câmera realizar o reconhecimento do ambiente interagindo com o mesmo controlando o braço robótico e locomovendo-se. Trata-se de um trabalho envolvendo hardware, software e robótica.



Acadêmico de Engenharia de Computação, Engenharia Elétrica e Licenciatura em Informática. Desenvolvedor de software, sistemas embarcados, aplicativos móveis, sistemas supervisórios, hardware e inteligência artificial. Maker desde a infância sempre gostei de estudar e construir coisas. Apaixonado pela ciência, desde de química até astronomia.



978-613-9-70443-9



Rafael Pablo Massocato

## Desenvolvimento de robô para exploração de ambientes perigosos

Hardware, placa microcontroladora e sistema supervisório

Rafael Pablo Massocato

**Desenvolvimento de robô para exploração de ambientes perigosos**

FOR AUTHOR USE ONLY



**Rafael Pablo Massocato**

# **Desenvolvimento de robô para exploração de ambientes perigosos**

**Hardware, placa microcontroladora e sistema  
supervisório**

FOR AUTHOR USE ONLY

**Novas Edições Acadêmicas**

### **Imprint**

Any brand names and product names mentioned in this book are subject to trademark, brand or patent protection and are trademarks or registered trademarks of their respective holders. The use of brand names, product names, common names, trade names, product descriptions etc. even without a particular marking in this work is in no way to be construed to mean that such names may be regarded as unrestricted in respect of trademark and brand protection legislation and could thus be used by anyone.

Cover image: [www.ingimage.com](http://www.ingimage.com)

Publisher:

Novas Edições Acadêmicas

is a trademark of

International Book Market Service Ltd., member of OmniScriptum Publishing

Group

17 Meldrum Street, Beau Bassin 71504, Mauritius

Printed at: see last page

**ISBN: 978-613-9-70443-9**

Copyright © Rafael Pablo Massocato

Copyright © 2018 International Book Market Service Ltd., member of  
OmniScriptum Publishing Group

FOR AUTHOR USE ONLY

FOR AUTHOR USE ONLY

Dedico este trabalho ao Alexandre e a Magda, que  
são sem dúvida, são os melhores pais! E a Daniele  
por toda a ajuda ao longo dos anos de graduação.

## **AGRADECIMENTOS**

Agradeço a todas as pessoas que em primeiro lugar acreditaram em mim e na minha força de vontade, sejam eles amigos, pais, professores ou familiares. Gostaria de agradecer especialmente ao professor Karl Phillip Purnhagen Ehlers Peixoto Dittrich Buhr por todo conhecimento adquirido nas aulas de programação, e pela preocupação que ele sempre teve com o nosso aprendizado e desenvolvimento não somente como acadêmicos, mas como programadores. Agradeço também os professores envolvidos no meu trabalho de conclusão de curso: Kleyton Hoffmann e Daniel Calixto Fagonde Moraes, que sempre estiveram acompanhando o meu trabalho e auxiliando com dicas e orientações de extrema importância.

FOR AUTHOR USE ONLY

## **RESUMO**

O robô coletor tem por objetivo auxiliar em situações onde existam ameaças para vida, seja pela suspeita de algum artefato explosivo, ambientes onde existam gases inflamáveis ou tóxicos, permitindo realizar um reconhecimento do ambiente por meio de câmeras e em alguns casos coletar o material e levá-lo a um local seguro. Por ser controlado a distância conta com um sistema supervisório, permitindo manipular o material suspeito sem colocar em risco a vida humana. Foram escolhidos os componentes a serem utilizados na fabricação do robô e identificadas quais são suas principais características e limitações, além do levantamento das adaptações necessárias. Em seguida todo o *hardware* do robô foi montado e suas capacidades construtivas foram avaliadas como: a locomoção, movimentação da câmera e do braço robótico, além da leitura de sensores. Entre as capacidades do robô estão a de identificar a presença de fumaça e alguns gases inflamáveis no ambiente, se o objeto coletado está emitindo calor e por meio da câmera realizar o reconhecimento do ambiente interagindo com o mesmo controlando o braço robótico e locomovendo-se. A conexão entre os componentes eletrônicos e microcontrolador utilizados no robô foram abordadas detalhadamente, da mesma forma que a construção do chassi e a conexão de todos os componentes envolvidos. Desenvolveu-se também todo *software* necessário para controle das funcionalidades apresentadas pelo robô bem como o sistema embarcado responsável pelo acionamento de *hardware* e o controle do mesmo. Foram realizados testes em uma sala comercial para identificar quais as capacidades do robô, simulando uma situação de perigo real, onde o robô conseguiu transmitir a leitura dos sensores de gases e temperatura até uma distância máxima de 50 metros. Durante as simulações foram obtidas imagens por meio da câmera com distância máxima de 25 metros utilizando *wi-fi*. O deslocamento do robô alcançou a velocidade máxima de 1,33 Km/h e o tempo de autonomia foi de 23 minutos. Em 66% das tentativas conseguiu coletar e transportar o objeto, utilizando o braço robótico.

Palavras-chave: Robô Coletor, Microcontrolador, Supervisório.

## **ABSTRACT**

*The robot collector aims to assist in situations where there are threats to life, either by the suspicion of some explosive device, environments where there are flammable or toxic gases, allowing to perform a recognition of the environment through cameras and in some cases collect the material and take it to a safe place. Being controlled at a distance has a supervisory system, allowing to manipulate the suspect material without endangering human life. The components to be used in the manufacture of the robot were chosen and identified its main characteristics and limitations, as well as the necessary adaptations. Then the robot's hardware was assembled and its constructive capacities were evaluated as: locomotion, movement of the camera and robotic arm, besides the reading of sensors. Among the capabilities of the robot are to identify the presence of smoke and some flammable gases in the environment if the collected object is emitting heat and through the camera perform the recognition of the environment interacting with it controlling the robotic arm and locomotion. The connection between the electronic components and microcontroller used in the robot were approached in detail, in the same way as the construction of the chassis and the connection of all the components involved. Also developed was all software required to control the functionalities presented by the robot as well as the embedded system responsible for hardware activation and control. Tests were performed in a commercial room to identify the capabilities of the robot, simulating a real danger situation, where the robot was able to transmit the reading of the gas and temperature sensors up to a maximum distance of 50 meters. During the simulations, images were obtained through the camera with a maximum distance of 25 meters. The displacement of the robot reached the maximum speed of 1.33 Km / h and the time of autonomy was of 23 minutes. In 66 % of the attempts he was able to collect and transport the object using the robotic arm.*

*Key-words:* Disposal Robot. Microcontroller. Supervisory System.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Robô Explorador Sojourner . . . . .	18
Figura 2 – PackBot 510 . . . . .	20
Figura 3 – Motor e caixa . . . . .	22
Figura 4 – Placa para controle de PWM . . . . .	23
Figura 5 – Exemplo de servomotor . . . . .	23
Figura 6 – Exemplo de suspensão de eixo (A) e independente (B) . . . . .	24
Figura 7 – Placa Reguladora XL4005 . . . . .	25
Figura 8 – Raspberry Pi 3 Model B . . . . .	26
Figura 9 – Módulo de câmera . . . . .	27
Figura 10 – Atmega2560 . . . . .	28
Figura 11 – Adaptador SMD/DIP . . . . .	28
Figura 12 – Módulo FT232R . . . . .	29
Figura 13 – Sensores de gases e temperatura . . . . .	30
Figura 14 – Braço robótico de 6 eixos . . . . .	31
Figura 15 – Exemplo de sistema supervisório . . . . .	32
Figura 16 – Módulo XBee . . . . .	33
Figura 17 – Controle Xbox 360 . . . . .	33
Figura 18 – Sistema completo com robô coletores e supervisório . . . . .	35
Figura 19 – Chassi do Robô . . . . .	36
Figura 20 – Sistema de suspensão, motores e rodas . . . . .	37
Figura 21 – Componentes eletrônicos . . . . .	37
Figura 22 – Protótipo do Robô concluído . . . . .	38
Figura 23 – Molde para chassi . . . . .	38
Figura 24 – Molde para chassi com tecido . . . . .	39
Figura 25 – Chassi para montagem da suspensão . . . . .	39
Figura 26 – Diagrama do circuito da placa microcontroladora . . . . .	40
Figura 27 – Circuito do cristal oscilador . . . . .	41
Figura 28 – Desenho da camada superior e inferior da placa desenvolvida . . . . .	42
Figura 29 – Corrosão da placa de fenolite . . . . .	42
Figura 30 – Soldagem dos componentes da placa . . . . .	43
Figura 31 – Componentes conectados ao Raspberry Pi . . . . .	44

Figura 32 – Circuito utilizado para movimentar a câmera . . . . .	45
Figura 33 – Circuito da ponte H . . . . .	46
Figura 34 – Circuito utilizado no braço robótico . . . . .	47
Figura 35 – Circuito de alimentação dos componentes do robô . . . . .	47
Figura 36 – Circuito dos sensores utilizados pelo robô . . . . .	48
Figura 37 – Braço robótico . . . . .	49
Figura 38 – Módulo da câmera . . . . .	50
Figura 39 – Tela central do Supervisório . . . . .	56
Figura 40 – Comunicação: Controle, supervisório e microcontrolador . . . . .	60
Figura 41 – Comunicação: Raspberry Pi e supervisório . . . . .	61
Figura 42 – Rampa para teste de locomoção . . . . .	62
Figura 43 – Leitura de sensores no Supervisório . . . . .	63
Figura 44 – Braço robótico com 4 eixos . . . . .	64
Figura 45 – Teste da potência do sinal Wi-Fi . . . . .	65
Figura 46 – Teste da câmera do Raspberry Pi e do supervisório . . . . .	66
Figura 47 – Identificação de objeto inflamável a ser coletado . . . . .	67
Figura 48 – Coleta do objeto inflamável e deslocamento . . . . .	67
Figura 49 – Disposição de material perigoso em ambiente ventilado . . . . .	67
Figura 50 – Circuito completo do robô . . . . .	73

## **LISTA DE TABELAS**

Tabela 1 – Operações e consultas de sistema . . . . .	54
Tabela 2 – Orçamento dos componentes utilizados . . . . .	68

FOR AUTHOR USE ONLY

## LISTA DE ABREVIATURAS E SIGLAS

ADC	<i>Analog Digital Converter</i>
ARM	<i>Advanced RISC Machine</i>
CAD	<i>Computer-Aided Design</i>
CSI	<i>Câmera Serial Interface</i>
DIP	<i>Dual In-Line Package</i>
GPIO	<i>General Purpose Input/Output</i>
GPRS	<i>General Packet Radio Services</i>
HDMI	<i>High-Definition Multimedia Interface</i>
HTTP	<i>Hyper Text Transfer Protocol</i>
I <sup>2</sup> C	<i>Inter-Integrated Circuit</i>
IDE	<i>Integrated Development Environment</i>
ISM	<i>The Industrial, Scientific and Medical radio bands</i>
NASA	<i>The National Aeronautics and Space Administration</i>
PWM	<i>Pulse Width Modulation</i>
SMD	<i>Surface Mount Device</i>
SPI	<i>Serial Peripheral Interface</i>
UART	<i>Universal Synchronous Receiver/Transmitter</i>
USB	<i>Universal Serial Bus</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>15</b>
1.1	DESCRIÇÃO DO PROBLEMA . . . . .	15
1.2	JUSTIFICATIVA . . . . .	16
1.3	OBJETIVOS . . . . .	16
<b>1.3.1</b>	<b>Objetivos específicos . . . . .</b>	<b>16</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>17</b>
2.1	Robôs . . . . .	17
<b>2.1.1</b>	<b>Robô Explorador . . . . .</b>	<b>17</b>
<b>2.1.2</b>	<b>Componentes do Robô Explorador . . . . .</b>	<b>18</b>
<b>2.1.3</b>	<b>Robô Coletor . . . . .</b>	<b>19</b>
<b>2.1.4</b>	<b>Custo de um Robô Coletor . . . . .</b>	<b>20</b>
<b>2.1.5</b>	<b>Componentes do Robô Coletor . . . . .</b>	<b>20</b>
2.1.5.1	Motores elétricos e caixas de redução . . . . .	21
2.1.5.2	Placa transistorizada para controle de PWM . . . . .	22
2.1.5.3	Servomotores . . . . .	22
2.1.5.4	Chassi e suspensão . . . . .	23
2.1.5.5	Baterias . . . . .	24
2.1.5.6	Placas Reguladoras de tensão . . . . .	25
2.1.5.7	Computador Raspberry Pi . . . . .	25
2.1.5.8	Módulo de câmera Raspberry Pi . . . . .	26
2.1.5.9	Microcontrolador Atmel ATmega2560 . . . . .	27
2.1.5.10	Placa adaptadora SMD para DIP . . . . .	27
2.1.5.11	Xbee Explorer Adaptador USB . . . . .	28
2.1.5.12	Sensores . . . . .	29
2.1.5.13	Braço robótico . . . . .	30
2.1.5.14	Supervisório . . . . .	31
2.1.5.15	Comunicação de Dados via Wi-fi . . . . .	31
2.1.5.16	Comunicação de Dados via Xbee . . . . .	32
2.1.5.17	Controle Xbox 360 . . . . .	33
2.1.5.18	Linguagens de Programação . . . . .	34
<b>3</b>	<b>METODOLOGIA . . . . .</b>	<b>35</b>

3.1	PROJETO E CONSTRUÇÃO DO CHASSI DO ROBÔ . . . . .	36
3.1.1	Projeto CAD do Robô . . . . .	36
3.1.2	Molde para chassis . . . . .	38
3.1.3	Fabricação do chassis . . . . .	39
3.2	Placa microcontroladora desenvolvida com ATmega2560 . . . . .	39
3.2.1	Círcuito do cristal oscilador . . . . .	41
3.2.2	Desenho da placa em duas camadas . . . . .	41
3.2.3	Corrosão da placa . . . . .	41
3.2.4	Soldagem de componentes . . . . .	42
3.2.5	Testes da placa microcontroladora . . . . .	43
3.3	Circuitos do Robô . . . . .	43
3.3.1	Conexões do Raspberry Pi . . . . .	44
3.3.2	Conexões da placa microcontroladora desenvolvida . . . . .	44
3.3.2.1	Círcuito dos servos para movimentar a câmera . . . . .	45
3.3.2.2	Círcuito da ponte H - Alteração para controle PWM . . . . .	45
3.3.2.3	Círcuito do braço robótico . . . . .	46
3.3.3	Alimentação dos Dispositivos . . . . .	46
3.3.3.1	Sensores . . . . .	48
3.4	MONTAGEM . . . . .	48
3.4.1	Montagem da suspensão e motores . . . . .	48
3.4.2	Montagem do braço robótico . . . . .	49
3.4.2.1	Adição de mais um eixo ao braço robótico . . . . .	50
3.4.3	Montagem do suporte para câmera . . . . .	50
3.4.4	Montagem das peças de acrílico . . . . .	50
3.5	Documentação de hardware . . . . .	51
3.5.1	Diagramas de contexto e requisitos . . . . .	51
3.6	Documentação de software . . . . .	52
3.6.1	Identificação de Operações e Consultas de Sistema . . . . .	52
3.6.2	Modelo Conceitual . . . . .	54
3.6.3	Diagrama de classes de projeto . . . . .	55
3.6.4	Projeto gráfico da tela central . . . . .	55
3.6.5	Desenvolvimento do software embarcado . . . . .	56
3.6.5.1	Biblioteca para controle de portas IO . . . . .	56

3.6.5.2	Biblioteca para leitura de portas analógicas . . . . .	57
3.6.5.3	Biblioteca para controle dos servomotores . . . . .	57
3.6.5.4	Biblioteca para comunicação USART . . . . .	58
<b>3.6.6</b>	<b>Desenvolvimento de Software Supervisório . . . . .</b>	<b>58</b>
3.6.6.1	Conexão com Raspberry Pi via IP . . . . .	58
3.6.6.2	Conexão com a placa microcontroladora via USART . . . . .	59
3.6.6.3	Detecção de eventos do controle Xbox . . . . .	59
<b>3.6.7</b>	<b>Comunicação entre controle, supervisório e robô . . . . .</b>	<b>60</b>
3.7	TESTES . . . . .	62
<b>3.7.1</b>	<b>Locomoção . . . . .</b>	<b>62</b>
<b>3.7.2</b>	<b>Leitura de sensores . . . . .</b>	<b>63</b>
<b>3.7.3</b>	<b>Braço robótico . . . . .</b>	<b>64</b>
<b>3.7.4</b>	<b>Alcance do Sinal . . . . .</b>	<b>64</b>
<b>3.7.5</b>	<b>Imagens obtidas da câmera . . . . .</b>	<b>65</b>
<b>3.7.6</b>	<b>Autonomia da Bateria . . . . .</b>	<b>65</b>
<b>3.7.7</b>	<b>Simulação de ambiente com materiais explosivos, inflamáveis e fumaça tóxica . . . . .</b>	<b>66</b>
3.8	RECURSOS E ORÇAMENTOS . . . . .	68
<b>3.8.1</b>	<b>Recursos de <i>hardware</i> . . . . .</b>	<b>68</b>
<b>4</b>	<b>RESULTADOS . . . . .</b>	<b>69</b>
<b>5</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>70</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>71</b>
<b>APÊNDICE A</b>	<b>Circuito completo do robô coletor desenvolvido . . . . .</b>	<b>73</b>
<b>APÊNDICE B</b>	<b>Inicialização da conexão com a câmera via IP . . . . .</b>	<b>74</b>
<b>APÊNDICE C</b>	<b>Inicialização da <i>thread</i> e leitura de dados para conexão via Serial . . . . .</b>	<b>75</b>
<b>APÊNDICE D</b>	<b>Assinatura de método utilizado para detecção de eventos do controle Xbox . . . . .</b>	<b>76</b>

## 1 INTRODUÇÃO

A robótica está se tornando cada vez mais popular e acessível a todos, seja para o lazer ou para o trabalho, os robôs estão à décadas ajudando o homem, nas mais diversas atividades, seja na montagem de linhas industriais, soldagens complexas, explorações oceânicas ou espaciais. Deste modo, no futuro os robôs estarão cada vez mais presentes em nossas vidas. Desde a época da corrida espacial, a Nasa envia robôs para explorar planetas vizinhos, sendo que para o planeta Marte foram enviados cerca de oito robôs com sucesso. Tendo como objetivo explorar os planetas, identificar suas propriedades e se um dia existiu vida neles. O mesmo acontece com os oceanos, que são explorados por submarinos controlados da superfície, por meio dos quais, é possível identificar novas espécies de animais e compreender melhor o ambiente. Existem outras tarefas que colocam em risco a vida humana, como desarmamento de bombas, ambientes radioativos, entre outros, em que os robôs se mostram muito eficientes. Foi construído um robô capaz de acessar locais perigosos, onde possa haver ameaças de explosões ou gases inflamáveis e tóxicos, permitindo realizar o reconhecimento do local e algumas vezes coletar o material explosivo ou inflamável levando-o para um local seguro. Este foi construído para que supra essas necessidades sem um custo muito elevado, foram escolhidos dispositivos prontos, já testados e que são facilmente encontrados no mercado, estes dispositivos devem ser adaptados para que os mesmos possam desempenhar as funções necessárias ao funcionamento do robô. É realizada uma explanação para identificar o que é um robô coletor, como estes são classificados e qual o tipo que será desenvolvido neste trabalho. Em seguida, é explicado detalhadamente os principais componentes que foram utilizados, e finalmente, os detalhes construtivos, montagem e testes de toda a parte de *hardware* do robô. São apresentadas também os registradores utilizados nas bibliotecas desenvolvidas para o *software* embarcado responsável pelo controle do robô, bem como o desenvolvimento do sistema supervisório. A etapa final consiste no teste do robô coletor em um ambiente que simula uma situação de perigo real, identificando quais são as capacidades e limitações.

### 1.1 DESCRIÇÃO DO PROBLEMA

Alguns ambientes podem ser fatais para os seres humanos, por conterem ameaças químicas, radioativas, entre outras. Da mesma forma, toda vez que um especialista se dirige até o local onde existe algum material explosivo ou gases inflamáveis, sua vida corre perigo. Segundo Reno (2000), o número de incidentes envolvendo artefatos explosivos nos Estados Unidos em

quatro anos ultrapassa os dois mil casos. Por este motivo, os robôs coletores são atualmente muito utilizados, eles auxiliam os especialistas a levarem os artefatos explosivos a um local seguro, sem colocar em risco a vida humana. No Brasil foram adquiridos 30 robôs para auxiliar na segurança da Copa de Mundo no ano de 2014, totalizando um investimento de 16,2 milhões de reais (CAPUTO, 2014). Estes robôs tem um custo muito elevado, não existindo modelos de baixo custo disponíveis no mercado e muito menos empresas nacionais que os fabriquem.

## 1.2 JUSTIFICATIVA

Com o desenvolvimento do robô coletor é possível ter acesso a locais potencialmente perigosos sem colocar em risco a vida humana. Sendo possível verificar a presença de gases inflamáveis e tóxicos ou coletar materiais explosivos para serem levados a um local seguro. Devido aos robôs comercializados com este fim serem importados, a proposta deste robô é que ele seja um protótipo para desenvolvimento de um produto nacional, levando em consideração os custos, sendo que muitas vezes o equipamento é perdido danificado por explosões ou chamas.

## 1.3 OBJETIVOS

Desenvolver um robô controlado remotamente para acessar locais perigosos onde existam ameaças de explosão, gases inflamáveis ou tóxicos permitindo realizar o reconhecimento do local ou coletar materiais explosivos ou inflamáveis de pequeno porte.

### 1.3.1 Objetivos específicos

- Desenvolver uma placa microcontroladora.
- Desenvolver o *software* embarcado a ser utilizado para controle.
- Desenvolver o supervisório e realizar a comunicação com o robô.
- Realizar ajustes gerais no robô
- Executar testes com o robô em um ambiente de perigo simulado para identificar as capacidades e limitações.
- Finalizar o projeto e apresentar conclusões.

## 2 FUNDAMENTAÇÃO TEÓRICA

### 2.1 ROBÔS

Conforme o Inovation (2005), localizado no Estado da Califórnia nos Estados Unidos, segundo a documentação disponibilizada pelo site: os robôs são máquinas que obtêm informações sobre o ambiente por meio de sensores e usam elas para seguir instruções de como agir. Porém, está se tornando cada vez mais difícil definir, o que de fato é um robô, sendo que existem atualmente muitos tipos de robôs que são utilizados nas mais variadas atividades. Pode-se entender isso, como as funcionalidades fornecidas por um *smartphone* como geolocalização, ou utilização de dados móveis, sabe-se que nem todos os robôs tem essas funcionalidades apesar de serem muito avançados em outros aspectos. Para Wise (2005), não existe uma definição exata do que seja um robô, no entanto eles possuem características essenciais para determinar se são ou não um robô. Desta forma é possível identificar ao construir uma máquina se ela será mesmo um robô. Entre os itens essenciais estão listados: sensores, movimento, energia e inteligência de acordo com o que foi programado para fazer. A definição do autor é então, um sistema que contém sensores, sistemas de controle, manipuladores, fonte de alimentação e *software* trabalhando juntos para executar determinada tarefa. E ainda informa que a concepção, construção, testes e programação de um robô envolve física, engenharia mecânica, engenharia elétrica, matemática e informática.

De acordo com Inovation (2005), existem vários tipos de robôs com os mais diversos propósitos. Com o passar do tempo os robôs modificaram a indústria e estão trazendo para nossas vidas cada vez mais conforto. Por meio deles a produtividade das fábricas aumentou e muitas tarefas se tornaram mais seguras e fáceis de realizar, por este motivo existem atualmente muitos tipos de robôs, sendo alguns desenvolvidos para realizar atividades específicas como soldas e montagens, enquanto outros, como *drones* voadores, podem realizar as tarefas mais variadas, de acordo com os acessórios acoplados a ele.

#### 2.1.1 Robô Explorador

O primeiro robô explorador foi desenvolvido pela NASA em 1997 chamado de Sojourner. Segundo o site Nasa (2015), seu objetivo era alcançar a superfície do planeta Marte e transmitir dados em imagens. Este robô foi levado pelo módulo Pathfinder e transmitiu várias fotos do planeta, além da composição de pedras encontradas no solo marciano. Com as dimen-

sões de 68 cm de comprimento, 48 cm de largura e 30 cm de altura o robô pesava cerca de 10 Kg. Sua missão foi encerrada no mesmo ano quando a comunicação com a Terra foi perdida, esta missão foi a segunda mais barata da história da NASA. A Figura 1 apresenta o protótipo da Sojourner.

Figura 1 – Robô Explorador Sojourner



Fonte: Nasa (2015)

Muitos robôs foram desenvolvidos com o intuito de explorar ambientes, inclusive outros exploradores foram enviados em missões ao planeta Marte. Atualmente ainda há um robô chamado Opportunity, que segundo a Nasa (2015), em onze anos de exploração já percorreu mais de 42 quilômetros de distância em solo marciano. Alguns robôs foram desenvolvidos para serem utilizados em escombros, desarme de bombas, entre outras atividades perigosas para humanos. Algumas empresas vendem ou alugam este tipo de robô, porém eles são de alto custo.

### 2.1.2 Componentes do Robô Explorador

Segundo as informações disponibilizadas pela Nasa (2015), os robôs exploradores possuem alguns componentes essenciais. Todo robô é dotado de um corpo que permite abrigar todos os componentes e suportar os esforços mecânicos exercidos sobre suas peças, este seria o esqueleto ou chassi onde tudo está agrupado. Conforme Nasa (2015), ele possui um cérebro capaz de realizar processamento dos dados que são coletados durante a exploração, assim como a energia que alimenta seus circuitos e faz com que o robô consiga operar satisfatoriamente. Um robô do tipo explorador é dotado de, no mínimo, uma câmera a fim de identificar o que está a sua frente e nas laterais e se ele pode avançar naquela direção, além de prover a

capacidade de observar vários objetos importantes ou qualquer tipo de ameaça (NASA, 2015). Podendo ser comparada a visão humana, é por meio da câmera que o robô irá perceber todos os elementos visuais a sua volta. Da mesma forma que o robô visualiza com a câmera ele pode ouvir sons através de microfones, permitindo assim identificar ruídos que sejam de interesse de quem o estiver operando. Um robô explorador também possui na maioria das vezes, segundo Nasa (2015), um braço robótico que permite a ele coletar amostras ou acionar algum dispositivo como se fosse um braço humano, obviamente sem a mesma precisão de movimentos, pois este não possui sensibilidade. Todavia, em muitos casos esta imprecisão pode ser desprezada, pois mesmo com movimentos bruscos ele será capaz de acionar um botão ou retirar algum obstáculo do seu caminho. Para conseguir se locomover um robô necessita de algum dispositivo que seja capaz de suportar o seu peso e ainda assim avançar. De acordo com a Nasa (2015), muitos robôs utilizam patas, outros são dotadas de esteiras e os mais comuns são aqueles que possuem rodas. Dependendo da aplicação, um robô explorador pode ter diversos acessórios que permitem ampliar suas capacidades na resolução de tarefas. Estes dispositivos podem ser sensores dos mais variados tipos, lanternas, acionadores ou qualquer outro dispositivo que possa ser acoplado a ele.

### **2.1.3 Robô Coletor**

Estes robôs foram desenvolvidos com o intuito de acessar locais perigosos, onde existam artefatos que possam causar explosões. Seu objetivo é encontrar o artefato e transportá-lo para um local seguro, onde possa ser detonado ou levá-lo até câmaras de contenção apropriadas para armazenamento de bombas (SKILLANO; KHATIB, 2015). Existem diversos tipos de robôs coletores, utilizados para aplicações militares, segurança pública e por esquadrões especializados em bombas. Estes robôs são um tipo de robô explorador, possuindo diversas características dos exploradores. Ambos são robôs controlados à distância e seus componentes são praticamente os mesmos (FOUNDATION, 2015). Segundo Benson (2008), o primeiro robô coletor foi desenvolvido pelo Tenente-coronel Peter Miller em 1972, cansado com a morte de colegas no desarmamento de bombas, Peter desenvolveu seu primeiro modelo para realizar esta tarefa. Tratava-se de um carrinho de mão motorizado que foi adaptado com um gancho capaz de puxar veículos que continham artefatos explosivos, sem colocar em risco a vida do especialista em desarmamento de bombas. Nos últimos anos os robôs coletores foram muito aperfeiçoados, de acordo com Ryder (2005), são capazes de lidar com explosivos construídos por especialistas, pois os movimentos executados pelo robô foram se tornando cada vez mais suaves, permitindo

assim que o artefato explosivo seja manipulado com maior segurança, sem que possa explodir ou dependendo da complexidade possa até mesmo ser desarmado. Algumas empresas, como a iRobotCorp, fabricam vários tipos de robôs exploradores, inclusive robôs destinado a coleta de materiais explosivos, que foram amplamente utilizados por tropas americanas para desarmamento de bombas em estradas na guerra do Iraque, Afeganistão, além de terem auxiliado no acidente nuclear de Fukushima. A Figura 2 apresenta o robô PackBot 510 produzido pela iRobotCorp e utilizado pelo exército dos Estados Unidos.

Figura 2 – PackBot 510



Fonte: iRobotCorp (2017).

#### **2.1.4 Custo de um Robô Coletor**

Os robôs coletores são fabricados em sua maioria por empresas norte-americanas e mesmo os modelos mais simples custam valores superiores ao de um carro esportivo. Os valores variam de U\$30.000 a U\$72.000 nos modelos pesquisados, levando em consideração a capacidade de cada um dos robôs, bem como as suas propriedades construtivas. No Quadro 1 são relacionados os principais robôs comercializados, bem como o valor aproximado de cada um deles.

#### **2.1.5 Componentes do Robô Coletor**

O robô que precisa carregar artefatos explosivos apresenta uma robustez um pouco maior do que um robô explorador, pois muitas vezes o artefato a ser movido pode possuir uma grande

Quadro 1 – Orçamento de robôs coletores

Fabricante	Modelo	Valor (U\$)
Dr Robot Inc	Jaguar V4 with arm	30.000,00
SuperDroid Robors	Super-Droid H2-S	32.500,00
Icor Technology	Mini-Caliber	50.000,00
Robotex	Avatar III	50.000,00
iRobot	510 Packbot	72.000,00

Fonte: o Autor.

massa. Esta robustez é importante para suportar pequenos impactos enquanto se desloca levando o artefato para um local seguro. Como em todo robô é necessário um cérebro, responsável pelo controle e pela transmissão ou recepção de dados. Este componente deve ser tão eficaz quanto de um robô explorador, devido a sua importância na execução das tarefas como um robô coletor. Em sua grande maioria os robôs coletores possuem mais de uma câmera em seu sistema, permitindo assim visualizar o objeto a ser coletado de vários ângulos diferentes sem mesmo precisar tocar o objeto. Os robôs exploradores atuais também utilizam mais de uma câmera em sua grande maioria (MANSOOR et al., 2001).

Todo robô que irá recolher um artefato explosivo necessita de um braço robótico para se movimentar em direção a carga explosiva e gentilmente segurá-la, para que durante o percurso não caia e acabe explodindo acidentalmente. Muitos robôs exploradores possuem braços robóticos de enorme precisão capazes de coletar e analisar amostras rochosas. Da mesma forma que os robôs exploradores, o robô coletor pode se locomover por rodas. Porém a maioria dos modelos utilizam esteiras capazes de transpor obstáculos muito maiores e subir escadas, que se mostram mais eficientes quando são duplas e articuladas (MANSOOR et al., 2001).

#### 2.1.5.1 Motores elétricos e caixas de redução

Atualmente os motores elétricos são amplamente usados na indústria, nas mais diversas aplicações; surgiram no século IX, conforme Sandin (2003), estes são conhecidos por sua capacidade de converter energia mecânica em elétrica ou vice-versa. Existem dois tipos de motores, os de corrente alterna e os de corrente contínua. Segundo Inovation (2005), estes últimos são aqueles que necessitam de uma fonte para alimentação, utilizam uma tensão não variável, ou seja, eles possuem dois polos sendo um negativo e outro positivo. Os motores de corrente contínua são alimentados por meio de uma corrente em que os polos não variam, apenas para inverter a rotação dos motores de corrente contínua ou CC é necessário inverter a polaridade, desta

forma ele irá girar no sentido horário ou anti-horário. As caixas de redução, como o próprio nome indica são responsáveis por reduzir a velocidade final do motor. Segundo Sandin (2003), fazendo com que ele gire mais lentamente seu eixo na saída da caixa. Automaticamente, ao diminuir a velocidade, é obtida uma grandeza física inversamente proporcional chamada torque, que é responsável pela força gerada na ponta do eixo, fazendo por exemplo com que o motor seja capaz de empurrar um peso muito maior do que conseguiria sem o acoplamento desta. Observa-se na Figura 3 um motor com a caixa de redução. Geralmente os motores são vendidos

Figura 3 – Motor e caixa



Fonte: Sandin (2003).

juntamente com a caixa, pois, na maioria das aplicações não se utiliza tanta velocidade e sem a redução proporcionada por esta, a força produzida nas rodas seria insuficiente para movimentar um robô com toda sua estrutura.

#### 2.1.5.2 Placa transistorizada para controle de PWM

De acordo com Sandin (2003), existem empresas que fabricam as mais diversas placas para montagens envolvendo dispositivos acionados por modulação por largura do pulso, ou PWM. Essas placas são utilizadas em aeromodelismo entre outros dispositivos controlados por rádio, e tem como principal função controlar a velocidade de motores de corrente contínua, que possuem elevado consumo de corrente. O torque fornecido pelo motor é proporcional a energia necessária para retirá-lo da inércia.

A placa escolhida para o desenvolvimento deste trabalho é a DBH-12V, que pode trabalhar com tensões que variam entre 5 à 12 Volts e corrente máxima de 30 Ampères por canal. A Figura 4 apresenta a placa utilizada. Além destas vantagens o módulo pesa cerca de 50 gramas e consegue trabalhar com frequências de até 16KHz.

#### 2.1.5.3 Servomotores

Segundo Sandin (2003), os servomotores possuem a característica de girar com precisão de acordo com o sinal que recebem, ao contrário dos demais motores em que não é possível esse

Figura 4 – Placa para controle de PWM



Fonte: XL Semi (2016).

controle. Existem vários tipos de servomotores, porém devido ao custo e velocidade os motores utilizados neste projeto são do tipo CC, ou seja, de corrente contínua. A Figura 5 apresenta um exemplo de servomotor. A capacidade de girar com certa precisão é proporcionada por um

Figura 5 – Exemplo de servomotor



Fonte: Adaptado de Inovation (2005)

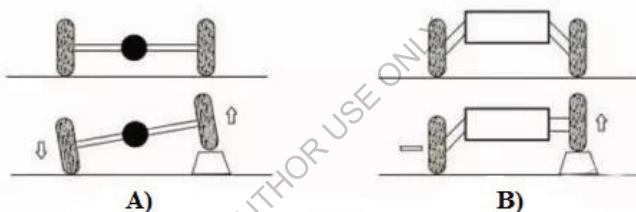
circuito montado no interior do servomotor CC, que consiste em uma placa com um circuito integrado e um potenciômetro, o giro do motor é controlado de acordo com o sinal recebido e da resistência apresentada no potenciômetro, de acordo com Inovation (2005).

#### 2.1.5.4 Chassi e suspensão

O chassi é a estrutura que sustenta todos os equipamentos e dispositivos que são abrigados pelo robô. Através dele são fixadas todas as demais partes, por isso, a sua grande importância. Conforme Siegwart e Nourbakhsh (2004), normalmente são fabricados em material leve e resistente que não apresente grandes deformações. Já a sua espessura irá determinar a quantidade de peso que pode ser suportado. A confecção pode ser em materiais como acrílico ou outros polímeros, para pequenos robôs e em alumínio ou aço para robôs de médio e grande porte. Segundo Adams (1992), a suspensão é acoplada diretamente ao chassi e é responsável

por absorver os impactos sofridos pelas rodas, fazendo com que o máximo destes impactos sejam absorvidos sem que afetem o chassi. Existem vários tipos de suspensões disponíveis para serem utilizadas e cada vez mais as indústrias automobilísticas pesquisam e desenvolvem novos conceitos, capazes de gerar mais conforto e reduzindo ainda mais os impactos sofridos pela estrutura. As suspensões utilizadas foram duas: a do tipo independente e a de eixo semi-rígido. Nas suspensões de eixo, as rodas se inclinam a medida que o obstáculo atinge as rodas, fazendo com que uma suba e outra acompanhe parte do movimento no sentido oposto, apresentado na Figura 6 em (A). Esse tipo de suspensão não é muito indicado para ser utilizado em rodas que exercem tração, pois se uma das rodas ficar suspensa a outra fará com que a estabilidade do veículo diminua consideravelmente. Já nas suspensões independentes, como o próprio nome

Figura 6 – Exemplo de suspensão de eixo (A) e independente (B)



Fonte: Adaptado de Adams (1992)

sugere, cada roda é capaz de sofrer e absorver impactos individualmente, conforme apresentado na Figura 6 em (B). De acordo com Adams (1992), isso permite ao robô quando atravessar um obstáculo, se manter firme ao chão quando as rodas atingirem o mesmo.

#### 2.1.5.5 Baterias

Atualmente existem diversos tipos de baterias, para as mais diversas aplicações e com as mais diferentes tensões e capacidades de carga (BRAUNL, 2003). Devido a essa variedade gigantesca é que será abordada apenas a bateria utilizada neste projeto. Conhecida como Li-Íon, ela é amplamente utilizada por todos nós, pois a maioria dos notebooks, celulares e câmeras digitais no mercado utilizam este tipo de baterias em associação. Entre suas principais características pode-se citar a grande capacidade de carga e o peso reduzido. Apesar de terem várias tensões de trabalho, as abordadas aqui, conhecidas pelo modelo 18650 trabalham com uma tensão de 3,7 V e podem fornecer em média 2 A/h de corrente. Segundo Braunl (2003), por meio de associações das baterias em série, é possível aumentar a tensão das baterias para serem

utilizadas por motores e circuitos, da mesma forma que associar as baterias paralelamente fará com que a corrente fornecida possa ser maior, aumentando assim a carga disponível. É possível utilizar ambas as formas de associação, obtendo tanto uma tensão maior, quanto uma corrente maior.

#### 2.1.5.6 Placas Reguladoras de tensão

As placas reguladoras de tensão são utilizadas para elevar ou reduzir a tensão fornecida a um circuito, permitindo assim, que baterias associadas com tensões diferentes das utilizadas nos circuitos possam alimentá-los. Estas placas, em sua maioria, são alimentadas com corrente contínua fornecida por baterias e possuem circuitos que permitem a regulagem da tensão na saída de acordo com a necessidade do projeto. A placa utilizada neste projeto possuí uma tensão variável de entrada de 4 a 38 Volts e uma saída variável de 1,25 a 36 Volts, capaz de fornecer uma corrente de até 5 ampères. Segundo o fabricante XL Semi (2012), o circuito integrado utilizado na placa é o XL4005, tem como principais características o tamanho reduzido e a simplicidade de uso, possuindo apenas cinco pinos, que são utilizados como: entrada, saída, aterramento, uma chave e um pino de retroalimentação que permite a regulagem da tensão de saída. Montado sobre uma placa, o circuito XL4005 possuí um potenciômetro multivoltas que permite realizar o ajuste da tensão na saída da placa. A Figura 7 apresenta a placa utilizada.

Figura 7 – Placa Reguladora XL4005



Fonte: XL Semi (2012).

#### 2.1.5.7 Computador Raspberry Pi

Segundo Pi (2005), Raspberry Pi é um computador muito pequeno, desenvolvido pela fundação de mesmo nome, no Reino Unido, no início do ano de 2012. Pode ser utilizado em diversas aplicações devido ao seu tamanho reduzido e ao baixo consumo de energia. Ele não possui monitor ou dispositivos de entrada/saída embutidos, ao invés disso, possui várias entradas e saídas que permitem conexão de dispositivos como teclado, mouse, monitor, entre outros.

Ele é capaz de navegar na internet e reproduzir vídeos em alta definição, sendo que geralmente roda sistemas operacionais Linux. De acordo com Pi (2005), entre as principais características do Raspeberry Pi pode-se citar na versão 3 traz inovações como Wi-Fi 802.11 e Bluetooth 4.1 e utiliza um processador BCM2837 de 4 núcleos ARM Cortex-A53 a 1.2GHz, equipado ainda com 1GB de memória volátil. O objetivo do Raspberry é ser um computador como outro qualquer, mas nele existem pinos GPIO que podem ser utilizados para conexão de sensores e atuadores, além de comunicação com outros dispositivos. Ve-se na Figura 8 o Raspberry Pi 3

Figura 8 – Raspberry Pi 3 Model B



Fonte: Raspberry Pi Foundation.

Modelo B+, possuí uma entrada para câmera próxima a saída HDMI, quatro portas USB, uma porta de alimentação micro-USB, uma saída para display de baixa resolução, uma saída HDMI para vídeo de alta resolução, saída de áudio, conexão ethernet e uma entrada para cartão de memória (não ilustrada na figura) que fica na parte inferior da placa.

#### 2.1.5.8 Módulo de câmera Raspberry Pi

A câmera desenvolvida pela empresa Adafruit para ser utilizada juntamente com o Raspberry Pi possuí duas versões diferentes, sendo possível conectá-las diretamente no Raspberry através da entrada CSI (interface para câmera serial), utilizada para este fim. A versão 2 permite uma resolução de até 1080p como uma taxa de 30 quadros por segundo e uma densidade de 8 Megapixels. A principal característica desta câmera é seu tamanho (cerca de 2,5 cm de largura por 2,5 cm de altura) e peso reduzido (em torno de 3 gramas), por ser um módulo, é montada sobre uma placa e conectada ao Raspeberry por um cabo do tipo *flat*. Existe também uma câmera paralela que pode ser utilizada em conjunto com o Raspberry e possuí um custo menor. É possível verificar na Figura 9 a câmera montada sobre a placa.

Figura 9 – Módulo de câmera



Fonte: Raspberry Pi Fundation.

#### 2.1.5.9 Microcontrolador Atmel ATmega2560

Conforme Wise (2005), atualmente existem muitos microcontroladores dos mais diversos fabricantes, sendo possível encontrar no mercado um microcontrolador com as características necessárias para o desenvolvimento de um protótipo ou um projeto qualquer. Segundo Pereira (2007), um microcontrolador é o dispositivo dotado de várias portas de entrada e saída. Estas portas podem ser utilizadas para conectar sensores para leitura de sinal, ou acionar um led, por exemplo. Isso acontece de acordo com a configuração realizada em seu *firmware*. Para Pereira (2007), *firmware* é um *software* que é rodado dentro de um sistema embarcado com o intuito de realizar uma determinada tarefa programada. Como por exemplo, executar as etapas para lavar roupas em uma máquina com esse propósito. Segundo Atmel (2017), existem diversos microcontroladores vendidos pela Atmel em sua linha conhecida como megaAVR, sendo abordado neste projeto o modelo ATmega2560. Utiliza um núcleo 8-bit AVR, rodando a uma velocidade de 16 Mhz. Também conta com uma memória flash de 256 Kb (para armazenar códigos a serem executados), memória RAM de 8 Kb (que deve ser utilizada para armazenar dados temporários) e ROM de 4 Kb (que permite armazenar dados de boot). Possui cem pinos dentre os quais pode-se citar: entradas/saídas digitais, PWM (Modulação por Pulso), comunicação UART, I2C e SPI entre outras funcionalidades distribuídas pelos seus 11 *ports*. Vê-se na Figura 10 o microcontrolador Atmel, que será utilizado neste projeto, nota-se que o encapsulamento deste microcontrolador é do tipo SMD, devendo ser montado sobre a superfície da placa.

#### 2.1.5.10 Placa adaptadora SMD para DIP

Devido aos componentes SMD terem seu tamanho reduzido, inviabiliza sua utilização em *protoboard*s e outros testes de prototipação de circuitos. Desta forma, para conseguir acessar

Figura 10 – Atmega2560



Fonte: Atmel (2017).

os pinos da placa, devido ao seu tamanho reduzido, utiliza-se um adaptador que permite a distribuição dos pinos do circuito integrado em um soquete do tamanho utilizado em modelos DIP. A Figura 11 apresenta a placa utilizada, é interessante notar que ela suporta circuitos integrados SMD de 32 até 100 pinos (BRAUNL, 2003).

Figura 11 – Adaptador  
SMD/DIP



Fonte: Hayoui Electronics (2017).

#### 2.1.5.11 Xbee Explorer Adaptador USB

Para tornar possível a comunicação entre o microcontrolador e o computador é necessária a utilização de uma placa, que tem como função principal realizar a interface entre os módulos XBee e as USART do computador e do microcontrolador. O chip principal utilizado nesta placa é fabricado pela FTDI. De acordo com o fabricante, o FT232R é disponibilizado em um encapsulamento de 28 pinos e todo o protocolo USB é administrado pelo dispositivo. Por meio das características presentes no circuito se torna possível trabalhar com tensões de 3,3 V utilizadas pelos módulos Xbee e 5 V presentes nas portas USB. A placa utilizada neste projeto possui quatro *leds* que estão conectados aos pinos, os quais recebem e transmitem os dados (RX

e TX), permitindo desta forma, identificar quando os dados estão trafegando entre a USB e o XBee. Vê-se na Figura 12, a placa utilizada neste projeto.

Figura 12 – Módulo FT232R



Fonte: FTDI Chip (2017).

#### 2.1.5.12 Sensores

Conforme Murphy (2000), sua principal função é captar informações do ambiente para serem transformadas e interpretadas por dispositivos eletrônicos dos mais variados, por meio deles é que sabemos qual é a temperatura, ou umidade em determinado local, se um motorista ingeriu álcool, ou se é necessário acender as luzes de um poste porque ficou escuro em uma avenida. Muitos sensores não terão utilidade neste projeto, portanto abordam-se apenas os sensores de: temperatura e fumaça. De acordo com Mazidi (2014), o sensor de fumaça MQ-135, é aquele que identifica determinados gases tóxicos no ambiente em que se encontra, sendo capaz de identificar a presença de vários gases como: amônia, dióxido de carbono, benzeno, óxido nítrico, e também fumaça ou álcool. É importante ressaltar que trata-se de um sensor simples que não possui a capacidade de identificar a concentração de cada um dos gases e sim a presença deles e em determinada faixa de concentração. O manual do fabricante especifica a precisão de leitura que varia entre 10 ppm até 1000 ppm, este sensor é apresentado na Figura 13 em A, montado sobre um módulo. Segundo Miyadaira (2013), o sensor de temperatura DS18B20 tem por objetivo determinar se está frio ou calor, obtendo este valor através do ar, ou dependendo do tipo de sensor, também pode identificar a temperatura do material com o qual estiver em contato seja ele sólido ou líquido. Geralmente para líquidos o sensor parece uma haste metálica presa a um fio e para sólidos, pode ser muito simples e com aparência de um transistor, como o LM35, fabricado pela National Semiconductor. O sensor DS18B20 utilizado neste projeto, segundo o manual fornecido pelo fabricante Integrated (2017), consegue mensurar temperaturas que variam de  $-55^{\circ}\text{C}$  até  $125^{\circ}\text{C}$ , com uma precisão de mais ou menos  $0,5^{\circ}\text{C}$  e uma resolução

Figura 13 – Sensores de gases e temperatura



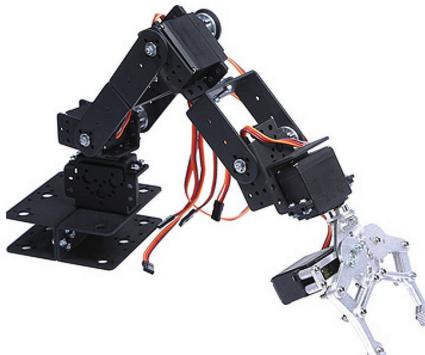
Fonte: Maxim Integrated (2015).

de até 12 bits. Este sensor é apresentado na Figura 13 em B.

#### 2.1.5.13 Braço robótico

Os braços robóticos são atualmente muito utilizados na indústria para soldagem e montagem de peças. Os primeiros surgiram com a Revolução Industrial e vêm sendo cada vez mais aperfeiçoados (MURPHY, 2000). Com o aumento da precisão, existem braços robóticos, que segundo especialistas, são capazes de construir pontes inteiramente sozinhos, apenas soldando à estrutura da mesma (INOVATION, 2005). O braço robótico utilizado neste projeto é muito mais modesto, uma vez que estes dispositivos de última tecnologia são de alto custo; e o objetivo do presente trabalho é que ele seja capaz de realizar tarefas simples, como retirar um pequeno obstáculo ou apertar um botão. Este dispositivo possui quatro eixos de rotação, permitindo que ele se movimente  $180^\circ$  com um movimento de giro, ou seja, em sentido horário e anti-horário, além de três articulações como se fosse um ombro e cotovelo e a mão, se comparado ao braço humano. Na extremidade do braço existe uma garra capaz de prender e segurar objetos de pequeno porte. Para realizar os movimentos das articulações do braço são utilizados servomotores, estes são ativados por um pulso de comprimento modulado, fornecidos por uma saída digital (SANDIN, 2003). Dependendo do tempo em nível alto recebido, o servomotor irá atuar em determinada quantidade de graus, sendo  $0^\circ$  a  $180^\circ$  na maioria das vezes. A garra irá funcionar da mesma forma, conectada por meio de engrenagens ao servomotor irá abrir ou fechar de acordo com o pulso recebido. Vê-se na Figura 14 o modelo de braço em questão, sendo possível observar que este já está com os servomotores acoplados bem como a garra, os fios deverão ser dispostos em um chicote que não restrinja a mobilidade do braço em todas suas extensões de movimento.

Figura 14 – Braço robótico de 6 eixos



Fonte: Sandin (2003).

#### 2.1.5.14 Supervisório

Supervisório é um *software* que roda em um computador e geralmente é utilizado para monitoramento e controle de dispositivos de *hardware*. Conforme Inovation (2005), por meio dele é possível, por exemplo, identificar se um gerador está funcionando adequadamente, quanta energia está sendo gerada, qual a temperatura dos componentes envolvidos, além de inúmeras outras grandezas que podem ser medidas e monitoradas.

Utilizando um *software* de controle é possível também controlar um robô usando um computador ou celular, para que ele realize determinadas ações de acordo com a vontade do controlador. Por meio deste, podem ser enviados ou recebidos dados, dependendo do meio de comunicação escolhido, seja ele cabeados ou sem fio. A Figura 15 mostra um exemplo de supervisório para Android para controle do robô PackBot desenvolvido pela empresa iRobot.

#### 2.1.5.15 Comunicação de Dados via Wi-fi

A comunicação de dados sem fio que teve início no século IX, com a comunicação de um telégrafo sem fio na Europa, evoluiu aos dias atuais onde quase todos os dispositivos possuem não apenas uma comunicação sem fio, mas várias delas. De acordo com Wise (2005), as mais utilizadas atualmente: são a Wi-fi, GPRS e Bluetooth, além, é claro, da transmissão de dados via satélite, seja para utilização de georeferenciamento ou outros tipos de transmissão e recepção de dados. Abordar-se-á aqui o tipo de comunicação utilizado entre o robô e supervisório utilizando a tecnologia Wi-fi, onde o notebook deverá se conectar com o robô através

Figura 15 – Exemplo de sistema supervisório



Fonte: iRobot.

de uma rede sem fio. Segundo Molloy (2011), os *Acess Point* são pontos de conexão sem fio, entre outros dispositivos sem fio ou cabeados e ainda dependendo do caso um Gateway, se necessário o acesso à internet. Há diversos modelos de AP, alguns deles permitem a replicação do sinal para aumentar a potência em locais onde o *Acess Point* está com o sinal fraco. Existem os mais diversos fabricantes de repetidores de sinal, no Brasil estes equipamentos precisam ser homologados pela Anatel para que possam ser comercializados no mercado.

#### 2.1.5.16 Comunicação de Dados via XBee

XBee é o nome de uma família de módulos compatíveis utilizados para transmissão de dados sem fio, fabricados pela empresa Digi International, com as mais diversas potências e aplicações. Uma das principais características apresentadas por estes módulos é a sua praticidade de conexão, utilizando apenas Vcc com tensão de 3.3V, terra e dois outros pinos utilizados para comunicação: para envio (Tx) e recebimento (Rx) dos dados.

Os módulos podem vir com algumas antenas diferentes de acordo com a necessidade do projeto e a intensidade do sinal desejada. Outra característica apresentada por estes módulos é a sua baixa taxa de transmissão de dados não ultrapassando 250 Kbps na maioria dos modelos. Existem diversos modelos de módulos XBee, compatíveis entre si, os modelos utilizado neste trabalho são XBee Series 2, apresentado na Figura 16.

Figura 16 – Módulo XBee



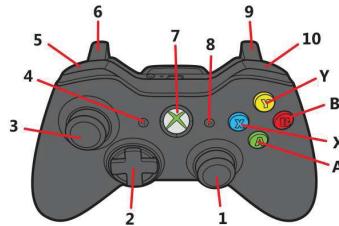
Fonte: Digi Internacional Inc.  
(2009).

Segundo Digi (2014), a grande vantagem na utilização desses módulos, se dá ao fato de seu alcance ser muito expressivo pela energia consumida pelos dispositivos. Pois eles conseguem enviar os dados a uma baixa velocidade, porém o consumo de energia também é baixo tendo em vista o alcance do sinal, operam a uma frequência ISM 2,4 Ghz.

#### 2.1.5.17 Controle Xbox 360

O controle Xbox 360 é utilizado no vídeo game produzido pela Microsoft, segundo a fabricante é possível conectar o mesmo na porta USB de um computador. A Figura 17, apresenta um controle Xbox 360, este possui onze botões (4 a 10, mais X,Y,A,B) e três controles direcionais (sendo dois analógicos) que enviam cerca de 9 sinais diferentes cada, estes geralmente são utilizados com o intuito de movimento, pois permitem trabalhar em pelo menos dois eixos cada.

Figura 17 – Controle Xbox 360



Fonte: Microsoft Xbox.

De acordo com Microsoft (2017), alguns dos botões possuem o que é chamado de interface de contexto sensitiva, de , assim o sinal enviado pelo botão pode ser de maior ou menor

intensidade, de acordo com a pressão sobre o mesmo; isso pode auxiliar em tarefas que exijam maior precisão e sensibilidade. O acessório é totalmente compatível com a plataforma Windows e ao acionar um botão é gerado um evento que pode ser tratado pelo sistema operacional, permitindo assim a interação com jogos e outros dispositivos que possam ser controlados pelo computador.

#### 2.1.5.18 Linguagens de Programação

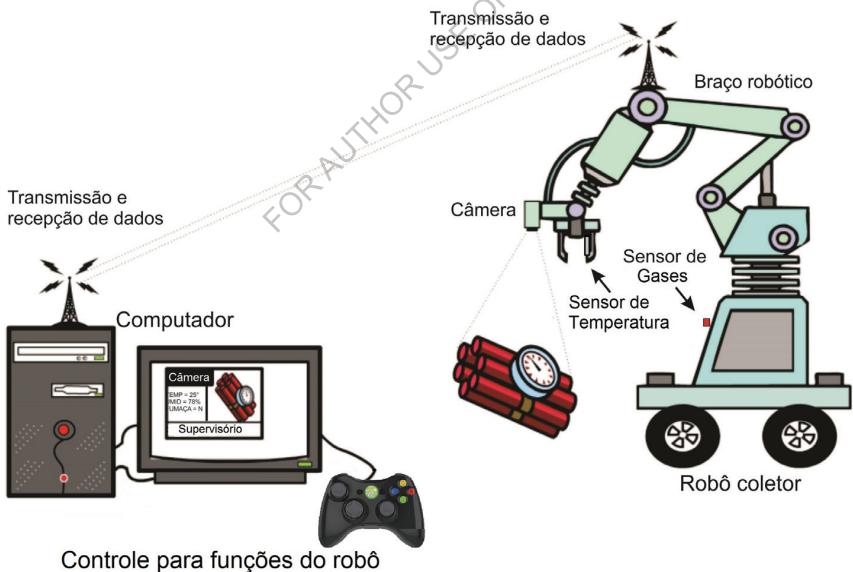
As linguagens de programação utilizadas para o desenvolvimento do trabalho são o C e o C#. Segundo Deitel e Deitel (2011), a linguagem C foi criada em 1972 por Dennis Ritchie, é uma linguagem estruturada que foi criada para desenvolvimento de sistemas operacionais, se tornando posteriormente muito popular e sendo compilada nas mais diversas plataformas, inclusive as de sistemas embarcados. A linguagem C# é uma linguagem compilada, desenvolvida pela Microsoft e é baseada em conceitos de C++, como a orientação à objetos, sendo uma das principais características herdadas (DEITEL; DEITEL, 2006). Por ser uma linguagem compilada, necessita de um compilador que na maioria das vezes vem incluído com a IDE de desenvolvimento, que é o conjunto de ferramentas que permite ao programador organizar, compilar e fazer o *debug* do projeto se necessário. Para realização deste projeto foi optado por utilizar o Ambiente de Desenvolvimento Integrado chamado Visual Studio, que é o principal para ser usado com ferramentas .NET fornecidas pela Microsoft (DEITEL; DEITEL, 2006). Segundo Waslawick (2011), a linguagem de modelagem utilizada para o desenvolvimento deste trabalho é a UML, traduzindo para português (Linguagem de Modelagem Unificada), pode ser utilizada para descrever coisas, e por meio desta serão criados diagramas para entendimento de como deve ser o desenvolvimento do *software*, ao mesmo tempo em que é elaborada a documentação.

### 3 METODOLOGIA

A pesquisa realizada é do tipo quanti-qualitativa e a modalidade de pesquisa do tipo experimental. Para realização deste trabalho o método adotado é hipotético-dedutivo, onde são levantadas hipóteses de uma forma geral e depois analisados dados particulares (MARTINS, 2000).

O sistema completo, a ser desenvolvido neste trabalho é apresentado na Figura 18, a comunicação é realizada entre o supervisório e o robô coletor, permite o controle e manipulação das ações. No supervisório são exibidos os dados coletados pelo robô como: imagens e a leitura do sensor de temperatura. Por meio destes dados existe a possibilidade de identificar se o material coletado está aumentando de temperatura, auxiliando assim, na identificação de alterações no material enquanto é coletado (MANSOOR et al., 2001).

Figura 18 – Sistema completo com robô coletor e supervisório



Fonte: Adaptado de Wirelles Bomb Disposal Robot

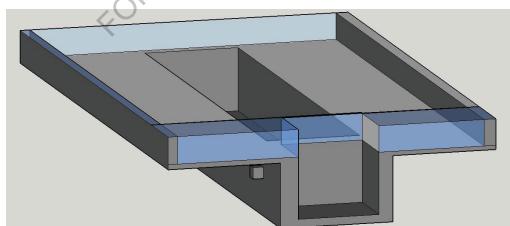
### 3.1 PROJETO E CONSTRUÇÃO DO CHASSI DO ROBÔ

#### 3.1.1 Projeto CAD do Robô

Um modelo desenvolvido com a ferramenta CAD, permitiu desenhar virtualmente os componentes utilizados no robô, auxiliando a projetar e visualizar a sua estrutura. Por meio deste modelo foram escolhidos os materiais utilizados na estrutura do robô e posicionados os componentes de forma que suas capacidades sejam mais eficientes. Foi projetado um local especial na parte frontal do robô para fixação do braço robótico, esse espaço se mostrou adequado aos movimentos das articulações do braço.

Inicialmente foi projetado o chassi, pois é o local de fixação de todos os componentes. O projeto do chassi, que deveria ter um formato trapezoidal em corte lateral, foi alterado para um modelo mais retangular, fazendo com que a suspensão e os motores ocupassem uma área grande útil no interior do chassi. Os principais motivos para a alteração foram o alto custo dos motores e caixas de redução e o melhoramento do centro de gravidade do robô, permitindo maior estabilidade ao locomover-se em terrenos inclinados. Ao longo de toda construção do chassi foram realizadas alterações para fixação da suspensão, possibilidade de instalação de rodas maiores e fixação da câmera. A Figura 19 apresenta o projeto inicial do chassi do robô.

Figura 19 – Chassi do Robô



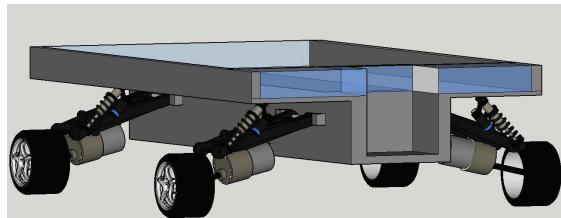
Fonte: O Autor.

A primeira parte que sofreu alterações do projeto, foi a suspensão do robô, devido ao tamanho e posicionamento dos amortecedores e molas foi necessário construir um prolongamento no chassi, permitindo a fixação da suspensão em três pontos, sendo dois na parte frontal que abrigam os motores e um na parte traseira. É por meio da suspensão que o robô consegue transpor alguns obstáculos com maior facilidade. Foram utilizados braços de alumínio e um sistema de molas helicoidais, juntamente com um amortecedor à óleo, conforme o projetado.

Na parte superior da suspensão foram fixados os motores com as caixas de redução, ao

contrário do projeto onde os motores ficavam na parte inferior, uma peça cilíndrica foi fabricada para abrigar os motores e permitir sua movimentação. A Figura 20 apresenta a concepção inicial do sistema de suspensão, motores e rodas.

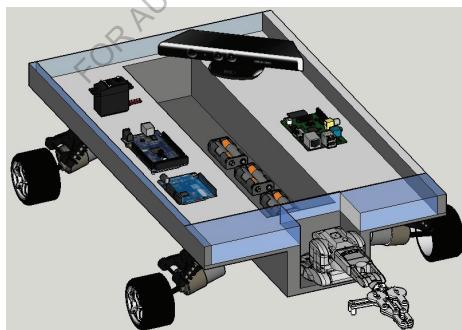
Figura 20 – Sistema de suspensão, motores e rodas



Fonte: O Autor.

O próximo passo consistiu em colocar alguns componentes eletrônicos, para que fosse possível montar os circuitos necessários para o robô operar e as baterias para alimentar os circuitos. Observa-se na Figura 21 a parte interna do robô, onde devem ser fixados todos os componentes eletrônicos.

Figura 21 – Componentes eletrônicos

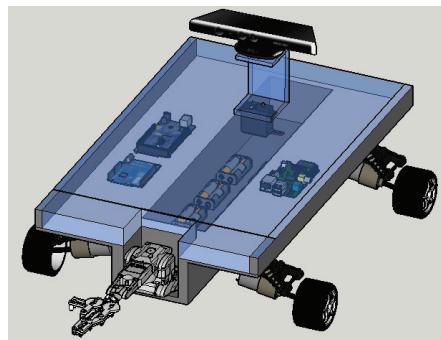


Fonte: O Autor.

Na última etapa, uma tampa em acrílico deve fechar a parte superior do robô, deixando-o com um aspecto mais bonito, além de proteger os circuitos contra umidade e poeira. Apresenta-se a versão que foi proposta para montagem na Figura 22.

Na prática, o desenvolvimento do chassi não teve apenas alteração no formato, mas foi adaptado conforme as necessidades surgiram. O sistema de suspensão precisou de cinco montagens para funcionar conforme será apresentado a seguir.

Figura 22 – Protótipo do Robô concluído



Fonte: O Autor.

### 3.1.2 Molde para chassis

O material escolhido para construção do chassi foi fibra de vidro, devido ao seu baixo custo e fácil manutenção, além de outras propriedades interessantes como: resistência e maleabilidade. Inicialmente foi desenvolvido um molde fabricado com madeira, cola para madeira e tecido, por meio deste, foi desenvolvida a forma em que a maior parte do chassi deveria assumir. Apresenta-se o molde fabricado na Figura 23.

Figura 23 – Molde para chassis



Fonte: O Autor.

Após o molde estar pronto, foi adicionado uma camada de tecido para permitir que a resina e a fibra ficassesem sobre a mesma, evitando assim que toda a área tivesse que ser fechada com madeira, o que iria deixar o trabalho muito mais lento e aumentar o peso do chassi desnecessariamente. A Figura 24 retrata a segunda etapa.

Figura 24 – Molde para chassi com tecido



Fonte: O Autor.

### 3.1.3 Fabricação do chassi

Concluído o molde, foram adicionadas duas camadas de resina e fibra, para acrescentar resistência ao molde, as próximas etapas incluiram ajustes nas peças e aplicação de massa plástica. Após todo o chassi ser lixado, as peças foram pintadas e a suspensão pronta para ser montada, o resultado é apresentado na Figura 25.

Figura 25 – Chassi para montagem da suspensão



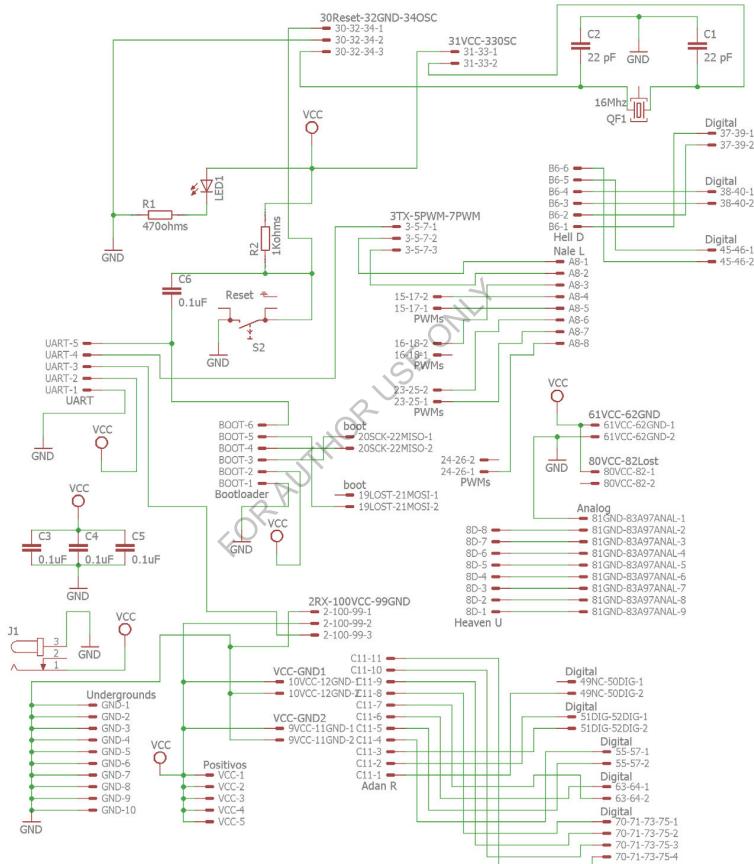
Fonte: O Autor.

## 3.2 PLACA MICROCONTROLADORA DESENVOLVIDA COM ATMEGA2560

Para o desenvolvimento da placa microcontroladora utilizada neste projeto, foram impressas e corroídas três versões de placas, as duas primeiras versões tinham trilhas muito finas e que mesmo que ficassem dispostas equidistantes umas das outras, acabavam se unindo durante a impressão na placa. As trilhas tinham largura de 0,4mm que é a mesma largura apresentada pelos pinos do microcontrolador ATmega2560.

Para confecção da placa foram consultados documentos disponibilizados pela fabricante Atmel, além de artigos relacionados a utilização deste microcontrolador entre outros projetos de placas encontrados. O circuito desenvolvido com a ferramenta Eagle é apresentado na Figura 26, e em seguida são explicados todos os detalhes no circuito utilizado. Dentre os principais

Figura 26 – Diagrama do circuito da placa microcontroladora



Fonte: O Autor.

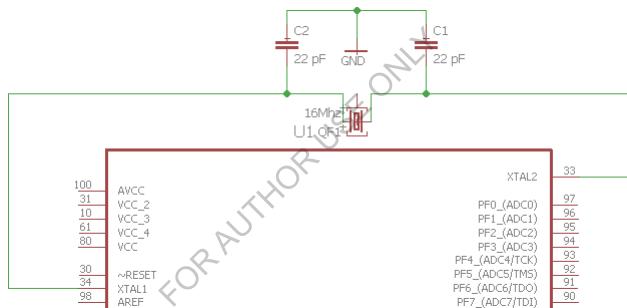
pinos relacionados ao projeto podemos citar os conectados ao Vcc sendo eles os de número: 10, 31, 61, 80 e 100, bem como os pinos conectados ao Terra sendo eles: 11, 32, 62, 81 e 99. Entre estes pinos o *datasheet* disponibilizado pela fabricante recomenda a utilização de 3 capacitores como filtro de *bypass*, com os valores de 0,1 uF, filtrando o sinal recebido pelo microcontrolador.

evitando possíveis ruídos de CA entre outras variações de tensão em toda a placa. Outros pinos muito utilizados são os responsáveis pela comunicação via UART, sendo eles: Terra, Vcc, RX0, TX0 e uma conexão com o *Reset* que possui ligado em série um capacitor de 0.1 uF conforme orientado pelo fabricante.

### 3.2.1 Circuito do cristal oscilador

De acordo com Atmel (2017), para utilização de um cristal cerâmico com *clock* de 16 Mhz, deve ser conectado aos pinos 33 e 34 do microcontrolador e em paralelo com cada um destes pinos e o terra deve ser conectado um capacitor de 22 pF. Esta parte do circuito é apresentada na Figura 27.

Figura 27 – Circuito do cristal oscilador



Fonte: O Autor.

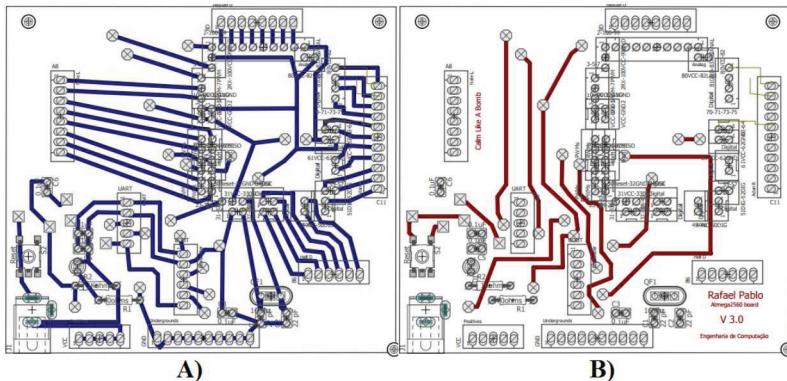
### 3.2.2 Desenho da placa em duas camadas

A placa foi desenvolvida com a ferramenta Eagle, que permite desenhar o circuito com os devidos componentes. Após serem realizadas todas as conexões dos componentes no circuito a placa foi roteada e foram então gerados arquivos que foram impressos sobre a placa que representa o circuito desenvolvido com o software. A Figura 28 apresenta em A o lado inferior da placa e em B o lado superior.

### 3.2.3 Corrosão da placa

Para realizar a corrosão da placa de fenolite e deixar apenas as trilhas que são parte do circuito foi utilizada uma solução de cloreto férrico ( $FeCL_3$ ), diluída em água com uma concentração de 30 a 40%. A placa foi mergulhada na solução e virada para que a corrosão

Figura 28 – Desenho da camada superior e inferior da placa desenvolvida



**Fonte:** O Autor.

atingisse de maneira mais uniforme possível os dois lados da placa, esse processo é apresentado na Figura 29, decorridos cerca de dez minutos a placa estava totalmente corroída.

Figura 29 – Corrosão da placa de fenolite



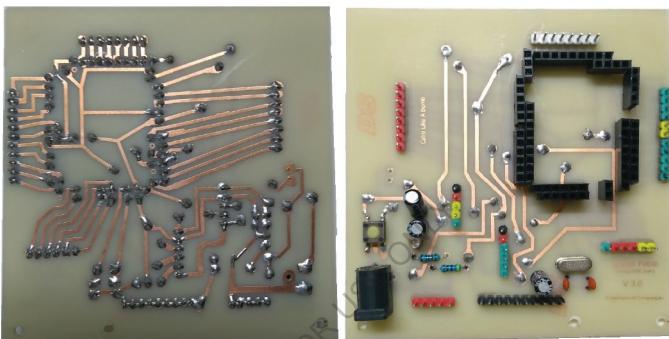
Fonte: O Autor.

### **3.2.4 Soldagem de componentes**

Os componentes foram soldados sobre a placa, devido a ser um circuito impresso com duas camadas existem diversos pontos onde as trilhas iniciam em um lado da placa e terminam em outro. Foi possível posicionar todos os componentes para que ficassem do mesmo lado da placa, como capacitores, resistores e conectores. Muitos *jumpers* foram utilizados para permitir

que as trilhas mudassem de lado ao longo da placa, demonstrando que seriam necessárias mais camadas para obter uma qualidade superior. O circuito integrado ATmega2560 foi o componente que envolveu mais tempo durante a soldagem, por se tratar de um componente SMD e conter 100 pinos com um tamanho reduzido, foi necessário montá-lo sobre o adaptador e posteriormente encaixar na placa desenvolvida. A Figura 30, apresenta a placa após a soldagem dos componentes nos lados inferior a esquerda e superior a direita.

Figura 30 – Soldagem dos componentes da placa



Fonte: O Autor.

### 3.2.5 Testes da placa microcontroladora

Foram realizados alguns testes com o intuito de averiguar o funcionamento da placa desenvolvida, o primeiro deles foi o teste de continuidade aplicado em todas as trilhas e em todos os pontos da placa, para se certificar que havia passagem de corrente, principalmente devido aos 25 *jumpers* existentes. Os pinos utilizados no circuito também foram testados, desde o funcionamento de portas digitais com PWM até a leitura de portas analógicas, além da comunicação via USART utilizando RX e TX, sendo estes os pinos 2 e 3 respectivamente.

## 3.3 CIRCUITOS DO ROBÔ

Todo robô possui vários componentes eletrônicos que devem ser conectados ou configurados de maneira adequada, para que seu funcionamento seja satisfatório. Conforme Inovation (2005), estes componentes são conectados e formam um circuito que varia em complexidade, de acordo com as capacidades do robô, assim quanto mais ferramentas ele possuir, maiores serão seus circuitos elétricos.

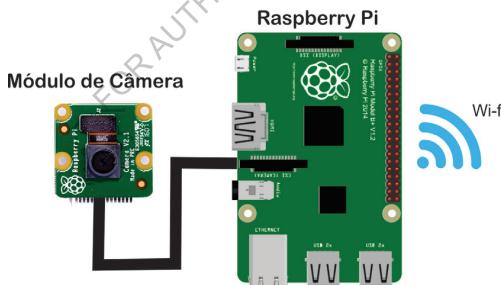
A comunicação entre Raspberry Pi e a placa microcontroladora se dá através de conexão XBee, mas existem também conexões do tipo seriais entre Raspberry Pi e a câmera conectada. A grande maioria das conexões está nas portas da placa microcontroladora que são distribuídas para os demais dispositivos disponíveis no robô. Cada servomotor possui dois fios para alimentação, assim como os sensores e demais placas, as tensões podem variar de acordo com a necessidade, todavia as conexões desses dispositivos são necessárias.

Para facilitar o entendimento do trabalho os circuitos são divididos em seções referentes a cada um dos componentes, o circuito completo encontra-se no Apêndice A.

### 3.3.1 Conexões do Raspberry Pi

O Raspberry Pi, por ser a placa de maior processamento, terá muitas conexões, entre elas pode-se destacar: a entrada da câmera e a transmissão de dados através de Wi-fi. A referida placa também estará conectada pela porta USB a placa microcontroladora desenvolvida. O objetivo principal é alimentar a placa com uma tensão de 5 volts. Na Figura 31 apresenta-se o esquemático de conexões do dispositivo Raspberry Pi.

Figura 31 – Componentes conectados ao Raspberry Pi



Fonte: O Autor.

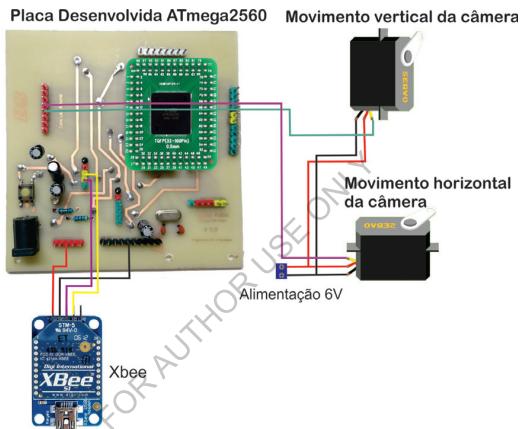
### 3.3.2 Conexões da placa microcontroladora desenvolvida

A placa desenvolvida utilizando o microcontrolador ATmega2560, irá controlar os motores responsáveis por locomover o robô, fazer a leitura dos sensores e movimentar a câmera, devido a quantidade de dispositivos envolvidos esse circuito foi dividido em seções para facilitar o entendimento.

### 3.3.2.1 Circuito dos servos para movimentar a câmera

Nas portas PB4 e PB5 da placa microcontroladoras são conectados dois servomotores responsáveis por girar a câmera para Raspberry Pi em dois eixos, permitindo ao robô além de observar a frente possa visualizar cerca de 200 graus no sentido horizontal e 150 graus na vertical. As portas PB4 e PB5 podem ser configuradas como saídas PWM, que permite o controle dos servomotores. A Figura 32 apresenta o circuito utilizado para movimentação da câmera.

Figura 32 – Circuito utilizado para movimentar a câmera

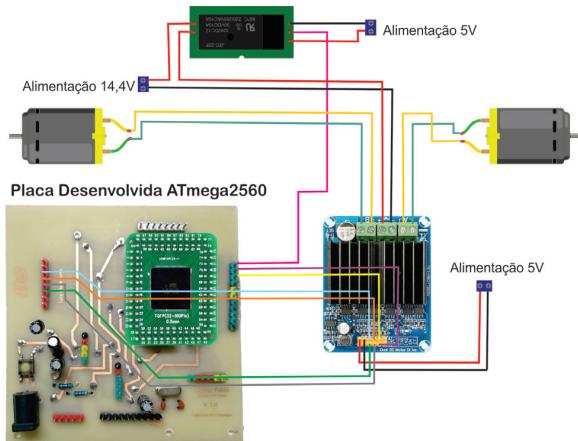


Fonte: O Autor.

### 3.3.2.2 Circuito da ponte H - Alteração para controle PWM

Os motores responsáveis por locomover o robô são conectados a um módulo da ponte H, permitindo acionar os motores em duas direções e por ser transistorizada é possível variar a velocidade em cada um deles. As saídas digitais responsáveis pelo controle da ponte H são conectados nos pinos PB6, PB7, PE4 e PE5, controlando o sinal de cada motor separadamente. Esse sistema é muito comum em circuitos de robôs locomovidos por rodas, e funciona basicamente enviando um sinal que permite acionar o motor em um determinado sentido e outro PWM que permite a variação da velocidade. A Figura 33 apresenta o circuito utilizado neste projeto. Foi necessário adicionar ainda um módulo de relé, responsável por controlar a alimentação do módulo da ponte H enquanto o sistema não esteja totalmente operante. Essa medida foi adotada pois algumas vezes, o robô começava a se locomover assim que seus circuitos eram

Figura 33 – Circuito da ponte H



Fonte: O Autor.

alimentados. Anteriormente a ponte H deste projeto utilizava um módulo com 4 relês que acabaram sendo substituídos por um módulo transistorizado, permitindo assim a além do controle da direção dos motores também a variação de velocidade dos mesmos. Essa medida foi adotada para que o controle do robô fosse mais preciso e eficiente.

### 3.3.2.3 Circuito do braço robótico

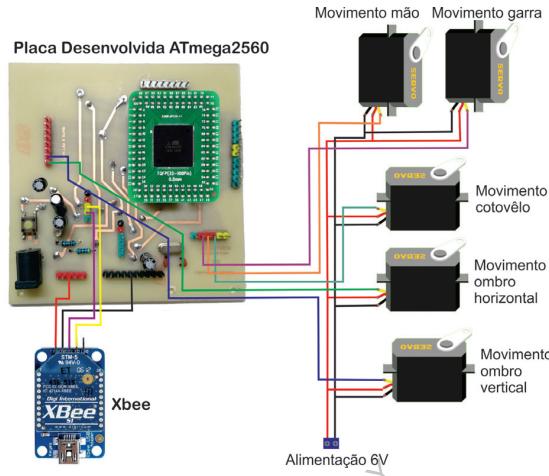
Na placa microcontroladora estão também conectados os componentes relacionados ao braço robótico, estes servomotores são responsáveis por acionar as articulações ou abrir e fechar a garra, conforme o pulso de sinal que recebem. As portas escolhidas para essas conexões foram PG5, PH3, PH4, PH5 e PH6, sendo possível configurar as mesmas para emitirem um sinal do tipo PWM. O circuito utilizado para o braço robótico pode ser verificado na Figura 34.

### 3.3.3 Alimentação dos Dispositivos

O circuito de alimentação exige várias tensões e potências diferentes, uma vez que cada componente utilizado no robô tem tarefas bem distintas.

Foram utilizadas 18 baterias do tipo 18650, sendo associadas a cada 3 em série proporcionando uma tensão de 11,1 V, que foram por sua vez, associadas em paralelo aumentando a capacidade de fornecimento da corrente para 13,6 Ah. A ponte H foi conectada diretamente as baterias disponibilizando uma tensão de 11,1 V para os motores.

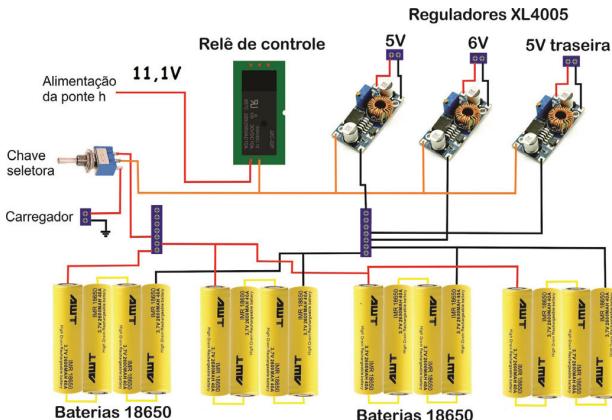
Figura 34 – Circuito utilizado no braço robótico



Fonte: O Autor.

Para regular as tensões conforme a necessidade dos circuitos, foram utilizadas duas placas reguladoras do modelo XL4005, uma ajustada para 5 V que alimenta o Raspberry Pi e a placa microcontroladora desenvolvida, e outra para alimentação dos servomotores com 6 V. O circuito utilizado para alimentar os dispositivos pode ser verificado na Figura 35. Além dos

Figura 35 – Circuito de alimentação dos componentes do robô



Fonte: O Autor.

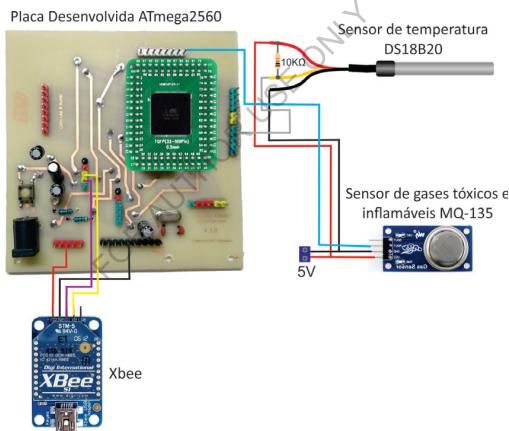
componentes já mencionados foi utilizado uma chave de duas posições e três terminais, esta

permite controlar a alimentação alimentando todos os dispositivos do robô ou deixando o circuito isolado para recarga das baterias. Para recarga das baterias foi fabricado um suporte em alumínio, que abriga o conector de entrada da alimentação fornecida por uma fonte que fornece 12 V para recarga das baterias.

### 3.3.3.1 Sensores

Os sensores de temperatura e de gases foram instalados em locais estratégicos do robô para permitirem melhores obtenções de medidas. O sensor de temperatura DS18B20 foi preso a garra do robô para que a leitura seja efetuada com o material que está em contato com a mesma, o Sensor de gases foi preso ao acrílico na parte frontal do robô, fazendo com que o ar a ser medido esteja sempre na direção em que o robô está posicionado. Os sensores foram ambos

Figura 36 – Circuito dos sensores utilizados pelo robô



Fonte: O Autor.

alimentados com tensão de 5 V, sendo que o sensor de temperatura utiliza uma porta digital para leitura e o sensor de gases (que permite a leitura tanto analógica, quanto digital), foi conectado a uma porta para realizar a leitura analógica, utilizando o conversor AD.

## 3.4 MONTAGEM

### 3.4.1 Montagem da suspensão e motores

Após a pintura das peças foi realizada a perfuração do chassis e foi constatado que a fixação da suspensão dianteira não seria possível, devido ao tamanho dos amortecedores e molas.

Para resolver este problema foi feito um molde de fibra mais extenso para fixação da suspensão, permitindo o funcionamento adequado. A suspensão traseira também apresentou falhas na estrutura da balança que foi reforçada com mais duas camadas de fibra e resina.

A montagem da suspensão dianteira utilizou duas balanças de alumínio, fixadas na parte inferior do chassi, duas peças de fixação para os motores que foram fabricadas, onde os motores se encaixaram. Em seguida foram montados os amortecedores e molas em duas barras roscadas instaladas na parte superior do chassi.

### 3.4.2 Montagem do braço robótico

O braço robótico foi montado com quatro articulações apenas, devido ao movimento lateral do braço, seu peso aumentou consideravelmente o que diminuiu consideravelmente o desempenho. Foi adaptada a garra uma extensão que permite segurar objetos com maior facilidade onde foi instalado o sensor de temperatura.

Figura 37 – Braço robótico



Fonte: O Autor.

A Figura 37 apresenta a montagem do braço robótico, na qual foram utilizados 5 servomotores e dois parafusos para fixação. Os cabos dos motores foram acondicionados em um chicote para mantê-los unidos e mais protegidos.

#### 3.4.2.1 Adição de mais um eixo ao braço robótico

Foi adicionado mais um eixo ao braço robótico permitindo a realização de movimentos laterais que antes eram executados somente pelas rodas do robô, para que o mesmo conseguisse posicionar-se para o objeto a ser coletado.

#### 3.4.3 Montagem do suporte para câmera

A câmera para Raspberry Pi foi fixada no alto do suporte e para abrigar-lhe foi necessário a fabricação de uma caixa de fibra e a adaptação de um servomotor no interior do cilindro. Esse servomotor é o responsável por girar a câmera na vertical e foi acoplado a ele um suporte que sobrou do braço robótico. A montagem da placa de acrílico com o servomotor inferior, o

Figura 38 – Módulo da câmera



Fonte: O Autor.

cilindro com o servomotor superior, e a caixa da câmera montada sobre o suporte é apresentado na Figura 38.

#### 3.4.4 Montagem das peças de acrílico

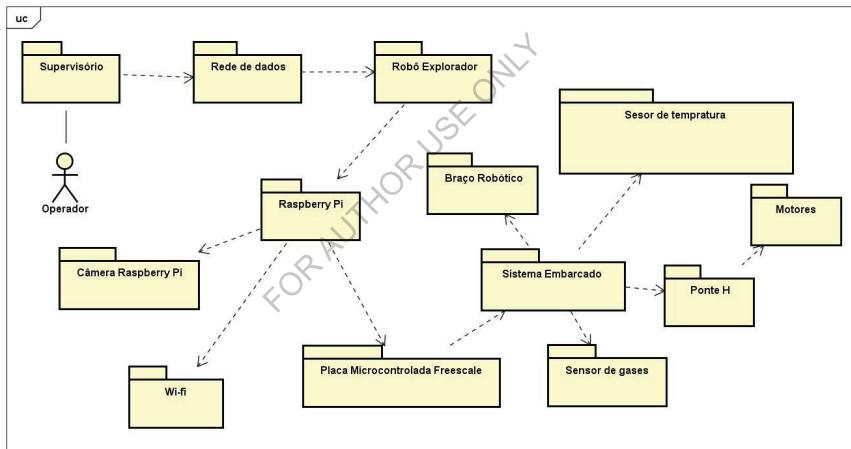
As peças de acrílico foram cortadas e fixadas sobre o chassi, permitindo um nível a mais para fixação de placas, sensores e também o servomotor, responsável por girar a câmera na horizontal. As laterais foram fixadas nas partes inferiores por meio de cantoneiras e parafusos, que acabaram não encaixando no chassi. A solução foi ajustar o chassi para o perfeito encaixe das peças de acrílico, a estas peças foram fixados o Raspberry Pi.

### 3.5 DOCUMENTAÇÃO DE HARDWARE

#### 3.5.1 Diagramas de contexto e requisitos

Os requisitos necessários para criação do *software* foram coletados e com base neles é possível desenvolver o diagrama de contexto de *hardware*, que indica quais são os componentes utilizados e tecnologias envolvidas. O Quadro 2 mostra este diagrama, onde o operador através do supervisório irá controlar o robô, enviando e recebendo dados da rede que está conectada ao Raspberry Pi que estará, também recebendo comandos e transmitindo imagens. O Raspberry Pi estará trocando dados com os microcontroladores para que seja possível executar todas as funcionalidades do robô, como se deslocar e acionar o braço robótico.

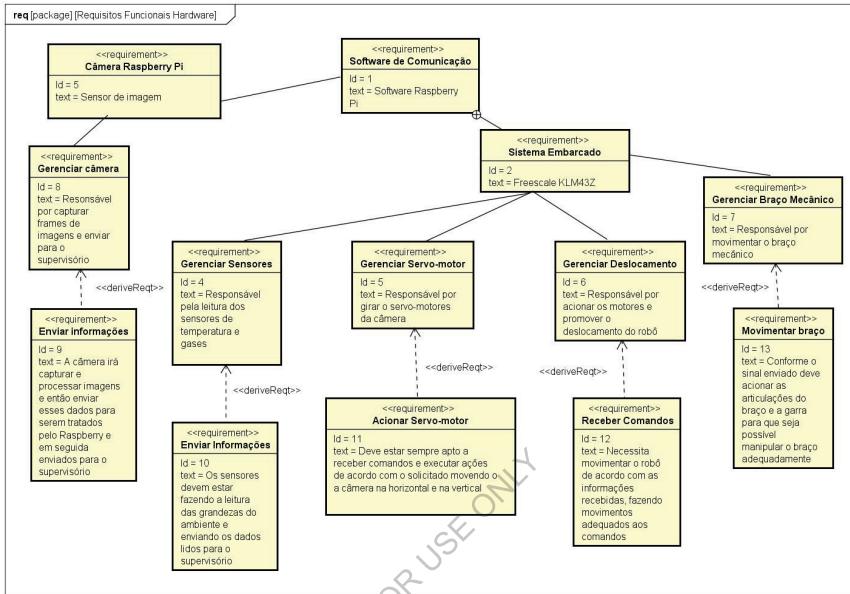
Quadro 2 – Diagrama de Contexto de *Hardware*



Fonte: o Autor.

No diagrama de projeto de *hardware* estão os requisitos funcionais envolvidos na parte física do projeto. Conforme o Quadro 3 o *software* de comunicação irá se conectar ao Raspberry Pi e ao microcontrolador para realizar as funcionalidades exigidas do robô. Em cada um dos quadros é visto o que está sendo gerenciado e na parte inferior o que é desejado como resultado; cada um dos quadros está conectado de acordo com as ligações existentes no *hardware* do robô.

Quadro 3 – Requisitos Funcionais de *Hardware*



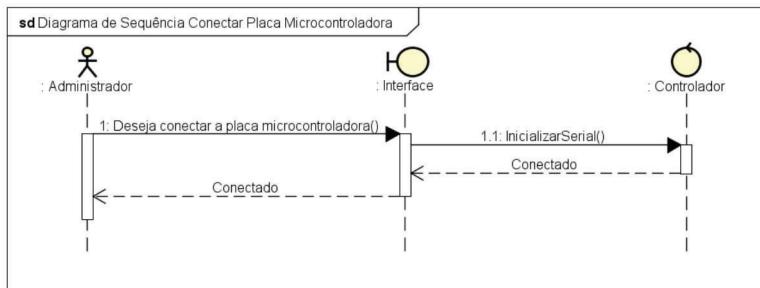
Fonte: o Autor.

## 3.6 DOCUMENTAÇÃO DE SOFTWARE

### 3.6.1 Identificação de Operações e Consultas de Sistema

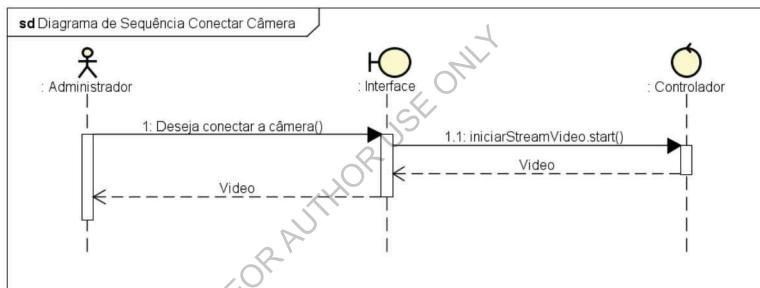
As operações e consultas do sistema foram mapeadas e são apresentadas nos diagramas que envolvem as principais funções do software que são: a de realizar a conexão da placa microcontroladora, da câmera e do controle de Xbox. O Quadro 4, indica a sequência necessária para conexão com a placa microcontroladora, onde ao pressionar o botão “Conectar Serial” é chamada a *Thread* responsável por gerenciar a conexão com a placa microcontroladora. O Quadro 5, indica a sequência necessária para estabelecer a conexão com a câmera do Raspberry Pi, onde ao pressionar o botão “Conectar Câmera” é chamada a *Thread* responsável por gerenciar esta conexão. É apresentado o diagrama com a sequência necessária para estabelecer a conexão com o controle de Xbox no Quadro 6, onde ao pressionar o botão “Verificar Conexão Controle” é exibido no campo imediatamente à direita do botão se o controle está “Conectado” ou “Desconectado”. As consultas e operações do sistema são aquelas que chamam determinados métodos presentes no *software*, esses métodos por sua vez realizam determinadas funcionalidades es-

Quadro 4 – Diagrama de Sequência para conexão com a placa microcontroladora



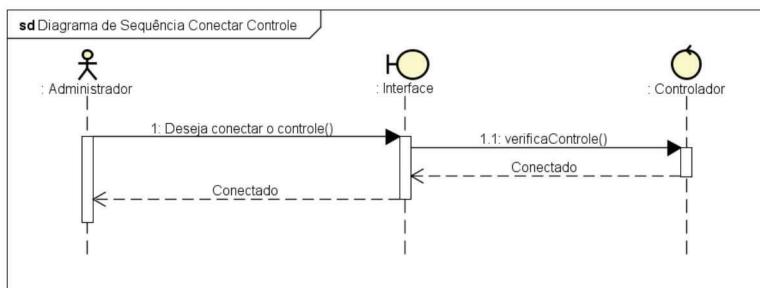
Fonte: o Autor.

Quadro 5 – Diagrama de Sequência para conexão com a câmera



Fonte: o Autor.

Quadro 6 – Diagrama de Sequência para conexão com o controle de Xbox



Fonte: o Autor.

Tabela 1 – Operações e consultas de sistema

Operação do Sistema
updateSerial()
initializeSerial()
closeSerial()
form(object: sender, EventArgs e)
xbox_cClick(object: sender, EventArgs e)
connect_serial_button_Click(object: sender, EventArgs e)
camera_video_button_Click(object: sender, EventArgs e)
stop_video_button_click(object: sender, EventArgs e)

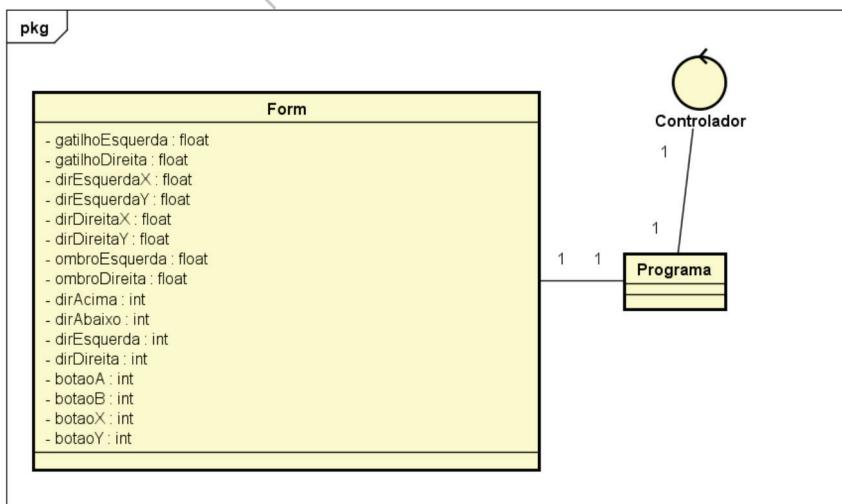
Fonte: O Autor.

peradas como realizar a conexão com algum dispositivo ou acionar um atuador. As consultas são aquelas que devem trazer como resultado uma lista de itens previamente salvos no software ou um banco de dados, este supervisório não utiliza esse conceito. As principais operações de sistema são apresentadas na Tabela 1.

### 3.6.2 Modelo Conceitual

O modelo conceitual traz um esboço mínimo de atributos que o *software* deve conter. O mínimo que se espera são aqueles apresentados no Quadro 7 que ilustra o Modelo Conceitual desenvolvido.

Quadro 7 – Diagrama do Modelo Conceitual

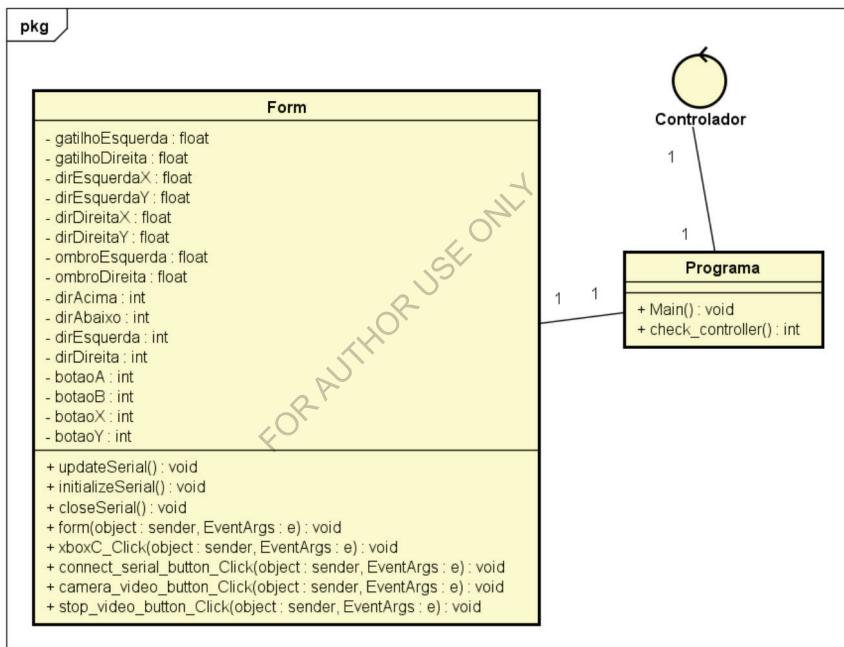


Fonte: o Autor.

### 3.6.3 Diagrama de classes de projeto

O diagrama de classes de projeto, como o próprio nome indica, apresenta quais são as classes desenvolvidas no software e como estas se relacionam umas com as outras. É similar ao projeto de um banco de dados onde as tabelas estão relacionadas umas com as outras e cada uma delas guarda dados importantes que podem estar relacionados com outras tabelas ou não. O Diagrama de classes de projeto é apresentado no Quadro 8.

Quadro 8 – Diagrama de classes de projeto

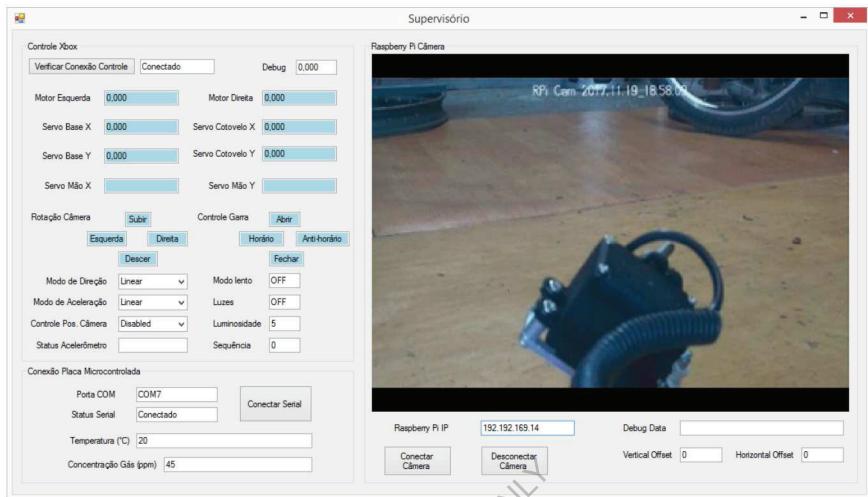


Fonte: o Autor.

### 3.6.4 Projeto gráfico da tela central

A tela central do *software*, é composta por três agrupamentos principais, sendo eles: A conexão via porta serial, conexão com a câmera e teste dos botões do controle de Xbox. O objetivo é que ela mantenha simplicidade, sendo de fácil utilização e apresentando apenas as funcionalidades essenciais para controle do robô além de obter informações sobre o ambiente em que o mesmo se encontra. O projeto gráfico da tela central é apresentado na Figura 39.

Figura 39 – Tela central do Supervisório



Fonte: o Autor.

### 3.6.5 Desenvolvimento do *software* embarcado

Com base na documentação disponibilizada no site e no manual do microcontrolador ATmega2560 Atmel (2017), foram criadas bibliotecas utilizando os registradores responsáveis por cada uma das funcionalidades necessárias para a utilização das portas disponíveis na placa. Essas bibliotecas são incluídas na solução que está sendo desenvolvida, com o objetivo de utilizar suas funções, deixando assim o código mais limpo e organizado. Normalmente são criadas bibliotecas seja para leitura de um sensor, um saída digital ou de modulação de largura de pulso utilizada pelos servomotores.

#### 3.6.5.1 Biblioteca para controle de portas IO

Para utilizar uma das portas do microcontrolador primeiramente verifica-se no diagrama de pinagem os nomes dos pinos, são sempre seguidos do PORT e o número correspondente. É então criada uma função para determinar se este pino será uma entrada ou saída, de acordo com os parâmetros recebidos por cada um dos registradores envolvidos. Os registradores envolvidos para a utilização das portas são: PORTx utilizado para alterar os valores das saídas, DDRx responsável por indicar se a porta será saída ou entrada, de acordo com o valor recebido 0 ou 1, respectivamente, PINx registrador de entrada utilizado para leitura de valores nos pinos. O

x em cada um dos registradores deve ser alterado pelo PORT a ser configurado e um número hexadecimal ou binário de 8 bits é passado como parâmetro.

Todos os registradores mencionados acima devem ser configurados para que a porta escolhida seja uma saída ou entrada digital, a ser utilizada no microcontrolador. Nesta biblioteca foram criadas duas funções, uma para entrada e outra para saída, com a assinatura em que recebe uma variável do tipo *char* e outra do tipo *int* para setar os registradores da forma adequada, indicando qual o PORT a ser configurado e o pino correspondente.

### 3.6.5.2 Biblioteca para leitura de portas analógicas

Para utilizar o conversor analógico/digital presente no microcontrolador ATmega2560 são necessárias a conclusão de três etapas: determinar a quantidade de amostras recebidas de acordo com os pulsos de *clock*, o *buffer* de leitura é então desconectado dividindo as amostras e por último os valores são convertidos para um valor digital. Sendo a saída do conversor analógico/digital de 10 bits os registradores envolvidos são: REFS1:0 para selecionar a fonte de tensão utilizada para leitura, ADLAR para configuração do bit mais significativo a direita ou a esquerda, MUX3:0 para seleção do canal analógico utilizado para leitura, ADEN habilita o conversor analógico-digital, ADSC inicia a conversão, ADPS2:0 determina a divisão do *clock* do processador para o *clock* do ADC.

Foram criadas três funções nesta biblioteca, sendo uma que não recebe parâmetros e inicializa a configuração do conversor, outra onde é indicado qual o PORT e pino a serem utilizados como entrada analógica e uma que realiza a leitura do sinal e controla os registradores envolvidos, em sua assinatura é retornado um valor do tipo *float*.

### 3.6.5.3 Biblioteca para controle dos servomotores

Para controle dos servomotores foi desenvolvida uma biblioteca capaz de variar o valor dos pinos como saída digital, gerando uma onda digital modulada por largura de pulso conhecida como *pwm*. Como os servos utilizados trabalham com um pulso de *clock* do tamanho de 20 ms, além de setar os registradores foi trabalhado com esse período para determinar as alterações na onda que permitissem uma precisão de 1°.

Os registradores envolvidos na criação dessa biblioteca são: TCNT0 que controla a contagem do *timer* recebe vários parâmetros através de 8 bits de configuração, o registrador TIRF0 determina as interrupções da *flag*, além dos registradores OCR0A e OCR0B que são utilizados para comparação das saídas.

Foram criadas três funções, sendo uma responsável por configurar as portas como saída, uma para configurar os valores desejados para o PWM e outra para variar os valores em graus, conforme o parâmetro passado na assinatura do método.

### 3.6.5.4 Biblioteca para comunicação USART

A última biblioteca desenvolvida é utilizada para receber e enviar dados para a placa microcontroladora, onde podem ser lidos valores de sensores, ou enviados caracteres de comando para executar uma ação já programada no *firmware*.

Para a configuração da USART os registradores envolvidos são: UBRR0 que determina a velocidade da transmissão dos dados, UCRS0C que determina em qual borda é usada a amostragem, o tamanho do dado transmitido e outros bits relativos a paridade e bits de parada são configurados nos registradores UCSR0B e UCSR0C respectivamente.

Foram criadas três funções, uma delas responsável pela configuração de todos os parâmetros envolvidos para envio e recebimento dos dados via USART0, usada pela porta serial do microcontrolador e outras duas responsáveis por enviar e receber dados, sendo a que a responsável por transmitir os dados em sua assinatura dois valores *float*, referentes a cada um dos sensores lidos. A função que recebe os dados pela porta serial, recebe uma *string*, que é dividida em vários parâmetros.

## 3.6.6 Desenvolvimento de Software Supervisório

O software supervisório foi desenvolvido em sua totalidade utilizando a linguagem C# e algumas bibliotecas escritas para ela, como a XInputDotNetPure utilizada para lidar com os eventos a serem capturados pelo controle Xbox ou a IO utilizada para realizar as conexões envolvendo a porta serial do computador e a rede wi-fi. O aplicativo contém apenas duas classes sendo uma delas a principal e outra responsável por monitorar os eventos enviados pelo controle Xbox, também foram criadas duas *Threads*, sendo estas responsáveis por atualizar a serial e o controle Xbox durante a utilização do *software*.

### 3.6.6.1 Conexão com Raspberry Pi via IP

A conexão de dados com o Raspberry Pi é realizada pela Wi-fi e utiliza o protocolo HTTP para se conectar com a câmera localizada no robô. Através de um servidor apache rodando no Raspberri Pi as imagens da câmera são transmitidas pela rede e podem ser capturadas por um dispositivo que se conecte a este servidor. O software carrega o endereço do vídeo em

uma *stream* (uma transmissão do vídeo), com um *buffer* (memória utilizada para carregamento) de tamanho definido, em seguida os dados que estavam no *buffer* são carregados em uma imagem que é exibida no supervisório. O Apêndice B apresenta um trecho do código utilizado para inicialização da conexão com a câmera presente no Raspberry Pi.

### 3.6.6.2 Conexão com a placa microcontroladora via USART

A conexão serial com a placa microcontroladora utiliza funções da biblioteca IO, primeiramente é verificada se existe a porta serial aberta, são então lidos e escritos valores na serial em forma de *strings* que são tratadas pelo supervisório e pelo *firmware* da placa. De acordo com as *strings* é que são enviados comandos ou lidos valores, de acordo com um formato específico separado por vírgulas.

A inicialização da *thread* responsável pelo processamento da serial é Apresentada no apêndice C, onde também é apresentado o controle da conexão serial e uma verificação de recebimento de dados.

### 3.6.6.3 Detecção de eventos do controle Xbox

Para detectar os eventos do controle Xbox foram criados métodos responsáveis por verificar quando os botões são pressionados no controle e quando são liberados. Foram desenvolvidos também modificadores para os direcionais analógicos e gatilhos que permitem alternar entre os modos linear, quadrado e cúbico permitindo que sejam realizados movimentos com menor velocidade e maior precisão, durante a utilização do controle no supervisório. Basicamente é instanciado um objeto do tipo *GamePadState* e nos métodos são verificadas as posições dos controles analógicos, valores de gatilhos e se os demais botões foram pressionados. Existem diversos exemplos com as mais variadas bibliotecas de como implementar estes métodos.

Devido a falta de precisão apresentada pelo teclado, a escolha de um *joystick* se mostrou necessária, através dele é possível variar não somente a intensidade com que as teclas são pressionadas, mas também a precisão apresentada pelas mesmas. Muitos robôs controlados remotamente contam com a opção de controle por um *joystick*. Um trecho do código utilizado onde é inicializada a *thread* utilizada para processamento dos eventos do controle de Xbox, bem como a instanciação de um objeto do tipo *Gamepad* e a detecção de eventos de alguns botões são apresentados no Apêndice D.

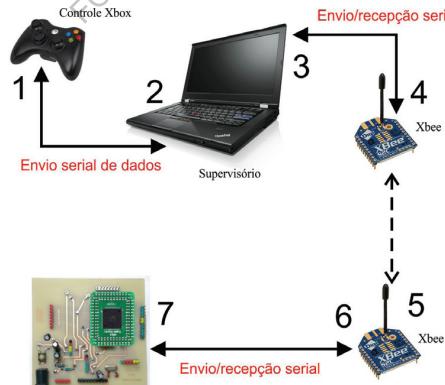
### 3.6.7 Comunicação entre controle, supervisório e robô

Para o completo funcionamento do sistema existe uma integração entre o controle, o supervisório e o robô. Os parâmetros escolhidos para conexão serial entre os dispositivos foi de 57600 bps, com 8 bits de envio de dados, 1 *stop* bit e nenhum bit de paridade. Estas configurações foram utilizadas tanto do lado do supervisório, como no microcontrolador e da mesma forma nos módulos XBee.

O controle é conectado via USB com o supervisório (em 1), e os dados são enviados de forma serial por uma determinada porta permitindo que o computador seja capaz de interpretar cada um dos valores enviados pelo controle (em 2). Em seguida o supervisório realiza a leitura de cada um desses valores utilizando um componente para conexão via serial (em 3). Os dados são então repassados por meio de outra serial para o XBee transmissor (em 4), que por sua vez, replica o sinal ao XBee receptor (em 5). Os dados são recebidos pelo outro módulo XBee (em 6), que envia os dados pela porta serial 1 do microcontrolador (em 7) e então processados, para realizar determinadas tarefas. A comunicação no sentido inverso ocorre da mesma forma para a leitura dos sensores, com exceção dos itens (1 e 2, pois os dados não retornam até o controle).

A comunicação via seriais é apresentada na Figura 40.

Figura 40 – Comunicação: Controle, supervisório e microcontrolador



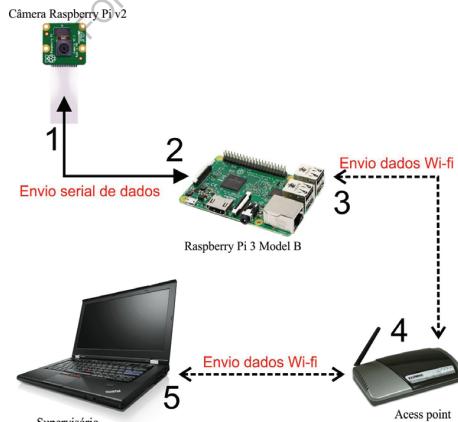
Fonte: o Autor.

Os pacotes de dados enviados do supervisório para o microcontrolador são montados seguindo o seguinte formato: 'L,R,B,VAL,PAM,TIL,Y', onde L e R são os valores para acio-

namento dos motores, B a articulação do braço robótico que está em movimento, VAL o valor do ângulo em que o servo escolhido deve se movimentar, PAM o ângulo do servo horizontal da câmera e TIL o ângulo do servo responsável pelo movimento vertical da câmera, o caractere Y é utilizado apenas para controle. Já os pacotes enviados do microcontrolador para o supervisório tem o formato: 'A,TEM,GAS,X', onde o supervisório realiza a verificação da serial procurando pelo caractere A, quando este é encontrado ele faz a leitura de valores até encontrar o X, que é o caractere utilizado como controle para encerrar a leitura de dados. Nesta mesma etapa os campos TEM e GAS são atribuídos a temperatura e a concentração de gás, respectivamente, exibidas na tela do supervisório.

As imagens transmitidas pelo Raspberry Pi utilizam a seguinte via para serem lidas pelo supervisório: as imagens obtidas pela câmera durante a leitura são enviadas via serial para o Raspberry Pi através de um cabo *flat* (em 1), estas imagens são capturadas por um pequeno software (em 2), responsável por disponibilizá-las através de um endereço HTTP definido em um servidor Apache (em 3), onde o supervisório realiza uma conexão, por meio de um *access point* (em 4), para capturar esses dados e exibi-los em um componente *picture box* (em 4). A transmissão dos dados entre o Raspberry Pi e o supervisório são apresentadas na Figura 41.

Figura 41 – Comunicação: Raspberry Pi e supervisório



Fonte: o Autor.

### 3.7 TESTES

Após a montagem do *hardware* iniciou-se os testes para verificar as capacidades do robô e solucionar os problemas que foram encontrados. As principais funcionalidades do robô foram testadas, entre elas estão: locomoção, leitura de sensores, obtenção de imagem da câmera, movimentação do braço robótico e autonomia da bateria.

#### 3.7.1 Locomoção

Para locomoção do robô foi desenvolvida uma biblioteca para ser usada no *software* embarcado capaz de movimentar o robô. Os principais movimentos incluem avançar, retroceder e virar para esquerda e direita, com a variação apresentada pela sensitividade dos botões do controle de Xbox 360 e da placa para controle de PWM melhorou exponencialmente a precisão ao locomover o robô.

Uma rampa foi montada para verificar quais as inclinações máximas suportadas pelo robô ao locomover-se, devido ao diâmetro das rodas o ângulo de entrada em obstáculos mostrou-se pequeno, permitindo subidas de até a  $15^{\circ}$ . A Figura 42, demonstra a rampa montada para testes.

Figura 42 – Rampa para teste de locomoção



Fonte: O Autor.

As inclinações laterais apresentam valores próximos aos  $40^{\circ}$ , sem que a estabilidade do robô seja comprometida, já o angulo de saída apresentou o mesmo problema da entrada, causando o travamento do robô com valores superiores a  $15^{\circ}$ . Levando em consideração que a

utilização do robô desenvolvido é para locomoção em pisos planos, os valor apresentaram-se aceitáveis. Para determinar a velocidade máxima do robô foi estendida uma trena e cronometrado o tempo necessário para percorrer 5 metros foi cerca de 13,55 segundos. Por meio desse valor foi possível determinar a velocidade média escalar de 1,33 Km/h por meio da Equação 3.1.

$$Vm = \frac{\Delta S}{\Delta t} \quad (3.1)$$

### 3.7.2 Leitura de sensores

Duas bibliotecas de *software* foram desenvolvidas para permitir a leitura dos sensores pela placa desenvolvida, uma delas faz a leitura digital (do sensor de temperatura) e outra realiza a leitura analógica (do sensor de gases).

A leitura do sensor de temperatura DS18B20, foi simulada por meio do aquecimento da garra do robô com isqueiro, este sensor tem uma particularidade que envolve a utilização de um resistor de *pullup* para a identificação correta dos valores.

Para a simulação do sensor de fumaça foi utilizado um isqueiro, sendo que o sensor de gases MQ-135 é capaz de realizar a leitura de fumaça e gases tóxicos, entre eles o gás butano, que é utilizado como fluído para isqueiros. De acordo com Machado (2014), o valor considerado como perigoso foi obtido por meio dos valores considerados perigosos por órgãos relacionados a saúde e segurança do trabalho , alguns gases em concentrações maiores que 100 ppm já oferecem risco a saúde e à vida. A Figura 43, apresenta os valores lidos na tela do supervisório

Figura 43 – Leitura de sensores no Supervisório



Fonte: O Autor.

com os testes realizados para a leitura dos sensores. Após a realização dos testes, os sensores foram instalados no robô, o sensor de fumaça foi preso sob o acrílico na parte frontal do mesmo, deixando o sensor exposto e a placa protegida. O sensor de temperatura foi fixado na garra do braço robótico, para que consiga captar as mudanças de temperatura do material que está sendo manipulado da melhor maneira possível.

### 3.7.3 Braço robótico

O braço robótico apresentou desempenho conforme o esperado, para adicionar maior torque o servo 996R, utilizado na base do braço, foi substituído por outro com o dobro da capacidade de torque, o modelo escolhido foi JX PDI-6221MG. As demais articulações se mostraram fortes o suficiente para levantar o braço, entretanto neste modelo de braço robótico a disposição dos servomotores que são distribuídos ao longo do braço, fazem com que parte da força que poderia ser utilizada para levantar a carga seja perdida levantando o peso dos próprios motores. A Figura 44 demonstra a montagem final do braço robótico agora com 4 eixos.

Figura 44 – Braço robótico com 4 eixos



Fonte: O Autor.

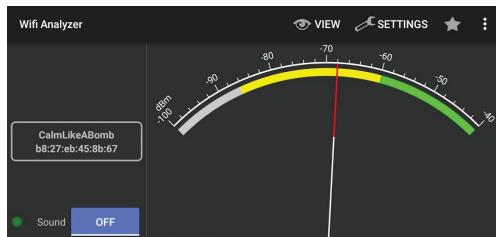
O braço robótico se mostrou capaz de levantar um objeto com peso de até 150 gramas e dimensões máximas de 5 cm de largura, 5 cm de altura e 25 cm de comprimento.

### 3.7.4 Alcance do Sinal

A potência do sinal Wi-Fi emitido pelo Raspberry Pi operando em modo de roteador é similar a de um aparelho roteador com antena interna, com potências que variam de -78 dBm a -82 dBm a uma distância de 20 metros. O teste foi realizado com o sinal atravessando uma parede de alvenaria, a uma distância de 20 metros a potência do sinal era suficiente para transmissão e recepção dos dados com uma velocidade de 11 Mbps, em uma faixa que compreende -76 dBm a -86 dBm, considerada como aceitável (porém um pouco lenta). A Figura 45 demonstra um exemplo de medição do sinal alcançando -68 dBm, utilizando o *software* Wifi Analyzer, aplicativo Android que mede potência do sinal de Wi-Fi.

Para que a conexão de dados fosse melhorada foram utilizados 2 módulos XBee, que permitiram controlar o robô a uma distância muito maior (alcançando aproximadamente 50

Figura 45 – Teste da potência do sinal Wi-Fi



Fonte: O Autor.

metros), do que utilizando somente a Wi-fi. Porém as imagens da câmera ficaram indisponíveis a uma distância superior a 25 metros.

### 3.7.5 Imagens obtidas da câmera

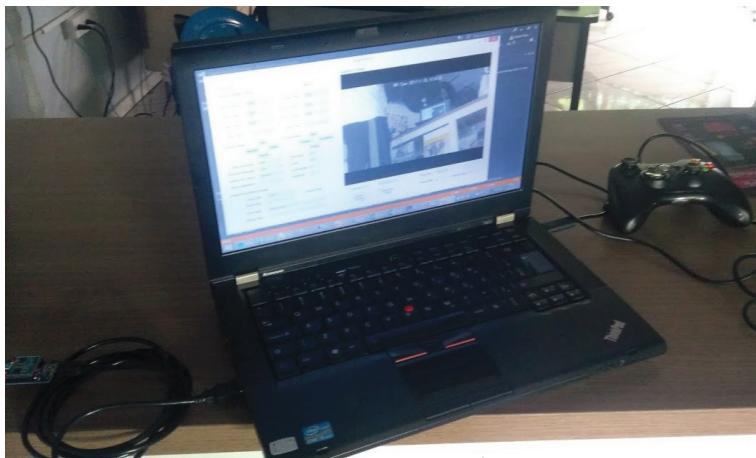
As imagens obtidas da câmera no supervisório apresentaram um pequeno atraso devido as oscilações apresentadas pela Wi-fi do Raspberry Pi, devido ao dispositivo estar em constante movimento ao se afastar o sinal começa a apresentar muita lentidão, diminuindo consideravelmente a velocidade e aumentando perda de pacotes. Uma forma de melhorar este desempenho seria instalar um roteador no robô e outro no supervisório garantindo assim melhor qualidade e potência no sinal.

De um modo geral, as imagens ficaram com boa qualidade e permitem que seja visualizado detalhes que estão a frente e aos lados do robô. É possível girar a câmera cerca de 180° e verificar se existem obstáculos aos lados e atrás do robô o que auxilia muito quando não se tem uma visada direta para o equipamento. A Figura 46 apresenta uma das imagens obtidas no supervisório durante os testes realizados.

### 3.7.6 Autonomia da Bateria

Para o teste de autonomia da bateria foi colocado o robô para se locomover com o PWM dos motores no máximo e cronometrado quanto tempo ele conseguia manter os circuitos funcionais nestas condições. A bateria durou aproximadamente 23 min, sabendo que a velocidade do robô é de 1,33 Km/h ele seria capaz de percorrer uma distância máxima de 522 metros antes da bateria acabar. Esse desempenho é compatível com a função do robô, levando em consideração que a distância máxima de entre o robô e o supervisório não pode ultrapassar 50 metros sem que o controle realizado pela conexão entre os módulos Xbee seja perdido.

Figura 46 – Teste da câmera do Raspberry Pi e do supervisório



Fonte: O Autor.

### 3.7.7 Simulação de ambiente com materiais explosivos, inflamáveis e fumaça tóxica

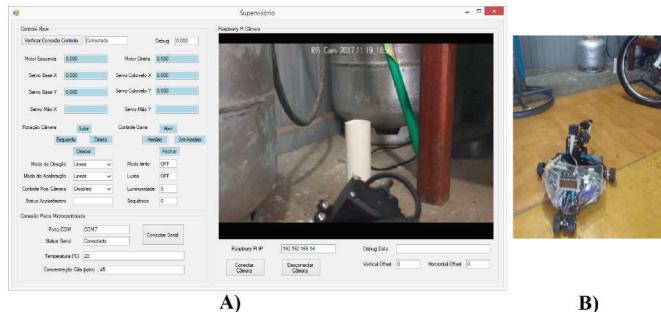
Após serem realizados alguns ajustes no posicionamento de dispositivos envolvidos, calibrações de servo motores com ângulos máximos e mínimos, além do posicionamento da câmera foi realizada uma simulação real onde o robô acessou um local onde existia um dispositivo inflamável coletado que foi posteriormente acionado para emitir calor e fumaça, que permitiram o teste completo das funcionalidades do robô.

A Figura 47 apresenta a imagem obtida no supervisório (em A) identificando o objeto a ser coletado próximo a botijões de Gás Liquefeito de Petróleo de 13 Kgs e a o robô se aproximando do material (em B).

A Figura 48 apresenta o momento em que o objeto coletado começa a emitir calor aumentando a temperatura medida pelo sensor e começa a liberação de fumaça, também identificada pelo sensor de gases (em A), o objeto coletado sendo transportado pelo robô para um ambiente ventilado (em B).

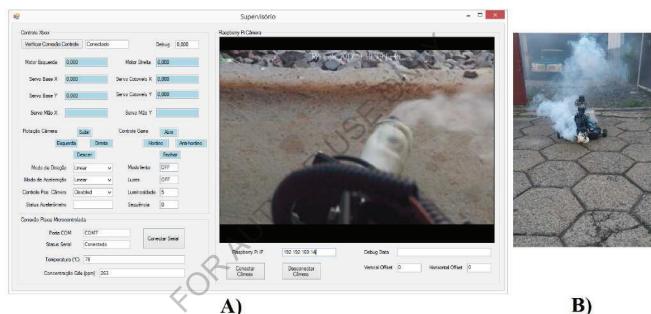
O objeto é deixado no ambiente ventilado, conforme mostra a Figura 49 e começa a ser totalmente consumido pelas chamas (em A), o operador realiza uma manobra evasiva do local para evitar os danos ao robô que possam ser causados pelas chamas (em B).

Figura 47 – Identificação de objeto inflamável a ser coletado



Fonte: O Autor.

Figura 48 – Coleta do objeto inflamável e deslocamento



Fonte: O Autor.

Figura 49 – Disposição de material perigoso em ambiente ventilado



Fonte: O Autor.

### 3.8 RECURSOS E ORÇAMENTOS

#### 3.8.1 Recursos de *hardware*

Conforme mencionado anteriormente, este projeto foi desenvolvido com diversos componentes de *hardware*. Entre eles estão: a placa Raspberry Pi, uma placa desenvolvida com microcontrolador ATMega2560, além de uma câmera. Além dessas placas e dispositivos maiores foram adquiridos componentes de menor capacidade, todavia não menos importantes, como: sensores de temperatura e gases, baterias, motores de corrente contínua e servomotores, entre outros.

Estes recursos estão incluídos no *hardware*, porém, como não serão adquiridos componentes prontos para uso e sim construídos ou adaptados, é importante citá-los separadamente, quais sejam os materiais para construção do chassis, suspensão, estrutura do braço mecânico e chapas de acrílico.

Todos os componentes foram adquiridos pelo acadêmico, desta forma o robô coletor será propriedade do mesmo. Na Tabela 2 são identificados os principais componentes que serão utilizados, bem como o valor aproximado de cada um deles (primeiro semestre do ano de 2017).

Tabela 2 – Orçamento dos componentes utilizados

Custos			
Descrição	Quantidade	Valor Un. Médio	Total
Acrílico (chapa de 4mm)	1	R\$ 60,00	R\$ 60,00
Microcontrolador ATMega2560	1	R\$ 40,00	R\$ 40,00
Placas de fenolite	3	R\$ 60,00	R\$ 180,00
Capacitores, resistores e outros componentes	1	R\$ 10,00	R\$ 10,00
Câmera para Raspberry Pi V2	1	R\$ 150,00	R\$ 150,00
Bandejas de suspensão	2	R\$ 20,00	R\$ 40,00
Baterias	16	R\$ 5,00	R\$ 80,00
Braço robótico	1	R\$ 160,00	R\$ 160,00
Chassi (palitos, cola, resina, fibra)	1	R\$ 150,00	R\$ 150,00
Molas e amortecedores	4	R\$ 35,00	R\$ 140,00
Motores e caixa de redução	2	R\$ 80,00	R\$ 160,00
PVC termo retrátil	1	R\$ 40,00	R\$ 40,00
Raspberry Pi 3 modelo B+	1	R\$ 300,00	R\$ 300,00
Rodas	4	R\$ 20,00	R\$ 80,00
Sensor de gases	1	R\$ 14,00	R\$ 14,00
Sensor de temperatura	1	R\$ 10,00	R\$ 10,00
Servo motores JX PDI-6221MG	1	R\$ 150,00	R\$ 150,00
Servo motores MG 995	8	R\$ 30,00	R\$ 240,00
Módulos Xbee S2	2	R\$ 120,00	R\$ 240,00
TOTAL			R\$ 2.224,00

Fonte: O Autor

#### 4 RESULTADOS

Observa-se que o robô consegue operar adequadamente de acordo com o que foi projetado, tanto em *software* como em *hardware*. Com a inclusão de mais um eixo no braço robótico e uma ponte H transistorizada, foi possível obter maior precisão na coleta de materiais, melhorando assim o controle geral do robô. Nos testes realizados em uma sala comercial com aberturas grandes o comportamento do robô foi conforme o esperado, conseguindo manter a comunicação com o supervisório a uma distância máxima de 50 metros, permitindo o controle via XBee e detectando dados do ambiente como: temperatura e a presença de gases tóxicos ou inflamáveis. O envio de imagens se mostrou estável até uma distância de 25 metros, apresentando baixa perda de quadros e atraso de poucos milissegundos durante as transmissões. A autonomia do robô foi de 23 minutos de operação, utilizando os motores para locomoção em velocidade máxima de 1,33 Km/h, ele consegue percorrer uma distância de até 500 metros, sem obstáculos.

## 5 CONSIDERAÇÕES FINAIS

A placa microcontroladora desenvolvida mostrou-se funcional, permitindo a gravação do *software* embarcado, permitindo habilitar todas as portas selecionadas para uso, porém as suas capacidades foram limitadas ao desenvolvimento do robô.

O *software* embarcado desenvolvido para comunicação via serial, permitiu o controle do robô de forma satisfatória, além da leitura de sensores e a utilização de recursos como PWM.

O *software* supervisório desenvolvido para controle, se mostrou adequado a aplicação, permitindo gerenciar tanto eventos da porta serial bem como do controle, além de exibir imagens da câmera localizada no Raspberry Pi.

Foram necessários a alteração da ponte H, banco de baterias e a adição de mais um eixo ao braço robótico. Esses ajustes permitiram o controle da locomoção do robô com velocidade variável, e maior amplitude do braço robótico, facilitando a coleta de objetos.

O robô foi testado em uma sala comercial com piso plano e com aberturas grandes e teve desempenho dentro do esperado, locomovendo-se para realizar leituras dos sensores de gases e temperatura no ambiente. Em 66% das tentativas conseguiu coletar e transportar o objeto.

Conclui-se que é possível desenvolver um robô explorador de pequeno porte e baixo custo. O tempo de autonomia das baterias foi de 23 minutos, carregando uma carga máxima com o braço robótico de até 150 gramas. Os valores obtidos pelos sensores foram transmitidos a uma distância máxima de 50 metros com visada e as imagens da câmera foram transmitidas a uma distância máxima de 25 metros utilizando *wi-fi*.

## REFERÊNCIAS

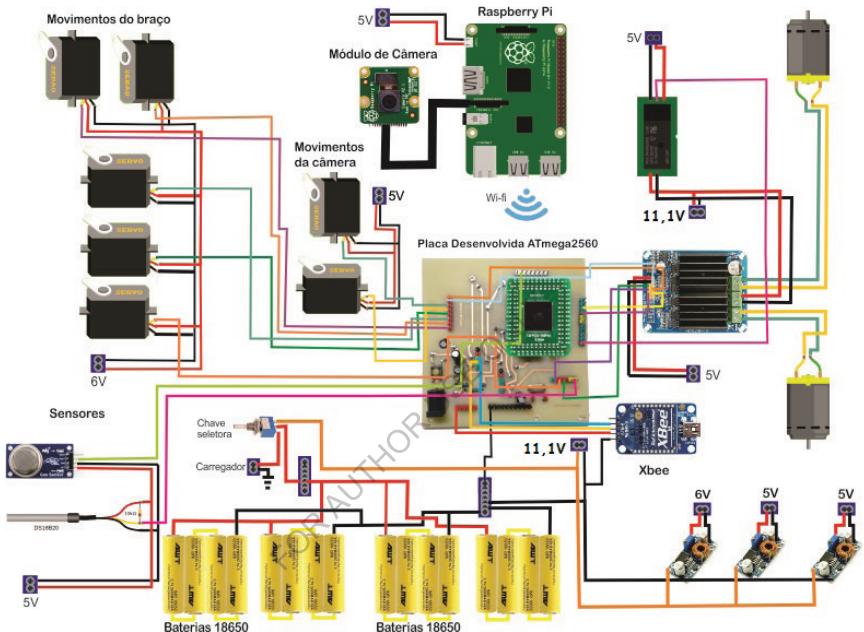
- ADAMS, Herb. **Chassis Engineering: Chassis Design, Building and Tuning for High Performance Handling.** [S.I.]: Federation of European Explosives and Manufactures, 1992. 144 p.
- ATMEL. **ATmega2560.** [S.I.], 2017. Disponível em: <<http://www.atmel.com/pt/br/devices/ATMEGA2560.aspx>>. Acesso em: 06.09.2017.
- BENSON, Coleman. **History of Robots: Timeline.** [S.I.], 2008. Disponível em: <[www.robotshop.com/media/files/pdf/timeline.pdf](http://www.robotshop.com/media/files/pdf/timeline.pdf)>. Acesso em: 19.09.2017.
- BRAUNL, Thomas. **Embedded Robotics.** [S.I.]: School of Electrical, Electronic and Computer Engineering, 2003. 458 p.
- CAPUTO, Victor. **Conheça o robô que cuidará da segurança na Copa do Mundo.** 2014. Revista Exame. Disponível em: <<http://exame.abril.com.br/tecnologia/conheca-o-robo-que-cuidara-da-seguranca-na-copa-do-mundo/>>. Acesso em: 18.05.2017.
- DEITEL, Harvey M.; DEITEL, Paul J. **C++ Como Programar.** [S.I.]: Pearson, 2006. 1208 p.
- \_\_\_\_\_. **C Como Programar.** [S.I.]: Pearson, 2011. 846 p.
- DIGI. **XBee and XBee-PRO RF Modules.** [S.I.], 2014. Disponível em: <<https://www.sparkfun.com/datasheets/Wireless/Zigbee/XBee-Datasheet.pdf>>. Acesso em: 09.10.2017.
- FOUNDATION, Detroit Public Safety. **Bomb Squad.** 2015. Bomb Squad. Disponível em: <<http://www.detroitpublicsafetyfoundation.org/police-dpd/bomb-squad/>>. Acesso em: 05.11.2017.
- INOVATION, The Tech Museum of. **What is Robot?** 2005. The Tech Museum of Inovation. Disponível em: <<https://www.thetech.org/exhibits/online/robotics/activities/page02.html>>. Acesso em: 13.08.2017.
- INTEGRATED, Maxim. **DS18B20.** [S.I.], 2017. Disponível em: <<https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>>. Acesso em: 06.10.2017.
- MACHADO, Claudinei. **20 Gases mais tóxicos.** [S.I.], 2014. Disponível em: <<http://www.protecaorespiratoria.com/20-gases-mais-toxicos/>>. Acesso em: 08.10.2017.
- MANSOOR, Shamyl Bin et al. **Wireless Bomb Disposal Robot.** 2001. Sir Syed University of Engineering and Technology. Disponível em: <[http://smansoor.seecs.nust.edu.pk/downloads/WBDR\\_Report.pdf](http://smansoor.seecs.nust.edu.pk/downloads/WBDR_Report.pdf)>. Acesso em: 07.11.2017.
- MARTINS, Gilberto de Andrade. **Manual para elaboração de monografias e dissertações.** [S.I.]: Atlas, 2000. 298 p.
- MAZIDI, Muhammad Ali. **Freescale ARM Cortex-M Embedded Programming: Using C Language.** [S.I.]: Mazidi and Naimi, 2014. 682 p.
- MICROSOFT. **Xbox 360 controle para Windows.** [S.I.], 2017. Disponível em: <<https://www.microsoft.com/accessories/pt-br/d/xbox-360-controller-for-windows>>. Acesso em: 10.10.2017.

- MIYADAIRA, Alberto Noboru. **Microcontroladores PIC18. Aprenda e Programe em Linguagem C.** [S.I.]: Editora Érica, 2013. 400 p.
- MOLLOY, Derek. **Exploring BeagleBone: Tools and Techniques for Building with Embedded Linux.** [S.I.]: Morgan and Claypool Publishers, 2011. 600 p.
- MURPHY, Robin. **Introduction to AI Robotics.** [S.I.]: Massachusetts Institute of Technology, 2000. 466 p.
- NASA. **Mars Exploration Rovers.** 2015. Nasa. Disponível em: <<http://mars.nasa.gov/mer/home/index.html>>. Acesso em: 18.08.2017.
- PEREIRA, Fábio. **Tecnologia ARM Microcontroladores de 32 bits.** [S.I.]: Editora Érica, 2007. 314 p.
- PI, Raspberry. **Raspberry Pi Model b+.** 2005. Raspberry Pi Fundation. Disponível em: <<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>>. Acesso em: 11.08.2017.
- RENO, Janet. **A Guide For Explosion And Bombing Scene Investigation.** [S.I.]: U.S. Department Of Justice, 2000. 52 p.
- RYDER, Chris. **A Special Kind of Courage.** [S.I.]: Methuen, 2005. 352 p.
- SANDIN, Paul E. **Robots Mechanisms And Mechanical Devices.** [S.I.]: McGraw-Hill Companies, 2003. 299 p.
- SIEGWART, Roland; NOURBAKHS, Illah. **Introduction to Autonomus Mobile Robots.** [S.I.]: Massachusetts Institute of Technology, 2004. 321 p.
- SKILLANO, Bruno; KHATIB, Oussama. **Springer Tracts in Advanced Robotics.** [S.I.]: Star, 2015. 510 p.
- WASLAWICK, Raul Sidnei. **Análise e Projeto de Sistemas de Informação Orientados a Objetos.** [S.I.]: Elsevier, 2011. 329 p.
- WISE, Edwin. **Robotics Demystified.** [S.I.]: McGraw-Hill Professional, 2005. 314 p.

## APÊNDICE A – CIRCUITO COMPLETO DO ROBÔ COLETOR DESENVOLVIDO

Apresentação de todo circuito utilizado para desenvolvimento do robô coletor. Figura 50.

Figura 50 – Circuito completo do robô



Fonte: O Autor.

## APÊNDICE B – INICIALIZAÇÃO DA CONEXÃO COM A CÂMERA VIA IP

```
private void cameraVvideoButtonClick(object sender, EventArgs e) {  
    Thread getVideoStream = new Thread(new ThreadStart(receiveVideoStream));  
    getVideoStream.IsBackground = true;  
    getVideoStream.Start(); }  
  
private void stopVideoButtonClick(object sender, EventArgs e)  
{ videorun = 0;  
    cameraImage.Image = Image.FromFile("defaultCamera.png"); }  
int videorun = 1;  
int upcommand = 0;  
int downcommand = 2;  
int leftcommand = 6;  
int rightcommand = 4;  
int horizOffset = 0;  
int vertOffset = 0;  
  
private delegate void imageDelegate(Bitmap image);  
private delegate void debugTextDelegate(string debugData);  
private delegate void vertOffsetDelegate(string vert);  
private delegate void horizOffsetDelegate(string horiz);  
  
private void displayImage(Bitmap image)  
{  
    cameraImage.Image = image; }  
  
private void displayDebug(string debugData)  
{ debugTextbox.Text = debugData; }  
  
private void displayVert(string vert)  
{ vertOffsetValue.Text = vert; }  
  
private void displayHoriz(string horiz)  
{ horizOffsetValue.Text = horiz; }
```

## APÊNDICE C – INICIALIZAÇÃO DA THREAD E LEITURA DE DADOS PARA CONEXÃO VIA SERIAL

```
Thread updateIO = new Thread (new ThreadStart(UpdateSerial));
updateIO.IsBackground = true;
updateIO.Start();
private void UpdateSerial() {
    while (true) {
        if (portaSerial.isOpen == true) {
            if (received.Contains("A") && received.Contains("Z")) {
                if (received.Contains("R")) {
                    rumblecount = 3;
                }
                serialInputTextbox.Invoke(new inputDdelegate(displayIinput), received);
            }
        }
        else
        {
            serialOutputTextbox.Invoke(new outputDelegate(displayOoutput), "Sem conexão");
        }
        if (portaSerial.isOpen == false)
        {
            Thread.Sleep(20);
        }
    }
}
```

## APÊNDICE D – ASSINATURA DE MÉTODO UTILIZADO PARA DETECÇÃO DE EVENTOS DO CONTROLE XBOX

```
Thread updatecontroller = new Thread(new ThreadStart(UpdateState));
updatecontroller.IsBackground = true;
updatecontroller.Start();

GamePadState state = GamePad.GetState(PlayerIndex.One);
sharedLeftTrig = state.Triggers.Left;
sharedLeftTrig = processRawThrottleData(sharedLeftTrig);
leftTrigValue.Invoke(new leftTrigDelegate(displayLeftTrig), displayLeftTrig);
sharedLeftStickX = state.ThumbSticks.Left.X;
sharedLeftStickX = processRawSteerData(sharedLeftStickX);
leftStickXValue.Invoke(new leftStickXDelegate(displayLeftStickX), displayLeftStickX);
sharedLeftStickY = state.ThumbSticks.Left.Y;
leftStickYValue.Invoke(new leftStickYVdelegate(displayLeftStickY), sharedLeftStickY);
sharedRightTrig = state.Triggers.Right;
sharedRightTrig = processRawThrottleData(sharedRightTrig);
rightTrigValue.Invoke(new rightTrigDelegate(displayRightTrig), sharedRightTrig);
sharedRightStickX = state.ThumbSticks.Right.X;
rightStickXValue.Invoke(new rightStickXDelegate(displayRightStickX), sharedRightS-
tickX);
sharedRightStickY = state.ThumbSticks.Right.Y;
rightStickYValue.Invoke(new rightStickYDelegate(displayRightStickY), sharedRightS-
tickY);
```

**Modelo – Termo autorização TCC Graduação**  
**Termo de Autorização para Publicação Eletrônica na Biblioteca Digital da**  
**Universidade do Oeste de Santa Catarina – Unoesc**

1. Identificação do material bibliográfico:

TCC Graduação

2. Identificação do documento/autor

Curso de Graduação: ENGENHARIA DE COMPUTAÇÃO

Área de concentração (Tabela CNPQ) 1.03.04.00-2 SISTEMAS DE COMPUTAÇÃO

Título do TCC: DESENVOLVIMENTO DE ROBÔ PARA EXPLORAÇÃO DE AMBIENTES PERIGOSOS UTILIZADOO SUPERVISÓRIO

Autor RAFAEL PABLO MASSOCATO

RG: 4755085 CPF: 051242099-84 Telefone (48) 98844-3447

E-mail: RAFAEL.MASSOCATO@GMAIL.COM

Orientador: DANIEL CALIXTO FAGONDE MORAES

CPF: 365253750-53 e-mail: DANIEL.MORAES@UNOESC.EDU.BR

Números de páginas: 74

Data de defesa: 18/12/17 Data de entrega do arquivo: 21/12/17

3. Informações de acesso ao documento

Este trabalho é confidencial?  Sim  Não

Pode ser liberado para publicação na Biblioteca Digital  Total  Parcial

Em caso de publicação parcial, assinale as permissões:

Sumário

Resumo

Bibliografia

Outras restrições:

Na qualidade de titular dos direitos de autor da publicação supracitada, de acordo com a Lei nº 9610/98, autorizo a Universidade do Oeste de Santa Catarina – Unoesc, a disponibilizar gratuitamente, sem resarcimento dos direitos autorais, conforme permissões assinadas acima, do documento, em meio eletrônico, na Rede Mundial de Computadores, no formato especificado<sup>2</sup>, para fins de leitura, impressão e/ou download pela Internet, a título de divulgação da produção científica gerada pela Universidade, a partir desta data.

4. Está sujeito a registro de patente?

Não

Sim. Informar o nº do processo de encaminhamento ao Escritório de Interação e Transferência de Tecnologia.

Local e data, 21/12/17

Rafael/1266  
Assinatura do autor

Calista  
Assinatura do orientador

<sup>1</sup> Esta classificação poderá ser mantida por até um ano a partir da data de defesa. A extensão deste prazo suscita justificativa junto à Coordenação do Curso. Todo resumo estará disponível para reprodução, conforme Regulamento do Programa de Pós-graduação e Pesquisa da Unoesc.

<sup>2</sup> Texto (PDF); imagem(JPG OU GIF); som (WAV, MPEG, AIFF, SND); vídeo (MPEG,AVI,QT), outros (específico da área)







# yes I want morebooks!

Buy your books fast and straightforward online - at one of world's fastest growing online book stores! Environmentally sound due to Print-on-Demand technologies.

Buy your books online at  
**[www.morebooks.shop](http://www.morebooks.shop)**

Compre os seus livros mais rápido e diretamente na internet, em uma das livrarias on-line com o maior crescimento no mundo!  
Produção que protege o meio ambiente através das tecnologias de impressão sob demanda.

Compre os seus livros on-line em  
**[www.morebooks.shop](http://www.morebooks.shop)**

KS OmniScriptum Publishing  
Brivibas gatve 197  
LV-1039 Riga, Latvia  
Telefax: +371 686 20455

[info@omnascriptum.com](mailto:info@omnascriptum.com)  
[www.omnascriptum.com](http://www.omnascriptum.com)

OMNI**S**criptum







