

# *Monografia de Graduação*

## **Projeto de um Veículo Aéreo Não Tripulado para Supervisão de Instalações de Petróleo e Gás Natural**

**João Paulo Ferreira Guimarães**

**Natal, julho de 2010**

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	O Grupo de Pesquisa AEROPETRO . . . . .	1
1.2	Objetivos Específicos Deste Trabalho . . . . .	1
<b>2</b>	<b>Fundametação Teórica</b>	<b>3</b>
2.1	O Microcontrolador . . . . .	4
2.2	Os Sensores . . . . .	4
2.2.1	Unidade de Medida Inercial - IMU . . . . .	5
2.2.2	Magnetômetros . . . . .	5
2.2.3	Global Positioning System - GPS . . . . .	5
2.2.4	Sonar . . . . .	5
2.3	Os Atuadores . . . . .	6
2.3.1	Os Motores Elétricos . . . . .	7
2.4	Formas de Comunicação . . . . .	8
2.4.1	Serial . . . . .	8
2.4.2	I2c . . . . .	9
2.4.3	USB . . . . .	9
<b>3</b>	<b>Metodologia</b>	<b>10</b>
3.1	O Veículo Aéreo . . . . .	10
3.1.1	O Protótipo Inicial . . . . .	11
3.1.2	O Protótipo Final . . . . .	13
3.2	Projeto e Dimensionamento . . . . .	13
3.3	A Arquitetura Escolhida . . . . .	16
3.4	A Eletrônica Embarcada . . . . .	16
3.4.1	A Interface Microcontrolada . . . . .	18
3.4.2	O Acionamento dos Motores . . . . .	18
3.4.3	GPS - Global Positioning System . . . . .	20
3.4.4	Sonar . . . . .	21
3.4.5	Bússola . . . . .	22
3.4.6	O Link De Rádio . . . . .	22
3.4.7	IMU - Unidade de medida Inercial . . . . .	22
3.5	A Interface de Software . . . . .	23
3.5.1	A classe C++ . . . . .	23
<b>4</b>	<b>Resultados e Discussões</b>	<b>25</b>
<b>5</b>	<b>Conclusões</b>	<b>26</b>
<b>6</b>	<b>Referências Bibliográficas</b>	<b>27</b>

<b>7</b>	<b>Anexos</b>	<b>28</b>
7.1	Esquemáticos contendo os circuitos desenvolvidos . . . . .	28
7.2	Código referente a classe C++ implementada . . . . .	28
	7.2.1 hexarotor.h . . . . .	28
7.3	Lista de componentes . . . . .	29
	7.3.1 Placa Microcontrolada 1 . . . . .	29
7.4	Placa Microcontrolada 2 . . . . .	29

# Lista de Figuras

1.1	Itens destacados em vermelho ressaltam as contribuições deste trabalho. . . . .	2
2.1	Vários exemplos de veículos aéreos não tripulados - <b>VANT</b> . . . . .	3
2.2	Exemplo de um microcontrolador. . . . .	4
2.3	Esquema que mostra os três eixos (Roll - X, Pitch - Y, Yaw - Z) onde atuam os sensores. . . . .	5
2.4	Grid imaginário que envolve o planeta e traça coordenadas de latitude e longitude que são informadas pelos satélites aos receptores GPS. . . . .	6
2.5	Funcionamento de um Sonar exemplificado . . . . .	6
2.6	Exemplo de funcionamento de um PWM. De acordo com a largura do pulso, teremos a tensão final aplicada sobre o motor. Neste caso, 50 e 20 respectivamente. Tal largura se repete num dado período T. . . . .	8
2.7	Funcionamento de um pseudo-PWM. A velocidade mínima é alcançada com uma largura de 1ms, a média com 1,5ms enquanto que a máxima a 2ms. . . . .	8
2.8	Esquema básico mostra a ligação ao barramento I2C de um microcontrolador mestre, um outro microcontrolador escravo além de conversores digital-analógico e analógico-digital também como escravos. O valor das resistências Rp podem variar de aplicação para aplicação e geralmente são informadas nos datasheets dos fornecedores dos dispositivos. . . . .	9
3.1	Exemplo de um Quadrotor. . . . .	10
3.2	Esquema que mostra o sentido de giro dos rotores. . . . .	11
3.3	Exemplo de um Hexarotor. VANT com 6 rotores. . . . .	11
3.4	Esquema que mostra o sentido de giro dos motores do Hexarotor. Os momentos são cancelados em cada um dos três eixos. . . . .	12
3.5	Primeiro protótipo do quadrotor. . . . .	13
3.6	Protótipo Final - Hexarotor . . . . .	14
3.7	Esquema que mostra a organização da arquitetura do Sistema Robótico proposto. .	16
3.8	Esquema que mostra a organização dos componentes distribuídos na arquitetura proposta. . . . .	17
3.9	Circuito desenvolvido numa protoboard que mostra o Microcontrolador PIC conectado ao GPS. O microcontrolador serve de interface entre o computador e os diversos sensores e atuadores do robô. . . . .	18
3.10	Esquema de comunicação entre o computador e os motores. . . . .	19
3.11	Exemplo de um motor Brushless. Este é um exemplar para aeromodelismo profissional produzido pela empresa Alemã Hacker. . . . .	20
3.12	Experimento realizado com o objetivo de validar o funcionamento do GPS. Se nota um erro no canto superior esquerdo. . . . .	21
3.13	Placas de circuito impresso. A placa da esquerda contém o transmissor de rádio acoplado a um computador comum via USB. A da direita foi embarcada no helicóptero e é responsável pela recepção da mensagem de rádio, interpretação e geração das ondas de acionamento dos motores. . . . .	22

3.14	Mesa de ensaio de empuxo. Microcontrolador situado na <i>protoboard</i> envia o sinal do pseudo-pwm ao controlador eletrônico de velocidade que aciona os motores Brushless.	23
3.15	Unidade de Medida Inercial utilizado no robô, produzida e vendida pela empresa Sparkfun. . . . .	23
4.1	Arquitetura de Hardware e Software do Robô AEROPETRO. . . . .	25
7.1	Circuito contendo a bussola e a IMU . . . . .	30
7.2	Circuito contendo o acionamento dos motores e o GPS . . . . .	31

**AEROPETRO** - Projeto de um Veículo Aéreo Não  
Tripulado para Supervisão de Instalações de Petróleo e Gás

João Paulo Ferreira Guimarães

13 de julho de 2010

# Capítulo 1

## Introdução

Em instalações industriais de petróleo e gás natural, encontramos um denso sistema de dutos e tanques que visam o transporte e armazenamento de substâncias inflamáveis, e essas muitas vezes encontram-se a elevadas pressões. Isso torna a segurança um fator crítico na indústria petroquímica. Na tentativa de prevenir qualquer tipo de acidente, existem atualmente em tais instalações, inúmeros sistemas de segurança. O problema se dá quando alguns fatores tais quais corrosão, fadiga mecânica ou até uma simples falha humana, por possuírem um difícil controle, deixam vulneráveis até os mais confiáveis sistemas que, em última instância, acabam por apresentar derramamento de produto por vazamento ou explosões que acabam provocando incêndios e desastres ecológicos, por exemplo.

Impactos ambientais e econômicos circundam tais acidentes e sendo assim, quanto mais rapidamente detectados e neutralizados, menores são os efeitos dos mesmos. No que se diz respeito ao campo relacionado a detecção de situações anômalas, a construção de sistemas automáticos de inspeção e supervisão através de controle ou sensoriamento remoto, vêm se destacando como uma entre as inúmeras técnicas utilizadas, em especial em situações perigosas e locais que são de difícil acesso para o ser humano.

Um sistema automático que representa uma solução para esse problema são os veículos aéreos não-tripulados. Eles são mais conhecidos pelas siglas **VANT**( *Veículos Aéreos Não Tripulados UAV* ou (*Unmanned Aerial Vehicle*). Eles aliam requisitos importantes como baixo-custo, segurança e eficiência.

### 1.1 O Grupo de Pesquisa AEROPETRO

Este trabalho faz parte de um projeto maior chamado AEROPETRO que visa a construção de um protótipo de um robô autônomo, que possa atuar em instalações de petróleo e gás natural, capaz de realizar tarefas de inspeção e supervisão através de um sistema de visão computacional que visa detectar situações de risco. Esse robô se trata de um veículo aéreo não tripulado.

Poucos são os sistemas encontrados no mercado e capazes de realizar o tipo de supervisão aérea proposto. Além disso, são importados e com algumas limitações que comprometem a detecção automática de situações anômalas e a sua navegação autônoma. Assim, o projeto visa o desenvolvimento de um produto nacional com maior autonomia e com custo inferior.

### 1.2 Objetivos Específicos Deste Trabalho

O objetivo deste trabalho é a implementação no que se diz respeito à infra-estrutura de hardware e software básico necessário para o desenvolvimento do protótipo do VANT propriamente dito. Isso significa o desenvolvimento de firmwares de controle que serão embarcados em placas microcontroladas que interfaceiam os diversos sensores e atuadores do robô com um computador embarcado. Além disso é objetivo primário também a implementação de uma classe em linguagem

C++ que tem como objetivo interfacear esses periféricos do robô. Este servirá como plataforma base para testes dos projetos de estabilização, navegação e detecção de situações anômalas que serão desenvolvido pelos doutorandos da equipe.

Na figura 1.1 podemos observar um diagrama de blocos bem resumido dessa infra-estrutura de hardware e software que deve ser implementada.

Teremos embarcado no robô um computador que deterá as tarefas que exigem o maior nível de processamento. É nesse nível que os algoritmos de estabilização, navegação e detecção de anômalias irá trabalhar além de uma classe em C++ (USBrobot) que estabelece a comunicação entre o sistema operacional do computador embarcado e microcontroladores com interface USB (Universal Serial Bus).

Para adquirir dados provindo de sensores e atuar sobre motores por exemplo, recorreremos a uma eletrônica microcontrolada que visa gerir essa interface entre os algoritmos de alto nível com os diversos sensores e atuadores do robô.

No lado direito da figura, podemos observar diversas tarefas que devem ser realizadas em cada uma dessas camadas. Destacadas em vermelho, são as contribuições desenvolvidas no decorrer deste trabalho.

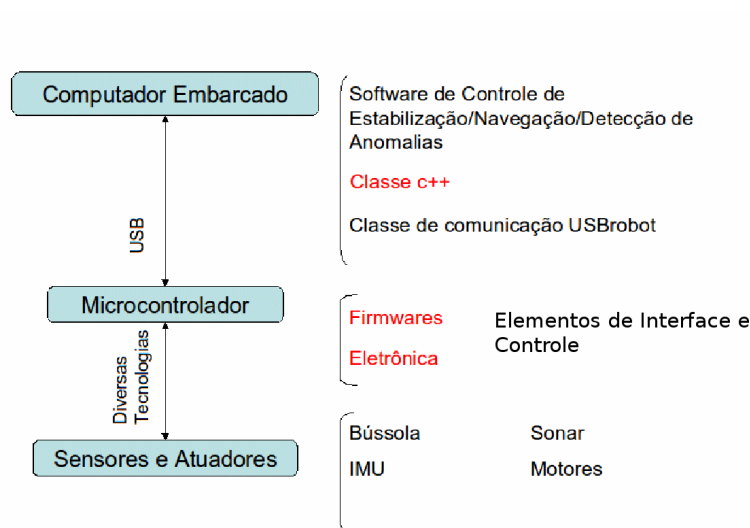


Figura 1.1: Itens destacados em vermelho ressaltam as contribuições deste trabalho.



## Capítulo 2

# Fundametação Teórica

Atualmente, o desenvolvimento de robôs aéreos autônomos vem incrementando a área de pesquisa em robótica. Tais veículos têm um potencial comercial elevado, principalmente em aplicações de inspeção de infra-estruturas como pontes e hidrelétricas, e supervisão de instalações industriais.

Os chamados **VANTs**, que podem ser vistos na figura 2.1, referenciam a qualquer veículo aéreo (avião, helicóptero, dirigível, etc) que navegue sem tripulação humana.



Figura 2.1: Vários exemplos de veículos aéreos não tripulados - VANT

Tais veículos aéreos devem ser equipados com uma instrumentação tal que esta possa captar dados e informações relevantes do ambiente para que possam auxiliar na realização da missão para quais foram construídos. Essas informações servirão tanto no quesito da missão específica como por exemplo a detecção de situações anômalas em instalações de petróleo e gás como também na própria navegação e estabilização do robô aéreo.

Neste capítulo analisaremos alguns aspectos teóricos sobre o sistema proposto assim como faremos uma breve introdução as diversas tecnologias utilizadas durante esse trabalho.

## 2.1 O Microcontrolador

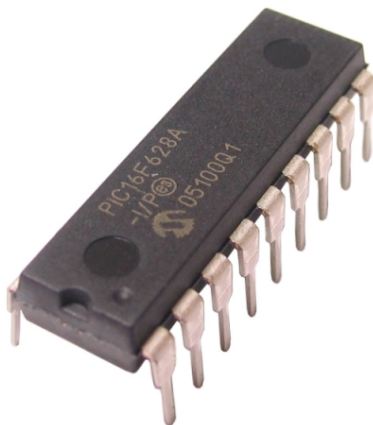


Figura 2.2: Exemplo de um microcontrolador.

Um microcontrolador, visto na figura 2.2 é um computador-num-chip. Um encapsulamento que contém um processador, memória e periféricos de entrada/saída. É um microprocessador que pode ser programado para funções específicas, em contraste com outros microprocessadores de propósito geral. Eles são embarcados no interior de algum outro dispositivo para que possam controlar as funções ou ações do mesmo.

Os microcontroladores se diferenciam dos processadores, pois além dos componentes lógicos e aritméticos usuais de um microprocessador de uso geral, o microcontrolador integra elementos adicionais em sua estrutura interna, como memória de leitura e escrita para armazenamento de dados; Memória somente de leitura para armazenamento de programas; EEPROM para armazenamento permanente de dados; Dispositivos periféricos como conversores analógico/digitais (CAD); Conversores digitais/analógicos (CDA); Interfaces de entrada e saída de dados; Etc.

Com frequências de clock de MHz (Megahertz) ou as vezes menos, os microcontroladores operam a uma frequência muito baixa se comparados com os microprocessadores atuais. No entanto são adequados para a maioria das aplicações usuais como por exemplo controlar uma máquina de lavar roupas ou uma esteira de chão de fábrica. O seu consumo em geral é relativamente pequeno, normalmente na casa dos miliwatts e possuem geralmente habilidade para entrar em modo de espera (Sleep) aguardando por uma interrupção ou evento externo, como por exemplo o acionamento de uma tecla, ou um sinal que chega via uma interface de dados. O consumo destes microcontroladores em modo de espera pode chegar na casa dos nanowatts, tornando-os ideais para aplicações onde a exigência de baixo consumo de energia é um fator decisivo para o sucesso do projeto.

## 2.2 Os Sensores

É necessário que o robô consiga perceber o mundo ao seu redor. Ele fará isso com auxílio de diversos sensores.

A seguir temos uma série de sensores que serão de extrema importância ao funcionamento do robô. Serão apresentadas nesta fundamentação teórica breves aspectos teóricos envolvidos no funcionamento dos mesmos.

### 2.2.1 Unidade de Medida Inercial - IMU

A Unidade de Medida Inercial (*Inertial Measurement Unit - IMU*) é um equipamento eletrônico capaz de medir a taxa de rotação sobre os eixos X, Y e Z e aceleração ao longo de direções longitudinais, verticais e laterais, conforme sugere a figura 2.3.

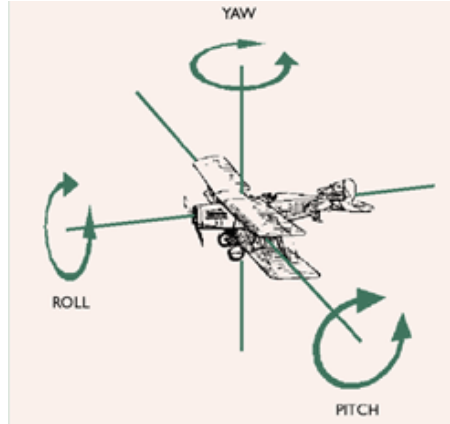


Figura 2.3: Esquema que mostra os três eixos (Roll - X, Pitch - Y, Yaw - Z) onde atuam os sensores.

Isso é possível graças a integração entre dois tipos diferentes de sensores. São eles:

**Girômetros** Dispositivo que utiliza dos princípios de conservação do momento angular para estimar uma orientação.

**Acelerômetros** Dispositivos capazes de medir a força específica.

A partir de um arranjo de tais sensores é que essa a IMU provê dados para que se faça possível o controle de estabilidade de veículos aéreos através da estimativa da posição e do ângulo de inclinação que o robô se encontra em relação a uma reta imaginária perpendicular a superfície.

### 2.2.2 Magnetômetros

O Magnetômetro é um instrumento usado para medir a intensidade, direção e sentido de campos magnéticos em sua proximidade. Assim, informam baseados no campo magnético do planeta o norte absoluto.

Esse dado dá ao robô um referencial absoluto que será utilizado nos algoritmos de controle e navegação.

### 2.2.3 Global Positioning System - GPS

O Sistema de Posicionamento Global, popularmente conhecido por GPS (Global Positioning System), inclui um conjunto de satélites artificiais que circundam o planeta e tem como objetivo fornecer a diversos dispositivos receptores móveis, através de um link de rádio, a posição do receptor com referencia as coordenadas terrestres. Tal posição é fornecida através de coordenadas de latitude e longitude. De posse desses dados, podemos dotar o robô de um senso de posicionamento global. Veja a figura 2.4.

### 2.2.4 Sonar

Vindo do termo em inglês Sound Navigation And Ranging, esse sensor utiliza do princípio da reflexão das ondas sonoras para detectar objetos e obstáculos que estejam a sua frente. Um

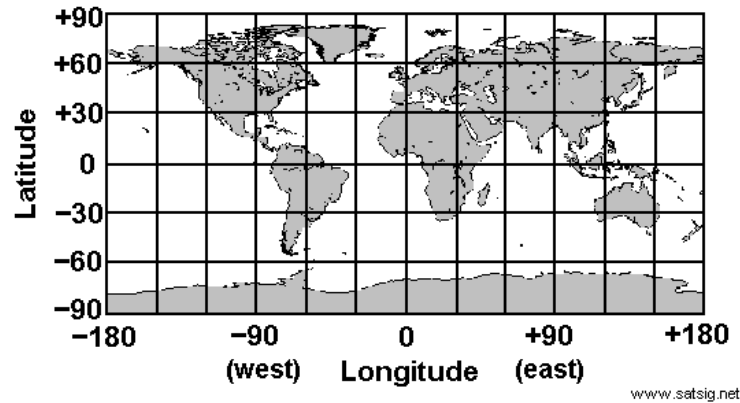


Figura 2.4: Grid imaginário que envolve o planeta e traça coordenadas de latitude e longitude que são informadas pelos satélites aos receptores GPS.

aparelho encapsulado ao próprio sensor é responsável pela emissão das ondas sonoras de alta frequência e um outro dispositivo é acoplado com a função de receber as reflexões da onda gerada. Pelo tempo passado entre a emissão e recepção da onda e sabendo a velocidade da propagação dessa no meio podemos inferir a distância que o sensor se encontra do objeto refletor. A figura 2.5 exemplifica esse funcionamento.

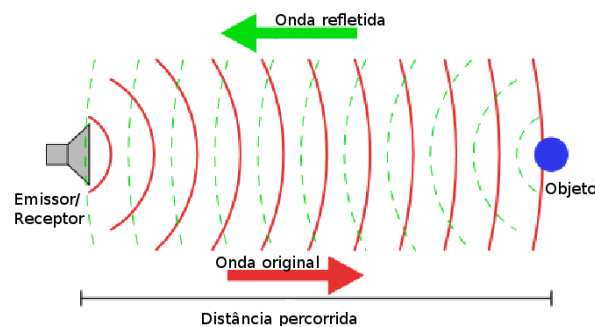


Figura 2.5: Funcionamento de um Sonar exemplificado

Nas tarefas de pouso por exemplo, o sonar pode ser muito útil. O robô deverá saber qual a distância em que ele se encontra da superfície para que consiga calcular de forma precisa as referências de velocidades ideais para um pouso suave.

## 2.3 Os Atuadores

O Atuador é um elemento que produz movimento, atendendo a comandos que podem ser manuais ou automáticos. Se pensarmos nos sensores como uma maneira de "ler" o mundo podemos encarar os atuadores como uma forma de "escrever" no mundo. Eles podem ser de diversos tipos; Pode-se citar atuadores de movimento induzido por cilindros pneumáticos ou cilindros hidráulicos e motores, dispositivos rotativos com acionamento de diversas naturezas.

### 2.3.1 Os Motores Elétricos

Motor elétrico é uma máquina destinada a transformar energia elétrica em mecânica. É o mais usado de todos os tipos de motores, pois combina as vantagens da energia elétrica, ou seja, baixo custo, facilidade de transporte, limpeza e simplicidade de comando – com sua construção simples, custo reduzido, grande versatilidade de adaptação às cargas dos mais diversos tipos e melhores rendimentos.

Para fazer com que um motor elétrico de corrente contínua atinja uma certa velocidade é necessário que se forneça a este uma determinada diferença de potencial elétrico em seus terminais. Sendo assim, variando a tensão aplicada a esse dispositivo podemos obter diversas velocidades de giro, proporcionais a esta tensão.

Entretanto, muitas vezes não dispomos de fontes de tensão variáveis como por exemplo num microcontrolado. Nele dispomos de diversas portas de saída de dados entretanto, todas elas são digitais. Isso significa dizer que só dispomos de dois possíveis valores de tensões: 0V e 5V, referentes ao bit lógico 0 e ao bit lógico 1 respectivamente. Assim, ligando diretamente o motor a uma das portas do microcontrolador só poderíamos a princípio controlar se o motor está ligado a força total ou desligado.

Visando contornar essa situação, se utiliza de uma estratégia baseada no chaveamento de um sinal elétrico contínuo a fim de se gerar uma tensão menor e apropriada. Por exemplo: Se temos uma fonte de tensão de 5V mas queremos fornecer somente 2,5V ao motor elétrico ou seja, queremos que ele gire com apenas a metade da velocidade máxima. Sendo assim, temos de ligar e desligar o motor a partir da geração de um sinal elétrico periódico através da porta de saída digital do microcontrolador tal que nela, metade do tempo tenhamos o nível lógico alto(5V), que forçaria o motor a impor sua velocidade máxima, enquanto que a outra metade do tempo o microcontrolador esteja colocando em sua porta digital o nível lógico baixo(0V), desligado o motor elétrico.

Mantendo a periodicidade na geração desse sinal elétrico, o motor elétrico vai tratar essa onda quadrada com uma única onda de amplitude menor, no caso, somente metade da tensão inicial como que numa média. Essa onda gerada a partir do chaveamento do motor pode ser vista como uma modulação por largura de pulso, o chamado **PWM** - Pulse-Width Modulation. Quanto mais tempo a onda quadrada estiver em alto dentro de um dado período, denominado Duty Cycle, maior o valor final da tensão visto pelo motor. Sendo assim, a partir do controle da largura do pulso, ou seja, o tempo em que o motor está ligado e desligado, podemos controlar diretamente a velocidade de giro de um motor elétrico.

Essa relação pode ser vista de forma matemática pela fórmula do chamado nível DC:

$$V_{mdia} = \frac{1}{T} * \int_0^T f(t) dt.$$

Onde  $V_{média}$  representa a tensão média final,  $t$  representa o tempo, e a função  $f$  é o sinal da onda quadrada gerado pelo microcontrolador.

Esse é o tipo de acionamento para os motores elétricos mais encontrados. Entretanto existe uma outra classe de motores elétricos de corrente alternada que exige um acionamento diferenciado. Nos motores do tipo **Brushless**, que significa sem escovas, não há nenhum ponto de contato mecânico entre o rotor e o estator para passagem de eletricidade. Por reduzir os atritos dentro de seu interior, esse tipo de motor elétrico apresenta uma maior eficiência no quesito de transformação de energia elétrica para mecânica, permitindo um menor consumo, uma maior potência e inclusive uma maior vida útil.

A única desvantagem atrelada a essa tecnologia é o seu acionamento que no caso, necessita de uma corrente alternada para funcionar. São ondas defasadas em 120 graus que alimentam esse tipo de dispositivo.

Geralmente as próprias fabricantes dos motores comercializam também dispositivos eletrônicos microcontrolados que visam facilitar o acionamento desse tipo de motor. Encapsulando o conjunto motor-driver, tais empresas vendem o produto final como motores sem escovas de corrente contínua. Os Controladores Eletrônicos de Velocidade, **ESC** - Eletronic Speed Control - são justamente esses drivers.

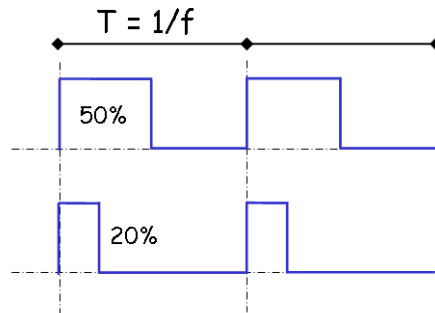


Figura 2.6: Exemplo de funcionamento de um PWM. De acordo com a largura do pulso, teremos a tensão final aplicada sobre o motor. Neste caso, 50 e 20 respectivamente. Tal largura se repete num dado período  $T$ .

Como os motores Brushless possuem uma grande aplicação na indústria do aeromodelismo, o sinal de entrada no ESC é exatamente o mesmo sinal que serve de acionamento para um servo-motor. Isso porque, os aeromodelos apresentam em sua composição geralmente inúmeros servo-motores. Sendo assim, no receptor que fica embarcado no aeromodelo, existem diversas saídas que tanto podem ser ocupadas por servo-motores quanto por um motor brushless acoplado a um ESC.

A figura abaixo mostra como deve ser feito o acionamento do ESC. Essa onda de largura variável é a mesma que aciona um servo-motor.

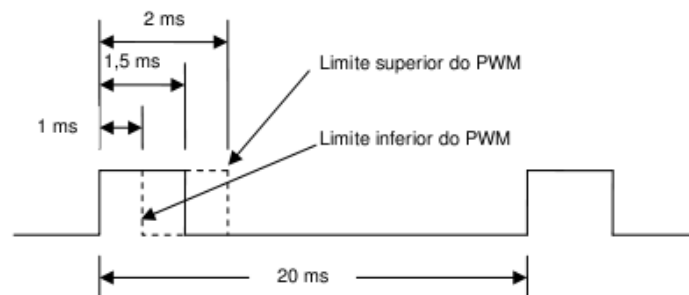


Figura 2.7: Funcionamento de um pseudo-PWM. A velocidade mínima é alcançada com uma largura de 1ms, a média com 1,5ms enquanto que a máxima a 2ms.

## 2.4 Formas de Comunicação

### 2.4.1 Serial

A comunicação serial é o processo de enviar dados um bit de cada vez, sequencialmente, num canal de comunicação ou barramento. É diferente da comunicação paralela, em que todos os bits de cada símbolo são enviados simultaneamente. A comunicação serial é usada em toda comunicação de longo alcance e na maioria das redes de computadores, onde o custo de cabos e as dificuldades de sincronização tornam a comunicação paralela impraticável. Para curtas distâncias, barramentos seriais estão se tornando cada vez mais comuns devido ao ponto em que as desvantagens dos barramentos paralelos (densidade de interconexão) superam suas vantagens de simplicidade.

### 2.4.2 I2c

I2c (Inter-Integrated Circuit) é um barramento serial multi-mestre desenvolvido pela Philips que é usado para conectar periféricos de baixa velocidade a uma placa mãe, a um sistema embarcado ou a um telefone celular. O nome significa Circuito Inter-integrado. Muito visto em microcontroladores e circuitos integrados, a forma serial de enviar os dados diminui os custos relacionados a confecção do hardware já que, a grosso modo, apenas um pino é necessário para o envio de qualquer quantidade de dados.

O I2C utiliza apenas duas linhas bidirecionais de dreno aberto, Dados Seriais (Serial Data - SDA) e Clock Serial (Serial Clock - SCL). Este protocolo especifica dois sinais de comunicação, um com o sinal de clock (gerado pelo mestre), e outro de dados, bidirecional. Um esquema mostrando as conexões pode ser visto na figura 2.8.

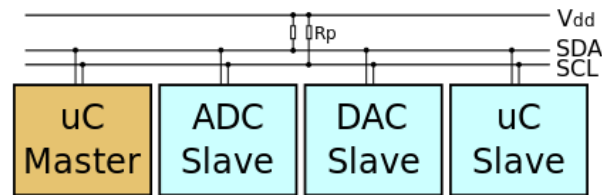


Figura 2.8: Esquema básico mostra a ligação ao barramento I2C de um microcontrolador mestre, um outro microcontrolador escravo além de conversores digital-analógico e analógico-digital também como escravos. O valor das resistências Rp podem variar de aplicação para aplicação e geralmente são informadas nos datasheets dos fornecedores dos dispositivos.

### 2.4.3 USB

A interface USB (Universal Serial Bus) é um meio de conexão entre um computador host e um certo número de periféricos. Ela foi inicialmente criada para substituir um conjunto de interfaces de comunicações mais lentas (portas paralelas e seriais) por uma única interface capaz de suportar todos esses dispositivos. Com o passar dos tempos, a USB deixou de ser usada apenas para conectar dispositivos de baixa velocidade, como os acima citados, e passou a ser a interface de comunicação padrão com praticamente qualquer dispositivo que possa ser conectado a um computador. A última revisão da especificação adicionou conexões high speed com um limite de velocidade de 480 Mbps.

## Capítulo 3

# Metodologia

### 3.1 O Veículo Aéreo

Boa mobilidade e possibilidade de vôo estacionário nos levaram a escolher um helicóptero como protótipo a ser desenvolvido. Entretanto, os helicópteros tradicionais de 2 rotores possuem modelos dinâmicos difíceis de serem controlados e com um alto número de variáveis de estado.

Os veículos multi-rotores tratam de compensar tais dificuldades.

Um Quadrotor, ilustrado na figura 3.1, é um tipo de helicóptero que utiliza de quatro rotores fixos, colocados em cada vértice de uma estrutura geralmente na forma de um quadrado e sendo que os rotores adjacentes giram em sentidos opostos para equilibrar os momentos e produzir os movimentos desejados como mostra a figura 3.2. O controle de vôo é feito a partir da variação das velocidades em cada um dos motores. São muito mais simples e relativamente mais fáceis de controlar do que os helicópteros comuns.

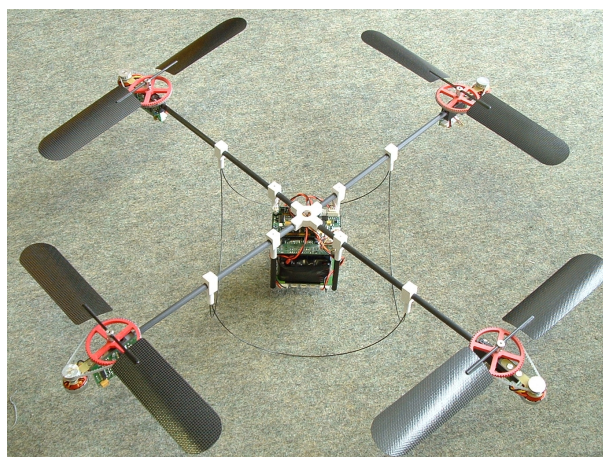


Figura 3.1: Exemplo de um Quadrotor.

Uma outra opção a ser analisada é o **Hexarotor**. Um exemplo do mesmo pode ser vista na figura 3.3.

A idéia é que em cada um dos três eixos do hexarotor temos os dois rotores que giram em sentidos opostos. Sendo assim o momento é cancelado em cada eixo. A diferença de empuxos entre os vários conjuntos de rotores é que produz os movimentos desejados ao veículo. Em contrapartida, caso haja alguma falha de um dos rotores teremos mais condições de controlar o robô para realizar um pouso de emergência por exemplo. Isso porque nessa configuração temos a presença de redundâncias nos atuadores.



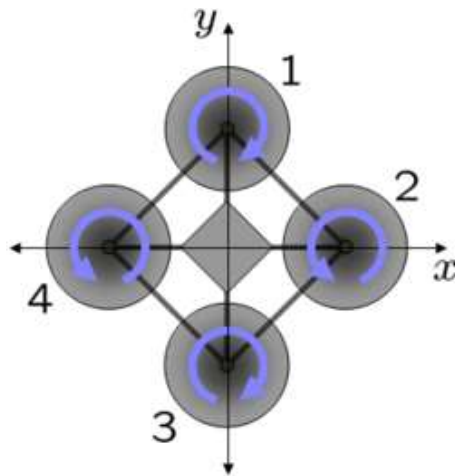


Figura 3.2: Esquema que mostra o sentido de giro dos rotores.

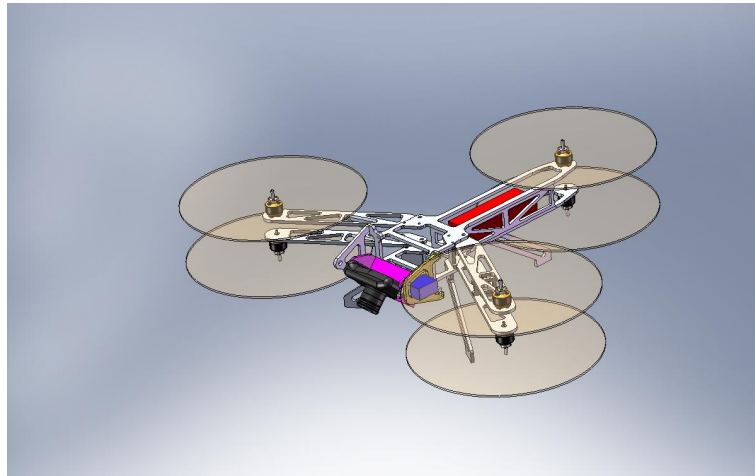


Figura 3.3: Exemplo de um Hexarotor. VANT com 6 rotores.

Um esquema pode ser visto na figura 3.4.

Uma desvantagem apresentada por essa configuração de seis rotores é que, quando posicionamos um rotor sobre o outro, não obtemos exatamente o dobro do empuxo como poderia ser esperado.

Durante o desenvolvimento deste trabalho, um primeiro protótipo baseado no esquema de um Quadrotor foi desenvolvido. Entretanto, o protótipo final é um Hexarotor.

### 3.1.1 O Protótipo Inicial

Foi construído empiricamente um helicóptero do tipo quadrotor, ou seja com quatro rotores distribuídos numa estrutura em forma de "X" como mostra a figura 3.5.

Tal protótipo tinha o objetivo de embarcar as diversas funcionalidades da eletrônica que fossem desenvolvidas a princípio. No caso, a primeira delas foi o acionamento dos motores.

Para esse robô, foram utilizados motores elétricos de corrente contínua bem simples. O seu acionamento se deu através de um microcontrolador (Microchip 18f2550) capaz de gerar as quatro ondas de PWM necessárias para controlar as velocidades de cada um dos motores.

Também foi desenvolvido um link para controle das velocidades remotamente via rádio. Uma

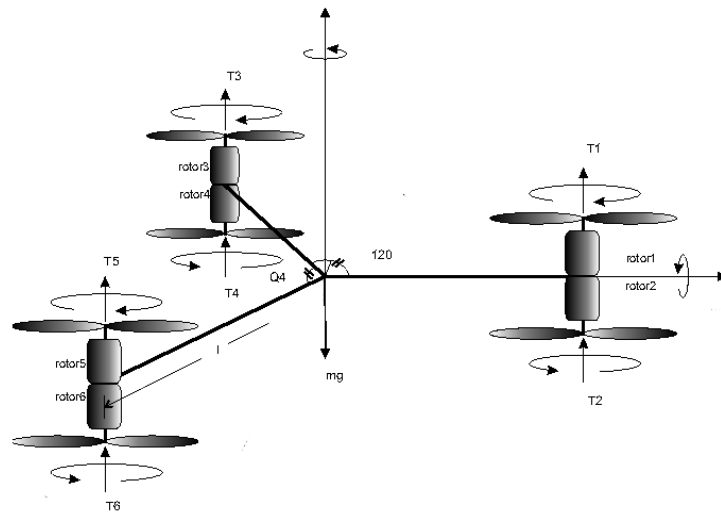


Figura 3.4: Esquema que mostra o sentido de giro dos motores do Hexarotor. Os momentos são cancelados em cada um dos três eixos.

placa, também dotada do mesmo microcontrolador, foi desenvolvida para se comunicar via USB com um computador. Este deveria ter um sistema operacional linux rodando e uma série de bibliotecas referenciadas em [10] para validar a comunicação.

No robô, uma placa era responsável por receber o pacote vindo do transceiver de rádio, decodificar a mensagem e alterar a velocidade dos motores. O uso desse protótipo foi bastante restrito pela qualidade do equipamento mecânico utilizando que apresentou um alto desgaste em pouco tempo. Entretanto, foi muito proveitosa a experiência nele adquirida para o desenvolvimento do projeto do protótipo final.



Figura 3.5: Primeiro protótipo do quadrotor.

### 3.1.2 O Protótipo Final

Ao contrário do primeiro protótipo construído, o protótipo final foi projetado para atender a todos os requisitos envolvidos. Podemos ver o resultado final na figura 3.6.

O detalhamento do projeto pode ser visto nos itens a seguir.

## 3.2 Projeto e Dimensionamento

No que se diz respeito ao projeto de um veículo aéreo qualquer, inúmeros desafios podem ser encontrados. O VANT deve conter uma instrumentação mínima necessária para que ele possa realizar a sua missão.

No caso, ele deve suportar uma massa relativa a essa instrumentação que o possibilite detectar situações anômalas em instalações de petróleo e gás. Essa instrumentação consiste de inúmeros sensores para a captura de dados do ambiente, de um hardware para coletar os vários dados destes sensores e repassar a um computador que os processará para realizar o controle de estabilidade e a navegação. Aliado a navegação, temos as câmeras. Não podemos nos esquecer também das baterias que irão alimentar todo o sistema e além disso, das próprias massas dos motores, hélices e da estrutura de sustentação propriamente dita.

Podemos citar alguns itens que deverão estar presentes no robô:

- Computador Embarcado;
- Motores
- Hélices
- Estrutura
- Baterias
- Sensores
- Microcontroladores

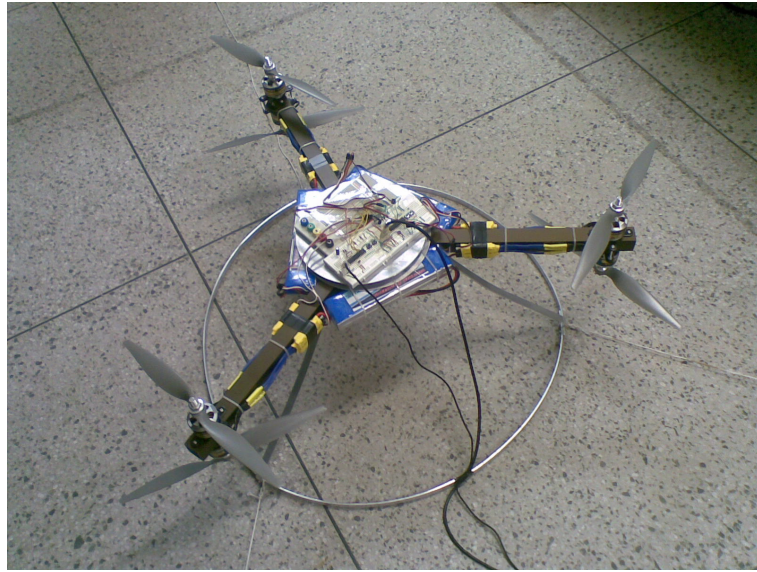


Figura 3.6: Protótipo Final - Hexarotor

- Câmera

No projeto inicial, pensamos em projetar um veículo aéreo que pudesse erguer até 10kg de massa que correspondiam a todos esses itens mencionados mais o próprio peso da estrutura. Vimos que para que esse protótipo pudesse ter uma autonomia mínima considerável, cerca de 9 minutos mínimos para a realização dos testes de controle e navegação, boa parte desses 10kg seriam utilizados para embarcar as baterias necessárias para tal feito. Isso porque os motores tinham um consumo muito elevado. Além disso, o helicóptero teria dimensões incoerentes com o projeto, quase 2 metros quadrados graças as imensas hélices necessárias. E, quando simulávamos a redução do tamanho do mesmo, tínhamos que reduzir as hélices utilizadas e com isso, tínhamos uma perda de empuxo muito grande que reduziriam nossa capacidade de embarcar a eletrônica necessária para a realização da missão. Para compensar a perda de empuxo, teríamos que aumentar as rotações nos motores que conseqüentemente, aumentavam o consumo. O tempo de voo caía para mínimos 3 minutos.

Com o amadurecer do problema, percebemos então que de nada adiantava aumentar a potência do motor. Seria preciso uma nova estratégia baseada na redução da massa total envolvida no veículo aéreo para que motores menos potentes fossem usados. Tais motores consomem menos energia e necessitam de hélices menores. Começamos os estudos com as simulações envolvendo um robô um pouco menor, escolhemos motores com potência também menor e que consumiam cerca de 50 % a menos que os motores iniciais. Baterias menos potentes seriam necessárias e conseqüentemente, mais massa poderia ser embarcada.

Sendo assim, o grande desafio consiste em encontrar um conjunto bateria-hélice-motor que possa gerar um empuxo tal, que seja suficiente para sustentar a massa de todos esses componentes discutidos. Além disso, é necessário que exista um tempo mínimo considerável relacionado a autonomia de voo do robô para que ele possa cumprir sua missão. Um outro fator que deve ser levado em conta é o tamanho. Ele não poderá ter dimensões tais que possam vir a interferir no andamento da missão.

Depois de algumas reuniões com toda a base de pesquisa, foi definido o conjunto que parecia ideal ao projeto. Conseguimos ao fim um projeto que era mais leve (cerca de 5,5Kg), que conseguia suportar toda a eletrônica requerida, e que nos dava uma autonomia de voo de quase 12 minutos. Além disso, as dimensões do helicóptero também foram reduzidas, chegando a estimativa de menos de 1 metro quadrado.

Depois que foi definida toda a massa que era necessária erguer, uma pesquisa exaustiva começou em busca de um conjunto bateria-hélice-motor que melhor se adequasse aos requisitos.

Ao final, os seguintes componentes foram selecionados:

<b>Motores</b>	E-Max Brushless Motor BL2815/09
<b>Baterias</b>	Thunder Power RC Li-Polymer 4450mAh 14.8V 22C cont/ 40C Burst
<b>ESCs</b>	E-Max 50A Brushless Electric Speed
<b>Helices</b>	APC 12x6 Thin Electric Propeller (Pusher)

Nossa pesquisa tomou rumo baseado nos dados que os próprios fornecedores traziam na documentação dos seus motores. Tais fornecedores produzem esses motores para o Aeromodelismo Profissional. Softwares de simulação baseados nesses dados também foram utilizados.

A seguir, temos uma listagem de todos os itens que deverão ser embarcados com suas respectivas massas.

<b>Componente</b>	<b>Peso</b>	<b>Quantidade</b>	<b>Total</b>
Motor Brushless	112g	6	672g
ESC	36g	6	216g
Bateria	464g	3	1392g
Hélice	25g	6	150g
Estrutura	1370g	1	1370g
Netbook	933g	1	933g
Payload Extra	700g	1	700g
<b>TOTAL</b>	×	×	<b>5433g</b>

O payload extra, último item da tabela, deve corresponder a toda a eletrônica que virá a ser embarcada para viabilizar o controle e a navegação do veículo aéreo além da câmera.

A tabela a seguir mostra esses componentes e suas respectivas massas:

<b>Componente</b>	<b>Peso</b>	<b>Quantidade</b>	<b>Total</b>
IMU	28g	1	28g
GPS	30g	1	30g
Bússola	20g	1	20g
Webcam	130g	1	130g
Microcontrolador	5g	2	10g
Eletrônica Auxiliar	50g	1	50g
<b>Total</b>	x	x	<b>278g</b>

Podemos notar, após estudar a tabela dos componentes, que a massa total do protótipo é de cerca de 5.4kg. Sendo que o empuxo gerado pelo sistema bateria-motor-hélice escolhido nos dá um empuxo de cerca de 2.6Kg. O Hexarotor é dotado de 3 desses conjuntos geradores de empuxo o que nos dá um empuxo resultante de quase 8Kg. Sendo que essa é o empuxo máximo que exige muito dos motores e conseqüentemente, das baterias, que descarregam muito mais rápido.

Para aumentar o tempo de voo, é necessário fazer com que os motores trabalhem sob menos esforço. Sendo assim, quanto menor a massa total do protótipo, maior o tempo de voo.

Isso significa que temos alguns quilos livres que poderão ser utilizadas para adicionar uma eletrônica que conterà novas funcionalidades ao robô ou ainda para embarcar mais baterias visando o aumento da autonomia de voo. Autonomia essa que foi estimada inicialmente em torno de 10 a 12 minutos.

Com relação ao material da estrutura física propriamente dita, optou-se pelo uso do alumínio na construção do protótipo já que este é mais facilmente encontrado no mercado e possui uma boa resistência e maior facilidade no quesito de usinagem. Entretanto, para um produto final, é indispensável o uso de fibra de carbono pois esse material além de ser mais resistente, é mais leve e econômico. O único empecilho no uso do mesmo está na manipulação/usinagem.

### 3.3 A Arquitetura Escolhida

Uma grande preocupação foi pensar numa arquitetura ideal para o sistema robótico em questão. Segundo Maranhão [7], a arquitetura de um sistema robótico refere-se a maneira como as estruturas heterogêneas de hardware e software interagem para controlar o robô. É necessária uma devida organização na forma com que os dados irão trafegar como por exemplo, a mudança gradual da velocidade dos motores para que um pouso possa ser realizado de forma suave.

A equipe decidiu abordar o problema na forma de uma arquitetura no modelo *Mestre-Escravo*. Tal escolha pode ser melhor justificada sob análise do trabalho [7]. Segundo este, no modelo mestre-escravo uma única unidade de maior poder computacional (o mestre) é responsável pelo processamento e administração do sistema. As demais unidades (os escravos) realizam apenas tarefas específicas requisitadas pelo mestre. No que tange a comunicação ela é feita apenas entre o mestre e um escravo, nunca de escravo para escravo. Embora esse modelo pareça demasiadamente restritivo, ele é suficiente para o sistema proposto. Contamos com apenas uma unidade mestre (o computador embarcado), que é responsável pelo processamento de imagem e controle, já os escravos são unidades microcontroladas responsáveis pelo acionamento dos motores, leitura de sensores, etc.

Um esquema representativo dessa arquitetura pode ser visto na figura 3.3.

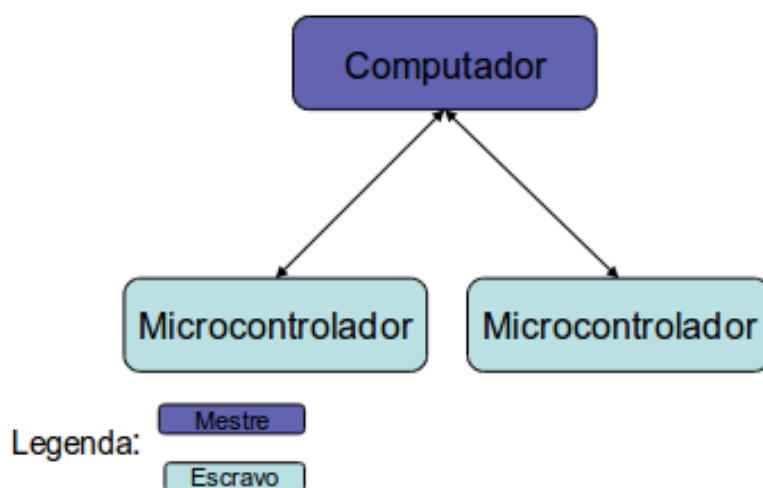


Figura 3.7: Esquema que mostra a organização da arquitetura do Sistema Robótico proposto.

### 3.4 A Eletrônica Embarcada

Tendo definido o projeto, uma outra fase foi iniciada, o projeto dos sistemas eletrônicos a ser embarcado no robô. Esta fase é um dos principais objetivos desse trabalho.

Tal eletrônica é muito importante pois servirá de interface entre os sensores e atuadores com o computador que virá a ser embarcado. Este computador fará o controle de estabilização e a navegação do robô. A seguir, podemos observar tudo o que deve estar embarcado no robô:

- Interface Microcontrolada

- Atuadores
  - Motores Brushless
- Sensores
  - GPS
  - Bússola
  - IMU
    - Acelerômetro
    - Girômetro
    - Magnetômetro
  - Sonar
- Computador Embarcado
- Câmeras

Tendo conhecimento da arquitetura Mestre-Escravo que será adotada no robô a partir da interface USB vimos que uma proposta seria usar apenas um microcontrolador como objeto de interface entre esses elementos. A principal vantagem dessa abordagem seria a redução da massa total da eletrônica e consequentemente do protótipo inteiro.

Entretanto, viu-se que a complexidade da programação em baixo nível, ou seja, no microcontrolador, teria sua complexidade muito aumentada principalmente se quiséssemos garantir o tempo-real na comunicação. Além disso, a massa correspondente a inclusão de mais um microcontrolador e sua eletrônica auxiliar em comparação ao ganho de massa do protótipo como um todo seria muito baixa.

Sendo assim, optou-se por dividir as tarefas de baixo nível em dois microcontroladores:

O primeiro deles seria responsável por um dos fatores mais críticos que é receber as referências de velocidade do controle de alto nível e acionar os seis motores com as respectivas ondas.

O segundo seria responsável por capturar os dados dos sensores da unidade de medida inercial e da bússola e encaminha-los aos programas de controle de estabilidade via USB.

O GPS tem uma frequência máxima de atualização de 1Hz, ou seja, 1 vez a cada segundo. Sendo assim, por ser um sistema com atualização considerada "lenta" pensou-se em coloca-lo junto ao microcontrolador responsável pelo acionamento dos motores.

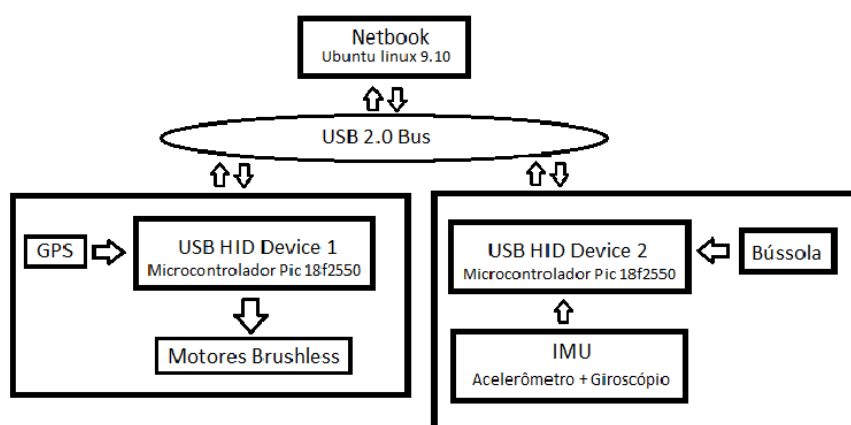


Figura 3.8: Esquema que mostra a organização dos componentes distribuídos na arquitetura proposta.

Assim teríamos o seguinte esquema operante na figura 3.8.

### 3.4.1 A Interface Microcontrolada

O trabalho [7] desenvolvido por um membro da equipe do projeto AEROPETRO, além de justificar a arquitetura adotada também mostra a implementação de uma interface para microcontroladores PIC da Microchip e computadores convencionais com sistemas operacionais Linux.

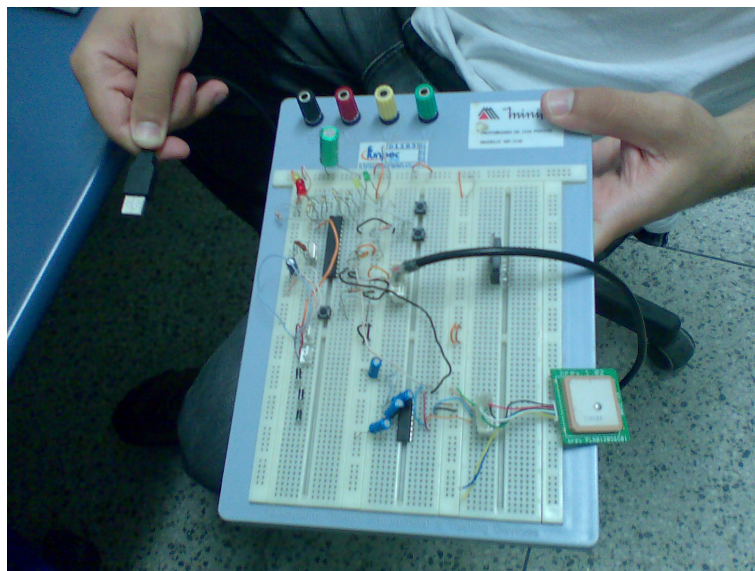


Figura 3.9: Circuito desenvolvido numa protoboard que mostra o Microcontrolador PIC conectado ao GPS. O microcontrolador serve de interface entre o computador e os diversos sensores e atuadores do robô.

A interface microcontrolada é de extrema importância ao projeto do sistema robótico em questão. Esta servirá de ponte entre o computador que exercerá o controle sobre o robô e os seus respectivos sensores e atuadores. Por exemplo, numa simples rotina de pouso: Para que nenhum dano grave seja causado a este devemos executar um procedimento de diminuir a velocidade dos motores aos poucos, ou seja, o computador que virá a ser embarcado, deverá requisitar a altitude em que ele se encontra no momento. Sendo assim, este deve mandar uma mensagem ao microcontrolador. O PIC tratará de estabelecer a comunicação devida com o dado sensor, no caso o sonar, pegar este dado e reportá-lo ao computador. Sabendo da altitude atual, o computador fará um cálculo para descobrir a velocidade ideal dos motores para que o pouso possa acontecer. Tendo feito isso, ele enviará as novas referências de velocidade ao microcontrolador que por sua vez tratará de enviá-las ao controlador eletrônico de velocidade de cada motor.

Nesse simples exemplo, podemos perceber a importância do microcontrolador como intermediador do computador e dos sensores e atuadores. Ele pode ser vista na figura 3.4.1.

### 3.4.2 O Acionamento dos Motores

Com relação aos motores para o helicóptero, foi escolhido o uso dos motores elétricos já que estes dependerem somente da tensão de entrada para que se determinar velocidade de giro do rotor e sendo assim, são mais facilmente controlados do que os motores a combustão por exemplo. Além disso, o transporte de combustível necessário aos motores a combustão também dificultaria o controle de estabilização, já que o fluido combustível se deslocaria a medida que o veículo se inclinasse em determinada manobra para mudar sua posição. Tal deslocamento iria produzir um torque indesejado no robô e aumentaria ainda mais a complexidade do controle em questão.

Foram escolhidos para o protótipo inicial motores comuns de corrente contínua. Entretanto, para o protótipo final motores elétricos de corrente alternada, optou-se pelos do tipo **Brush-**



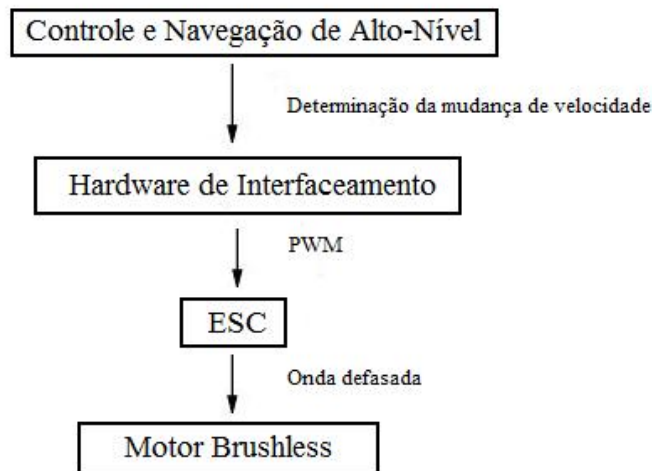


Figura 3.10: Esquema de comunicação entre o computador e os motores.

less(Sem escovas). Um exemplo desse motor pode ser visto na figura 3.11. Estes oferecem inúmeras vantagens em relação aos seus concorrentes, os motores de corrente contínua com escovas (**Brushed**), tais quais confiabilidade mais elevada, ruído reduzido, uma vida útil mais longa, etc. Tais dispositivos necessitam de um acionamento um pouco diferenciado de um motor elétrico de corrente contínua padrão. São necessárias três ondas defasadas em 120 graus para acionar esse tipo de motor. Esse sinal é gerado muitas vezes por um dispositivo microcontrolado. Uma das aplicações onde esses motores Brushless são comumente encontrados é no Aeromodelismo Profissional. É justamente de lá que iremos buscar tais dispositivos já que possuem uma potência similar a da aplicação requerida, em conjunto com massa e dimensão também ideal ao projeto além de serem acessíveis do ponto de vista econômico. A cada motor Brushless encontrado no mercado, temos já pronta essa interface microcontrolada que gera as tais ondas defasadas que irão acionar o motor. Elas são chamadas de Controladores Eletrônicos de Velocidades(*Electronic Speed Control* - **ESC**). Um esquema dessa comunicação pode ser visto na figura 3.4.2.

Tendo esses tipos de componentes definidos, são necessárias baterias para alimentação do sistema mecânico. Foram escolhidas as baterias de Polímeros de Lítio, as chamadas **LiPo**, muito usadas também no Aeromodelismo Profissional. São leves, seguras e apresentam uma densidade energética relativamente boa em comparação as outras encontradas no mercado atualmente.

Foi desenvolvido um firmware de controle que permite a geração de uma onda quadrada modulada na largura do pulso, **PWM**. Essa funcionalidade foi utilizada no primeiro protótipo que possuía motores elétricos DC comuns. Os controladores eletrônicos de velocidades, usados nos motores da versão final do projeto, tem uma variação desse tipo de onda como entrada para a geração das ondas defasadas para o controle do motor **Brushless**. Num PWM padrão, quanto maior a largura do pulso, maior será a velocidade de giro do motor, que estará submetido consequentemente a uma maior tensão que pode ser calculada pelo nível DC da onda em questão. Um exemplo disso pode ser melhor visualizado na figura 2.3.1.

Como já foi dito na Revisão Bibliográfica desde trabalho, o PWM usado na entrada do ESC tem uma variante do formato de onda encontrado no acionamento de motores de corrente contínua, um pseudo-pwm. O acionamento é o mesmo de um servo-motor. Isso porque esses controladores eletrônicos de velocidade são muito usado no ramo do aeromodelismo. E sendo assim, são conectados diretamente a um dos canais do receptor de rádio.

O formato da onda do pseudo-pwm pode ser visto na figura 2.7.



Figura 3.11: Exemplo de um motor Brushless. Este é um exemplar para aeromodelismo profissional produzido pela empresa Alemã Hacker.

O dispositivo escolhido para gerar esse tipo de onda foi o próprio microcontrolador responsável pela comunicação USB. Por serem muito usados em projetos de sistemas robóticos, tais dispositivos já possuem um hardware específico que implementa a funcionalidade do PWM. Sendo que, ele limita-se a geração de apenas duas ondas distintas. Como o quadrotor possui 4 motores e o hexarotor 6 motores e eles deverão girar em velocidades independentes, seriam necessários mais de um microcontrolador para realizar esse controle.

Para que esses microcontroladores possam funcionar corretamente, é necessário o uso de uma eletrônica auxiliar que trabalha em conjunto com esses chips. O uso de dois microcontroladores requisita também o dobro em eletrônica para o funcionamento do mesmo. Sendo assim, como a massa total do projeto é um requisito crítico, decidiu-se reduzir ao máximo a massa da eletrônica embarcada. Logo, pensou-se em emular as demais ondas de pwm via software no microcontrolador. Viu-se que seria mais fácil o controle se emulássemos não apenas algumas mas todas elas.

Construímos uma mesa de testes afim de averiguar o empuxo gerado por aquele sistema além da autonomia gerada pelas baterias. Uma foto desse sistema pode ser visto na figura 3.14.

A partir de diversos testes realizados, conseguimos verificar que cada um dos conjuntos motor-bateria-hélice rendeu um empuxo máximo de até 2,3Kg. Até mais um pouco do que o que foi previsto no projeto. Além disso, verificamos que sob uma carga de 1,2 a 1,5kg, que equivale aproximadamente a quanto cada motor do helicóptero precisará erguer, temos uma autonomia de até 13 minutos de voo.

### 3.4.3 GPS - Global Positioning System

Sistemas de posicionamento global (GPS), por possuírem uma margem de erro de 5 a 10 metros, são úteis quando temos tarefas de navegação de longa distância, mas não são uma boa alternativa em tarefas como por exemplo circular a 1 metro de distância de um tanque reservatório. Para esse último caso, a visão, obtida à partir das câmeras, é particularmente útil para alcançar esse objetivo.

O robô foi dotado dessa funcionalidade através de um chip, GT320RW da Unitraq, que pode ser visto no canto direito da protoboard da figura 3.4.1. O chip é conectado a um outro circuito integrado que converte para a tensão adequada de trabalho do microcontrolador. Este recebe as informações através de sua porta serial. Os dados vêm codificados segundo o protocolo NMEA(National Marines Eletronics Association). As informações pertinentes são devidamente separadas e repassadas ao computador mestre, conectado via USB a placa.

Os dados recebidos via satélite vem codificados num protocolo desenvolvido pela Marinha dos Estados Unidos da América. O chamado NMEA (National Marine Electronics Association) é na verdade uma especificação elétrica e de dados para comunicação entre dispositivos militares tais como sonares submarinos, pilotos automáticos, GPS, etc.

Trata-se basicamente de uma forma de passar uma única informação a diversos receptores. Isso se dá de uma forma serial via caracteres ASCII. Um pacote contendo diversos caracteres ASCII contém inúmeros campos com informações como por exemplo para o GPS, latitude, longitude, altitude, número de satélites em comunicação, etc.

Experimentos foram realizados para se confirmar os dados fornecidos pelo fabricante dos chips: O microcontrolador, que se comunicava com o chip GT320RW era responsável pela aquisição dos dados através de sua interface serial. Esse montante de dados era encaminhado a um laptop que os armazenou em um arquivo através de um programa em C++. Uma caminhada foi realizada em torno do departamento de Engenharia de Computação e Automação. Os resultados obtidos foram satisfatórios e correspondem as expectativas. Um erro de 3 a 4 metros pode ser encontrado na medição do GPS. O microcontrolador informava a cada segundo a sua posição ao computador via USB. Nele um programa tratava de guardar esses dados num arquivo. Após terminado o experimento o trajeto foi pintado numa tela do Google Maps.

Se vê claramente um exemplo do tipo de erro que esse sensor apresenta. Na figura 3.12 se pode ver um desvio na trajetória no canto superior esquerdo. Entretanto, como já foi dito, o GPS servirá somente sobre uma ótima de macro-navegação, onde as distâncias desejadas são relativamente grandes. Para navegar em distâncias menores utilizaremos a visão computacional como principal fonte de dados.



Figura 3.12: Experimento realizado com o objetivo de validar o funcionamento do GPS. Se nota um erro no canto superior esquerdo.

### 3.4.4 Sonar

Utilizamos do chip 6500 Series Ranging Modules da empresa SensComp para implementar um sonar para o robô AEROPETRO. Ainda não sabemos ao certo onde iremos posicionar na estrutura do robô o sensor em questão, mas, de qualquer forma, ele já se encontra operacional e inclusive com a comunicação via USB também já implementada. Testes foram realizados e indicaram um alcance máximo de cerca de 6 metros de altura. Isso pode implicar que deveremos usar de outros sensores para navegar o robô em altitudes maiores do que esta. Talvez o sensor mais indicado seja um altímetro baseado na pressão atmosférica. Os testes realizados também mostram que em ambientes mais fechados o sonar recebe interferências muito facilmente, o que seria um problema para uma navegação indoor.

### 3.4.5 Bússola

Um chip contendo a bússola digital, CMPS03 - Robot Compass Module, foi também utilizado para dotar o robô de um senso de referência universal. O dispositivo utilizado fornece os dados através de um barramento serial chamado **I2C**(*Inter-Integrated Circuit*). Tal barramento funciona como um intermediário na comunicação entre dois microcontroladores. No caso, o microcontrolador encapsulado no chip da bússola e o microcontrolador PIC da Microchip.

### 3.4.6 O Link De Rádio

Para facilitar os testes com tal helicóptero, foi introduzido também ao microcontrolador de acionamento dos motores uma outra funcionalidade. Através de um chip transceiver foi implementado um link de rádio entre ele e um computador. Tal interface possibilitava a um usuário no computador pudesse variar independentemente as velocidades de um dos quatros motores.

A interface de controle do rádio pelo computador foi implementada utilizando um microcontrolador PIC que possui a interface USB implementada. Os dados provenientes do rádio são tratados e devidamente colocados em pacotes que são encaminhados a USB de um computador com sistema operacional Linux.

Da mesma forma se criou uma outra placa que trata de receber os pacotes do sistema operacional Linux via USB e encaminhá-los via rádio através do chip conectado ao microcontrolador.

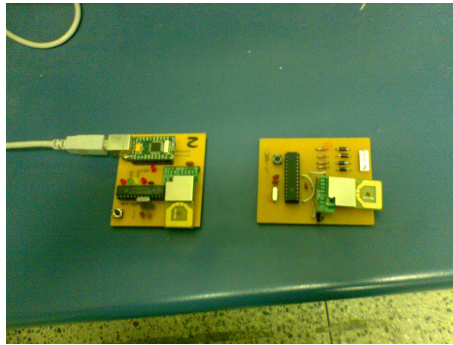


Figura 3.13: Placas de circuito impresso. A placa da esquerda contém o transmissor de rádio acoplado a um computador comum via USB. A da direita foi embarcada no helicóptero e é responsável pela recepção da mensagem de rádio, interpretação e geração das ondas de acionamento dos motores.

### 3.4.7 IMU - Unidade de medida Inercial

Foi utilizado o chip 6DOF v4 produzido pela empresa americana Sparkfun. Trata-se de uma unidade de medida inercial - IMU(Inertial Measurement Unit). Essa apresenta três tipos de sensores. Girômetros, acelerômetros e magnetômetros. Uma foto dessa unidade pode ser vista na figura 3.15.

O módulo possui três unidades de cada um desses tipos de sensores distribuídos em cada um dos eixos, X, Y e Z.

Cada um desses dados são disponibilizados ou via Bluetooth ou serialmente através de um conector presente no módulo. Através dessa última interface, capturamos o dado com ajuda do microcontrolador, que o encapsula num pacote que é encaminhado ao computador através da USB.

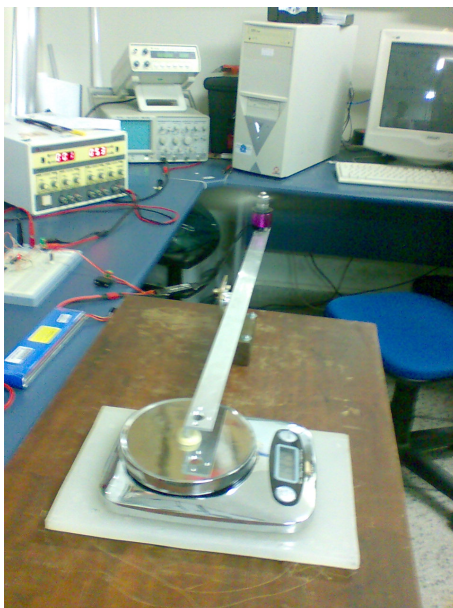


Figura 3.14: Mesa de ensaio de empuxo. Microcontrolador situado na *protoboard* envia o sinal do pseudo-pwm ao controlador eletrônico de velocidade que aciona os motores Brushless.

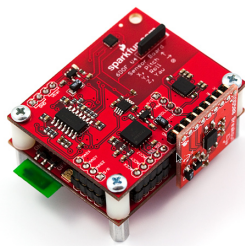


Figura 3.15: Unidade de Medida Inercial utilizado no robô, produzida e vendida pela empresa Sparkfun.

## 3.5 A Interface de Software

Como foi dito no início deste documento, o objetivo desse trabalho é de fornecer aos alunos doutorandos da equipe AEROPETRO ferramentas de hardware e software integradas ao robô para que essas possam aplicar na prática suas pesquisas.

Além de uma placa microcontrolada, que intermedia comunicação entre os sensores e atuadores do robô com o computador embarcado, foi desenvolvido também uma classe em C++ denominada Hexarotor que funciona acima dos elementos de comunicação.

Isso tem como objetivo abstrair todas as diferentes formas de comunicação e tecnologias para facilitar a programação do controle e da navegação do robô.

### 3.5.1 A classe C++

A seguir podemos ver os métodos mais importantes presentes na classe implementada:

**getUsbBuffer:** Método utilizado para retornar um atributo da classe que funciona como um buffer dos dados vindos do microcontrolador pela USB.

**set\_vel\_motor:** Modifica a referência da velocidade de um dos seis motores do veículo aéreo. Essa é enviada ao microcontrolador através da USB e por fim ao controlador eletrônico de velocidade(ESC). A função recebe dois parâmetros. O primeiro deles é número inteiro que pode variar de 1 a 6. Esse representa um dos 6 motores que vai haver a sua velocidade variada. O segundo parâmetro é outro número inteiro usado para informar a nova velocidade do motor. Esse número pode variar de no mínimo 3000, velocidade mínima, até no máximo 6000, velocidade máxima.

**enviar:** Envia um buffer de dados contendo 64 bytes ao microcontrolador permitindo a configuração de algum parâmetro pré-programado como por exemplo a alteração de todas as seis velocidades simultaneamente.

**getLongCompass:** Retorna ao usuário um número inteiro de 0 a 3599 representando uma orientação em graus (0° a 359,9°) em relação ao norte magnético fornecido pela bússola.

**getLatitude:** Retorna ao usuário um número inteiro indicando a latitude informada pelo GPS.

**getLongitude:** Retorna ao usuário um número inteiro indicando a longitude informada pelo GPS.

## Capítulo 4

# Resultados e Discussões

Conseguimos projetar um protótipo eficaz no que diz respeito ao atendimento dos requisitos que são necessários ao desenvolvimento dos algoritmos de controle e navegação do veículo aéreo. Além disso, todos os módulos da eletrônica que irá ser embarcada foram desenvolvidos, testados e integrados e com eles uma classe em C++ capaz de controlar o fluxo de dados no robô.

Na figura 4.1, podemos observar um esquema completo da arquitetura de hardware e software do robô desenvolvido.

Os dispositivos USB listados juntamente com a comunicação entre esses e os sensores e atuadores foram o principal foco do trabalho dado que a comunicação desenvolvida por Maranhão [7] entre a classe USBrobot e o sistema operacional já havia sido validada. Além disso a classe Hexarotor desenvolvida, trabalha uma camada acima da USBrobot e abstrai para aqueles que irão desenvolver os algoritmos de controle e navegação todas as funcionalidades e tecnologias distintas nas camadas abaixo.

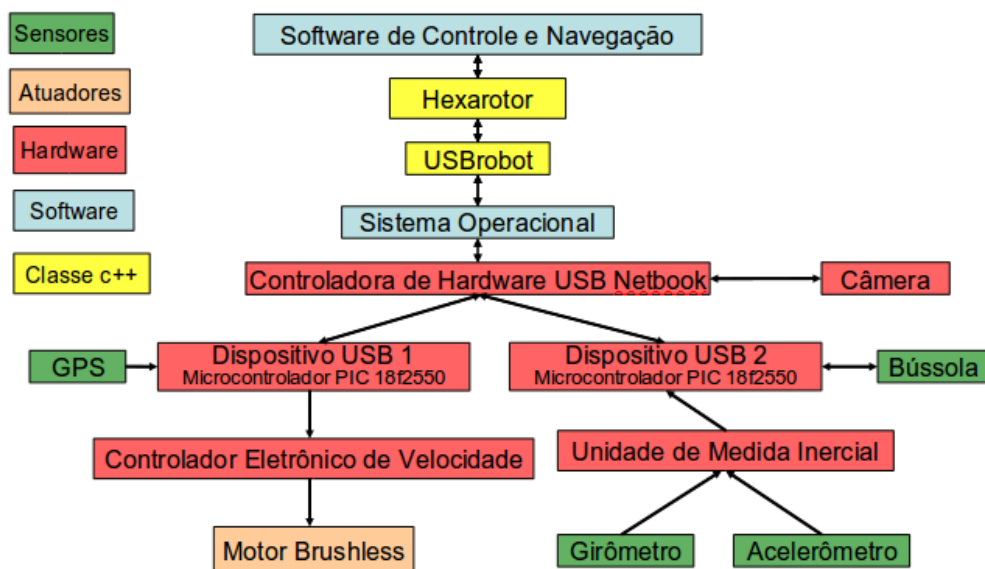


Figura 4.1: Arquitetura de Hardware e Software do Robô AEROPETRO.

## Capítulo 5

# Conclusões

O objetivo de um Trabalho de Conclusão de Curso de Engenharia de Computação é de integrar diversos conhecimentos teóricos adquiridos durante a formação. No decorrer deste trabalho foi possível aplicar conhecimentos de diversas disciplinas tais como: Conceitos e Técnicas de Programação, Estrutura de Dados, Circuitos Digitais, Arquitetura de Computadores, Sistema de Transmissão de Dados, Sistemas Operacionais, Sistemas Digitais e Eletrônica Industrial.

No presente texto apresentou-se detalhes de projeto e implementação de um veículo aéreo não tripulado que será usado em tarefas de navegação de áreas de risco, buscando a detecção de situações anômalas em plantas de petróleo e gás natural.

Diversas melhorias estruturais, de hardware e software podem ser implementadas visando o incremento do tempo de vôo do protótipo. Entretanto o objetivo principal desse trabalho que é o fornecimento de ferramentas de hardware e software para facilitar a tarefa de construção de algoritmos de controle e navegação que virão a ser desenvolvidos pelos outros integrantes do projeto AEROPETRO foi cumprido.



## Capítulo 6

# Referências Bibliográficas

1. Padmaraja Yedamal (2003), Brushless DC (BLDC) Motor Fundamentals, Microchip Technologies Inc.
2. Braga, Newton C. (2004), Eletrônica Básica para a Mecatrônica,
3. Microchip (2007), MPLAB C18 User Guide, Microchip Technologies.
4. Microchip (2007), MPLAB C18 Libraries, Microchip Technologies.
5. Microchip (2007), PIC 18F2455/2550/4455/4550 Data Sheet, Microchip Technologies.
6. Souza e Lavinia (2008), Conectando o PIC - Recursos Avançados
7. Maranhão, Djalma Teixeira Neto (2009), 'Projeto de uma Arquitetura de Hardware e Software para um Veículo Aéreo Não-Tripulado para Supervisão de Instalações de Petróleo', Trabalho de Conclusão de Curso, Engenharia de Computação, UFRN, Natal,RN.
8. Sanca, A.S., Guimarães, J.P.F., Cerqueira, J.J.F. and Alsina, P.J. (2010), 'Hexarotor Design Project: Modelling and Energy Management for Outdoor Aerial Robot', VI Congresso Nacional de Engenharia Mecânica - CONEM
9. Sanca, A.S., Alsina P.J., Cerqueira, J.J.F (2010), 'Hexarotor Micro-Aerial Vehicle: Modeling, Backstepping Control with Nonlinear Inputs and Sensor Fusion', XVII International Congress of Electronic, Electrical and Systems Engineering - Intercon
10. Skytrack (2009), Technical Data Sheet Version 1.2 GPS Mxodules, ME-1000RW
11. HTM (2003), CMPS03 - Compass Module Documentation
12. Sparkfun (2008), IMU 6 Degrees of Freedom v4 Datasheet, 6DOF v3

# Capítulo 7

## Anexos

### 7.1 Esquemáticos contendo os circuitos desenvolvidos

Nas figuras 7.1 e 7.2 temos os esquemáticos das placas desenvolvidas contendo as funcionalidades da bússola e IMU (figura 7.1) e o acionamento dos motores e do GPS (figura 7.2).

### 7.2 Código referente a classe C++ implementada

#### 7.2.1 hexarotor.h

```
#include "USBrobot.h"
class Hexarotor{

private:
    int vel_motores [6];
    int contador;
    int acelX;
    int acelY;
    int acelZ;
    int pitch;
    int roll;
    int yaw;
    int magX;
    int magY;
    int magZ;
    int longCompass;
    int shortCompass;

    USBrobot *handler; //Interface de comunicac~{a}o com a USB

    uint8_t buffer [64]; //Buffer

    //Metodos Privados
    int attachToInt (int i, int j, uint8_t *toAttach);
    int* decode ( uint8_t * dados );
    void refreshBuffer ();

    //Metodos Publicos
public:
    Hexarotor ();
    void set_vel_motor (int nmotor, int velocidade);
    void enviar (uint8_t *dados);
    void enviar ();
    int * getUsbBuffer ();
    void printBuffer ();
    int getShortCompass ();
    int getLongCompass ();
    int getLatitude ();
    int getLongitude ();
};
```

## 7.3 Lista de componentes

Componente	Quantidade	Detalhes	Local de Compra
Estrutura	1	Alumínio	
Motor Brushless	6	E-Max BL2815/09	<a href="http://www.asaseletricas.com.br">www.asaseletricas.com.br</a>
ESC	6	E-Max 50A	<a href="http://www.asaseletricas.com.br">www.asaseletricas.com.br</a>
Hélices	6	APC 12x6"Thin Electric Propeller	<a href="http://www.asaseletricas.com.br">www.asaseletricas.com.br</a>
Bateria	3	TP 4450mAh 14.8V 22C cont/40C Burst	<a href="http://www.asaseletricas.com.br">www.asaseletricas.com.br</a>
IMU	1	v6	<a href="http://www.sparkfun.com">www.sparkfun.com</a>
Bússola Digital	1	CMPS03 Digital Compass	<a href="http://www.tato.ind.br">www.tato.ind.br</a>
GPS	1	Unitraq GT320RW	<a href="http://www.tato.ind.br">www.tato.ind.br</a>

### 7.3.1 Placa Microcontrolada 1

Quantidade	Dispositivo	Partes	Descrição
1	CRYSTALHC49U-V	20MHZ	CRYSTAL
2	DE2L	-	DIODE
1	PINHD-1X4	CONECTOR-IMU	PIN HEADER
1	PINHD-1X9	CONECTOR-BUSSOLA	PIN HEADER
2	RMPC70-2	R1 R2	Resistor
1	10uF (V>12V) Eletrolítico	C3	CAPACITOR,
1	22pF	C2	CAPACITOR
1	22uF	C1	CAPACITOR
1	104nF	..	CAPACITOR
1	Conector Mini-USB	MINI-USB4P	MINI USB 4 Pol.
1	PIC18F255028DIP	PIC18F255028DIP	USB Microcontrollers

### 7.4 Placa Microcontrolada 2

Quantidade	Dispositivo	Partes	Descrição
1	CRYSTALHC49U-V	20MHZ	CRYSTAL
6	PINHD-1X3	CM1, CM2, CM3, CM4, CM5, CM6	PIN HEADER
4	1uF (50V) Eletrolítico	C4, C5, C6, C7	CAPACITOR
1	10uF (V>12V) Eletrolítico	C3	CAPACITOR
1	22pF	C2	CAPACITOR
1	22uF	C1	CAPACITOR
1	104nF	..	CAPACITOR
1	Conector Mini-USB	X1	MINI USB
1	Conector-GPS	PINHD-1X4	PIN HEADER
1	MAX232-Alimentado com 5V	MAX232	RS232 TRANSEIVER
1	PIC18F2550-28DIP	MICROCONTROLADOR1	USB Microcontrollers

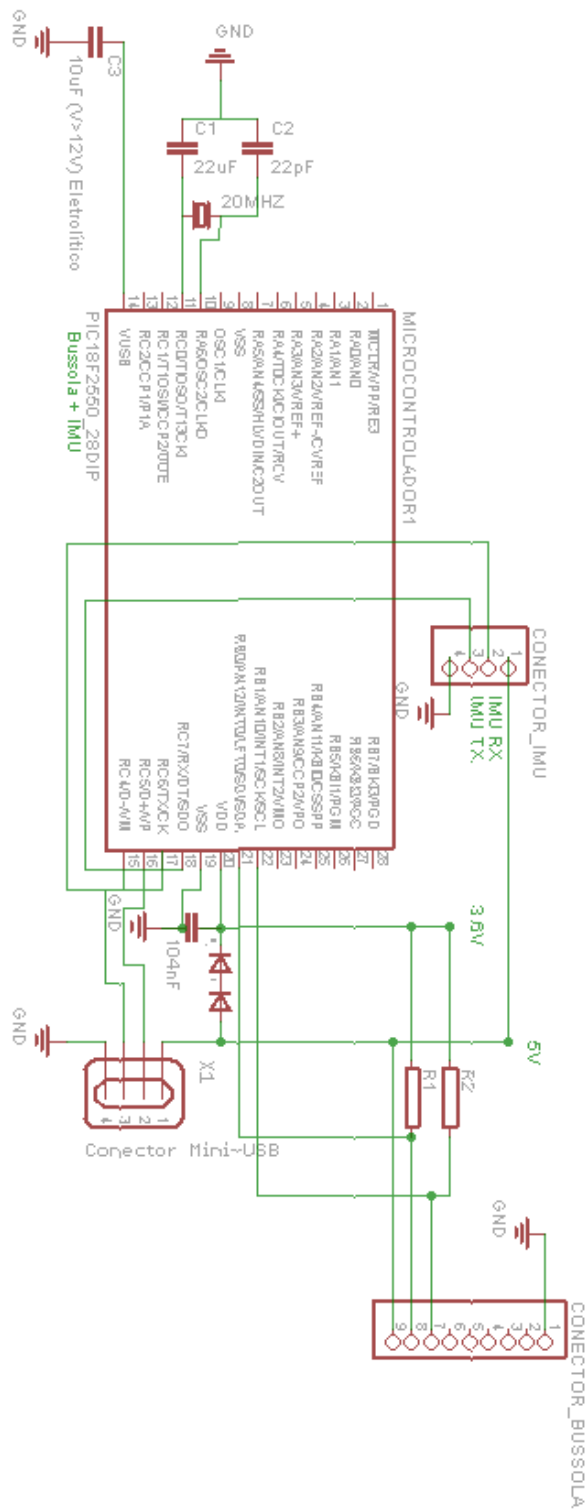


Figura 7.1: Circuito contendo a bussola e a IMU

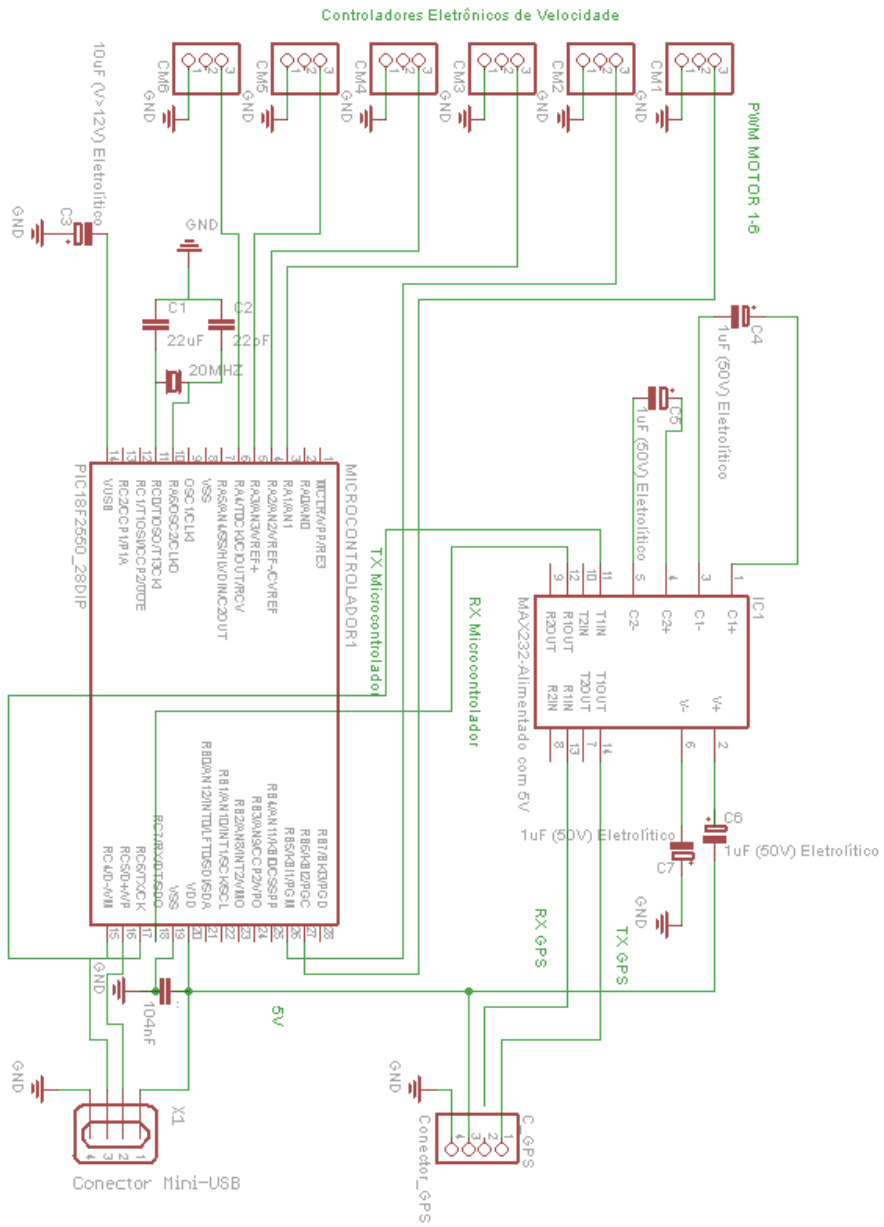


Figura 7.2: Circuito contendo o acionamento dos motores e o GPS