

Drone Stand-Alone



Caio Silva



UNIVERSIDADE AUTÓNOMA DE LISBOA

Engenharia de Informática

Departamento de Ciência e Tecnologia

Setembro de 2018

Drone Stand-Alone

Caio Silva

Orientado pelo Prof. Doutor Mário Marques da Silva



UNIVERSIDADE AUTÓNOMA DE LISBOA

Engenharia de Informática

Departamento de Ciência e Tecnologia

Setembro de 2018

Resumo

Nos últimos anos a tecnologia de Sistemas de Aeronaves Remotamente Pilotadas tem se mostrado progressivamente mais presente em iniciativas inovadoras para detecção, análises e monitoramento, tanto no campo militar como em áreas ligadas a agricultura entre outras.

A aeronave é constituída de um Raspberry PI e um controlador de voo PXFMini (Pixhawk Fire Cape Mini) para estabilizar a aeronave. Esta é feita pela Erle Robotics, uma shield de baixo custo para o Raspberry Pi que permite que seja criado um piloto automático com suporte para o ArduPilot. A shield foi projetada especialmente para o Raspberry Pi Zero, mas também é compatível com os outros modelos da família (3, 2B, 1B+ e Zero/W). Além do sistema de controle de voo, a aeronave dispõe de um módulo de telemetria 915 Mhz com capacidade de transmitir dados até 250 Kbps e com um alcance de cerca de 1600 metros, além de motores Brushless, ESC (Controlador de velocidade dos motores), Baterias, Power Module, GPS com Bússola e Câmera.

Será utilizado um software distribuição OpenSource (ArduPilot) para controle, este deve ser instalado e executado na estação base e ser capaz de comandar remotamente a aeronave, transmitindo novas rotas da navegação e missões. Essa investigação deve viabilizar a utilização de aeronaves remotamente controladas para realizar monitoramento e análises de lugares com difícil acesso e até fazer uma ampla cobertura do local com poucos recursos.

Palavras-chave: Autopilot, Raspberry Pi, PXFMini, ArduPilot, VANT, Mission Planner.

Abstract

In recent years the technology of Remotely Piloted Aircraft Systems has been progressively more present in innovative initiatives for detection, analysis and monitoring, both in the military field and in areas linked to agriculture, among others.

The aircraft consists of a Raspberry Pi and a PXFMini (Pixhawk Fire Cape Mini) flight controller to stabilize the aircraft. This is made by Erle Robotics, a low-cost shield for Raspberry Pi that allows you to create an autopilot with ArduPilot support. The shield was specially designed for Raspberry Pi Zero, but is also compatible with other family models (3, 2B, 1B + and Zero / W). In addition to the flight control system, the aircraft has a 915 Mhz telemetry module capable of transmitting data up to 250 Kbps and with a range of about 1600 meters, in addition to Brushless, ESC (Engine Speed Controller) motors, Batteries, Power Module, GPS with Compass and Camera.

An OpenSource distribution software (ArduPilot) will be used for control, it must be installed and run on the base station and be able to remotely command the aircraft, transmitting new navigation routes and missions. This investigation should enable the use of remotely controlled aircraft to perform monitoring and analysis of difficult-to-reach places and even provide ample coverage of the site with few resources.

Keywords: Autopilot, Raspberry Pi, PXFMini, ArduPilot, VANT, Mission Planner.

Lista de Imagens

Figura 1 - PXFMini Board	12
Figura 2 - PXFMini Specs.....	12
Figura 3 - ArduPilot Mega	13
Figura 4 - ArduPilot Mega Specs	13
Figura 5 - Mission Planner	16
Figura 6 - MAVLink Funcionamento.....	17
Figura 7 - Frame HexaCopter.....	18
Figura 8 - Frame QuadCopter.....	19
Figura 9 - RC Transmissor e Recetor	19
Figura 10 - Esquemático do Circuito eletrônico PPM Encoder	20
Figura 11 - VANT com GPS externo	21
Figura 12 - Carregador inteligente de Baterias LiPo.....	23
Figura 13 - Bateria LiPo 5000 mAh.....	23
Figura 14 - Bateria LiPo danificada	24
Figura 15 - Case bateria LiPo	26
Figura 16 - Radio de Telemetria.....	26
Figura 17 - ESC (Controlador eletrônico de velocidade).....	27
Figura 18 - BLHeli Interface	28
Figura 19 - ESC Protocolo	29
Figura 20 - Cartão de programação de ESC.....	30
Figura 21 - UBEC.....	30
Figura 22 – Motor Brushless	32
Figura 23 - Motor Brushed	33
Figura 24 - Motor Disco 3508	34
Figura 25 - Hélices Goupner 9x4” / 8mm.....	35
Figura 26 - Power Module (Conector XT60).....	36
Figura 27 - Kit FPV	37
Figura 28 - Módulo de medição HC-SR04.....	38
Figura 29 - Funcionamento do módulo de distância	38
Figura 30 - F450 com trem de aterragem	38
Figura 31 - EPM Gripper.....	39
Figura 32 - EPM Gripper.....	39

Figura 33 - Configuração do Joystick	40
Figura 34 - Diminuir atraso do Joystick	41
Figura 35 - PPM Encoder	44
Figura 36 - Arduíno AT328P	45
Figura 37 - Circuito impresso.....	45
Figura 38 – PPM Encoder HomeMade	46
Figura 39 - Suporte GPS.....	47
Figura 40 – Router 4G	49
Figura 41 – Motor Turnigy D3536/6 1250kv	50
Figura 42 - Teste de motor e ESC	51
Figura 43 - Bateria inchada	52
Figura 44 - HexaCopter montado	55
Figura 45 - F450 Montado / Nova VANT	56
Figura 46 – Parâmetros do "preFlight"	58
Figura 47 - Parâmetros de Pre-Arm.....	59
Figura 48 – Comando para “Armar” a VANT	59
Figura 49 – Drone alternativo (Grupo 4G).....	61
Figura 50 - Primeira missão autónoma.....	63

Lista de Tabelas

Tabela 1 - Material para fabricar PPM-Encoder "Home Made"	20
Tabela 2 – Comparação de ESC Linear e ESC de Comutação	32
Tabela 3 - Materiais disponíveis para o projeto	42
Tabela 4 - Materiais adquiridos	43
Tabela 5 - Diferenciação de frequência de banda por território	48

Sumário

1.	Introdução.....	9
1.1.	Objetivos.....	9
1.2.	Estrutura do relatório	10
2.	Estado do conhecimento.....	10
2.1.	Controladores de Voo (Hardware)	10
2.1.1.	PXFMini	11
2.1.2.	ArduPilot Mega 2.5 / 2.6	12
2.2.	Controladores de Voo (Firmware).....	14
2.2.1.	ArduPilot (APM)	14
2.2.2.	PX4	14
2.2.3.	Principal diferença entre APM e PX4	15
2.3.	Estação em solo	15
2.3.1.	Mission Planner	15
2.4.	MAVlink.....	16
2.4.1.	Aplicações do MAVlink.....	16
2.4.2.	Funcionamento MAVlink.....	17
2.5.	Construindo uma VANT	17
2.5.1.	Frames / Quadro	18
2.5.2.	RC Transmissor e Recetor de canais	19
2.5.3.	Placa controladora de voo	20
2.5.4.	Estação em Solo	21
2.5.5.	GPS + Compass.....	21
2.5.6.	Baterias de LiPo e carregador.....	21
2.5.7.	Radio de Telemetria	26
2.5.8.	ESC (Controlador eletrônico de velocidade).....	26
2.5.9.	BEC – Circuito eliminador de Bateria.....	30

2.5.10.	Motores e Hélices	32
2.5.11.	Power Module	35
2.6.	Módulos Opcionais.....	36
2.6.1.	FPV – First Person View	36
2.6.2.	Módulo de medição de distância	37
2.6.3.	Trens de aterragem	38
2.6.4.	EPM gripper	39
2.6.5.	Joystick / Game Pad	39
2.7.	ROS – Robot Operating System.....	41
3.	Execução	42
3.1.	Planeamento.....	42
3.2.	Materiais	42
3.2.1.	Materiais disponíveis.....	42
3.2.2.	Testes realizados nos materiais	44
3.2.3.	Montagem da VANT	54
3.3.	Testes de confiabilidade	57
3.3.1.	Testes de voo guiado	57
3.3.2.	Testes de voo autónomo	62
4.	Considerações finais.....	63
4.1.	Conclusões.....	63
4.2.	Contribuições.....	64
4.3.	Trabalhos Futuros	64
	Lista de Referências	65
	Anexos.....	67

1.Introdução

Veículos Aéreos Não-Tripulados (VANTs), popularmente conhecidos como drones ou também ARPs (Aeronaves Remotamente Pilotadas) são aeronaves capazes de voarem através do controle de um operador que não está a bordo. Comumente associados ao uso militar, tem-se visto um interesse crescente no desenvolvimento de VANTs para o uso doméstico, setores privados e recreacional. Isto se deve a diminuição de custo ao longo dos anos da tecnologia VANT e também ao fato destes possuírem vantagens funcionais sobre os veículos tripulados. VANTs podem pairar no ar, além de manobrar suavemente e precisamente através de pequenos espaços ou em conjunto com outros drones. Tudo isso enquanto carregando instrumentos como uma câmera de vídeo estabilizada e uma infinidade de outras tecnologias a bordo. A extensão de sua versatilidade é o que os torna uma opção viável para uma série de tarefas diferentes [16].

O voo autônomo, uma das principais características dessas aeronaves, permite uma redução de custos quando comparados aos veículos aéreos convencionais. [17]

Essa diminuição ocorre, principalmente, pela inexistência de piloto a bordo da aeronave. Entretanto, ainda há os gastos com todos os equipamentos e montagem. No caso de um VANT, este é dividido em: estrutura, motores, hélices, controladores de velocidade, controlador de voo, baterias, sensores e rádio. Por não necessitar de um piloto, essas aeronaves podem ser utilizadas em missões que seriam impossíveis ou muito arriscadas para serem executadas por um ser humano.

Tendo em vista todos os pontos já alinhados até agora, este trabalho tem como tese comprovar a possibilidade de utilizar uma VANT para realizar missões que são mais arriscadas para seres humanos. Além de diminuir o custo com essas missões, sendo elas de qualquer área aplicada, com as VANT's há possibilidade de uma única pessoa com mínimos recursos realizar análises de terrestres ou aéreas em grandes áreas, além de tabular os dados obtidos nas análises.

1.1. Objetivos

O objetivo principal desse projeto é investigar os métodos utilização do sistema de AutoPilot utilizado por VANT's, sendo controlado diretamente pelo software que executa a estação em solo, por meio de comunicação sem fio, sem a necessidade de um rádio controle. Também tendo como objetivo a montagem de uma VANT e inicialmente realizar voos com a auxílio do comando remoto. O projeto aborda, especificamente, o uso do software Mission

Planner, que constitui de uma Ground Station (estação de solo) pra fim de controle da VANT pelo software.

O Mission Planner é um software de código aberto e pode ser modificado para que futuramente possa ser atribuída sugestões de projetos e aplicadas sobre o mesmo. O software possui um repositório no site GitHub e atualmente tem a contribuição de diversos desenvolvedores ao redor do mundo.

1.2. Estrutura do relatório

O relatório do projeto esta dividido em 4 capítulos. Os próximos capítulos são, Estado do conhecimento, Execução das tarefas programadas, Considerações finais.

Estado do conhecimento são apresentadas características de componentes e sistemas relacionados a VANT, adquiridas durante o estudo da tecnologia.

Execução das tarefas programadas está relacionado com o planejamento, montagem e testes realizados na VANT durante o período do projeto.

Considerações finais apresenta a conclusão científica do projeto, apresentando simplesmente a conclusão do objetivo e apresentando exemplos de trabalhos futuros que podem ser realizados através da ajuda deste relatório.

2.Estado do conhecimento

2.1. Controladores de Voo (Hardware)

Os controladores de voo são o principal componente dos drones, tem a função de processar os sinais de entrada e gerar saídas adequadas, e em alguns momentos pode servir apenas de extensão entre as interfaces I/O sem realização de processamento.

As controladoras variam muito de especificação entre uma diversidade enorme de marcas e modelos, algumas mais baratas apresentam apenas as funções básicas necessárias ao voo, como o mapeamento dos canais do controle do rádio em informações para mudança de velocidade, direção e altitude. Essa é a função mais básica da controladora, mapear os movimentos do controle em novas rotações que são definidas dinamicamente para os motores através do sinal enviado pelo ESC (controlador eletrônico de velocidade, necessário em motores elétricos sem escovas (brushless)). Para a realização desse mapeamento o

giroscópio não é necessário (sem o giroscópio o drone não estabiliza automaticamente em X e Y), uma controladora operando dessa forma estaria no modo ACROBAT, o mais difícil de pilotagem.

As controladoras de voo de melhor qualidade têm a capacidade de processar dados de sensores auxiliares embutidos em seu próprio circuito ou externos, como o GPS por exemplo, nesse caso, não são somente os comandos do controle que ditarão a velocidade de cada motor, pois a controladora vai conseguir estabilizar o drone baseada em informações dos sensores inerciais e GPS, dessa maneira, a pilotagem se torna muito mais fácil.

As controladoras podem ser divididas primariamente entre código aberto e código fechado, as de código aberto, são desenvolvidas pela comunidade e permitem alterações nos algoritmos que controlam o hardware, as de código fechado não permitem alterações, mas algumas tem APIs que permitem estender funcionalidades.

No projeto serão mostrados apenas dois dos diversos controladores de voo que existem hoje no mercado, esses são: PXFMini e ArduPilot Mega, ambos são controladores de código aberto.

2.1.1. PXFMini

O PXFmini pesa apenas 15 gramas e incorpora toda a eletrônica de potência necessária para cumprir a maioria dos componentes existentes para drones através de suas portas I2C e UART. Inclui 3 eixos de sensor de gravidade, 3 eixos de giroscópio, 3 eixos de bússola digital, sensor de pressão, sensor de temperatura. Inclui novos conectores JST GH e possui canais de saída servo 8PWM e uma entrada PPMSUM.

O PXFMini trabalha com código aberto, fornecendo suporte às plataformas padrões (APM e PX4) utilizadas pelos drones.



Figura 1 - PXFMini Board

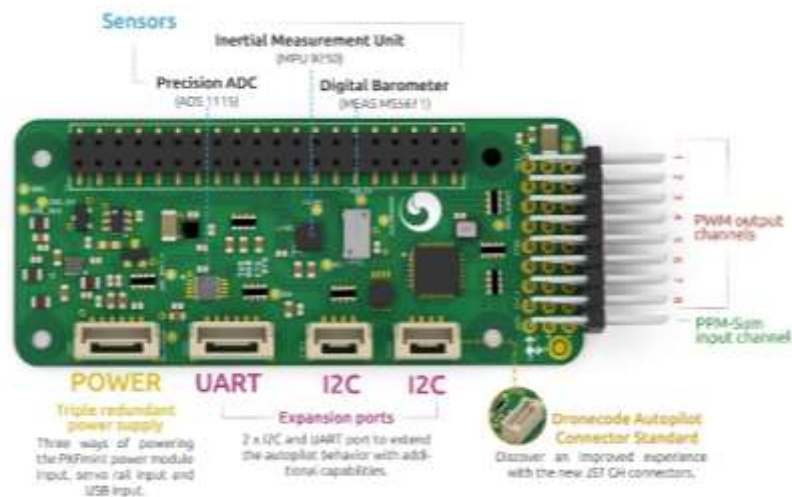


Figura 2 - PXFMini Specs

2.1.2. ArduPilot Mega 2.5 / 2.6

O ArduPilot Mega (APM) é um piloto automático IMU de qualidade profissional baseado na plataforma Arduino Mega. Este piloto automático pode controlar aeronaves de asa fixa, helicópteros multirrotor, bem como helicópteros tradicionais. É um piloto automático completo com capacidade de estabilização autônoma, navegação baseada em waypoints e telemetria bidirecional com módulos sem fio Xbee. Suportando 8 canais RC com 4 portas seriais.

A placa APM 2.5/2.6 não requer montagem e está pronta para o firmware. Você tem a opção de configuração de pino de entrada lateral ou superior, para acomodar uma variedade de instalações.



Figura 3 - ArduPilot Mega

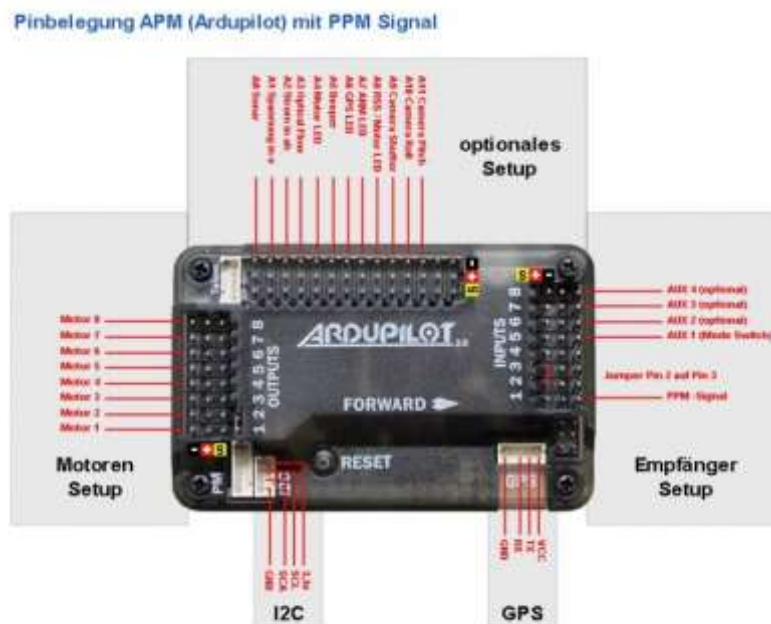


Figura 4 - ArduPilot Mega Specs

2.2. Controladores de Voo (Firmware)

Existem vários projetos de controladores de voo. Cada um geralmente se concentra em algum hardware, tecnologia ou objetivos específicos. Este relatório lista alguns dos principais firmwares disponíveis, ArduPilot, PX4 Flight Stack, iNav, LibrePilot, dRonin, MutiWii.

Este relatório destaca apenas dois deles.

2.2.1. ArduPilot (APM)

O ArduPilot visa permitir a criação e o uso de sistemas de veículos não tripulados confiáveis e autônomos para o benefício pacífico de todos. O Ardupilot é um projeto que atualmente pode ser descrito como uma suíte de piloto automático. Como um projeto de código aberto, está constantemente evoluindo e sendo desenvolvido. A equipe de desenvolvimento usa a comunidade e parceiros comerciais para adicionar funcionalidades ao ardupilot que beneficiam a todos. Embora o Ardupilot não fabrique nenhum hardware, o firmware Ardupilot funciona em muitas placas diferentes (hardware) para controlar veículos não tripulados de todos os tipos. Juntamente com o software de controle de solo, os veículos não tripulados que utilizam o Ardupilot podem ter funcionalidades avançadas, incluindo comunicação em tempo real com os operadores [1]

O Ardupilot é provavelmente o firmware de piloto automático de drone mais popular. O firmware pode controlar praticamente qualquer veículo, aviões de asa fixa, quadricópteros e tudo entre incluir híbridos, veículos terrestres e barcos. O firmware é dividido em três ramos: Ardupilot para aeronaves de asa fixa, Arducopter para multirotores e helicópteros e Ardurover para veículos terrestres. O firmware possui um repositório localizado no site GitHub e pode ser acessado por qualquer pessoa comum ou desenvolvedor que queira arriscar-se a produzir novas funcionalidades.

2.2.2. PX4

O PX4 é outro piloto automático poderoso e completo. Este suporta vários tipos de veículos, também tem uma comunidade de desenvolvedores ativa. A plataforma mais famosa que comanda a pilha de voo do PX4 são alguns dos drones da Yuneec.

Você pode considerar a pilha de voo do PX4 como sendo semelhante ao ardupilot, pois há um monte de hardware que pode rodar o firmware do PX4 ou do Ardupilot. Da mesma forma, como ambos os sistemas usam o mesmo protocolo de telemetria (MAVlink), um monte

de software também é compatível com QGroundControl. É somente quando você chega a funções muito específicas que você pode notar algumas variações entre os dois projetos. No entanto, em termos de funções autônomas centrais PX4 e Ardupilot são muito semelhantes.

2.2.3. Principal diferença entre APM e PX4

A diferenciação chave é com a licença de software. O PX4 usa o BSD enquanto o ArduPilot usa o GPL. Isso torna o PX4 mais atraente para uso comercial, já que você não precisa fazer modificações no código aberto. Com o ArduPilot, quaisquer alterações precisam ser compartilhadas como código aberto. Portanto, se você planeja desenvolver alguma função especial que deseja manter privada ou vender, o PX4 é o caminho a percorrer. É por isso que o PX4 é mais atraente para uso comercial.

2.3. Estação em solo

Uma estação em solo é tipicamente um aplicativo de software, executado em um computador, que se comunica com o VANT por meio de telemetria sem fio. Ele exibe dados em tempo real sobre o desempenho e posição das VANT e pode servir como um "cockpit virtual", mostrando muitos dos mesmos instrumentos que você teria se estivesse pilotando um avião real. Um Ground Control Station (GCS) também pode ser usado para controlar em vôo, fazer o upload de novos comandos de missão e definir parâmetros. Muitas vezes também é usado para monitorar os fluxos de vídeo ao vivo das câmeras de um VANT.

Existem pelo menos dez diferentes estações de controle terrestre. Há (Mission Planner, APM Planner 2, MAVProxy, QGroundControl. Para Tablet / Smartphone há (DroidPlanner 3), MAVPilot, AndroPilot que pode ser usado para se comunicar com ArduPilot.

2.3.1. Mission Planner

O Mission Planner é uma estação de controle em solo para o Plane, Copter e Rover. É compatível com o Windows e pode ser emulado em Mac OS X usando o Mono. O Mission Planner pode ser usado como um utilitário de configuração ou como um suplemento de controle dinâmico para seu veículo autônomo.

O Mission Planner possui uma licença OpenSource (GPLv3), pode ser feito o download do código fonte em um repositório do GitHub.



Figura 5 - Mission Planner

O Mission Planner permite que seja carregado o firmware que deseja utilizar em sua board de controle de voo, permite configurar e ajustar a sua VANT para um melhor desempenho. Possibilita que planeie, salve e carregue as missões autônomas na board. Faz download e análise dos dados da missão executada. Possui uma interface com um simulador de voo, permitindo criar uma VANT virtual completa. Com a comunicação sem fios, permite capturar dados de telemetria da VANT enquanto estiver em operação além de fornecer os dados para análise futura. Ainda permite que opere a VANT em PFV (visão em primeira pessoa).

2.4. MAVlink

O MAVLink (Micro Air Vehicle Link) é um protocolo de mensagens muito leve para comunicação com drones (e entre componentes onboard drone). O MAVLink segue um padrão moderno de design de assinatura e ponto-a-ponto híbrido: Os fluxos de dados são enviados como tópicos enquanto os subprotocolos de configuração, como o protocolo da missão ou o protocolo de parâmetros, são ponto a ponto com retransmissão [12]

2.4.1. Aplicações do MAVlink

Ele é usado principalmente para comunicação entre uma estação de controle de solo (GCS) e veículos não tripulados e na intercomunicação do subsistema do veículo. Pode ser usado para transmitir a orientação do veículo, sua localização GPS e velocidade.

Muitos pilotos automáticos, estações em solo, APIs de integração, projetos e outros pacotes de software usam o MAVLink.

2.4.2. Funcionamento MAVlink

Os comandos MAVLink são enviados por comunicação serial, cada mensagem carrega 17 bytes de informação, sendo 6 bytes de header, aonde você enviará algumas informações obrigatórias, como número de sequência, ID de identificação do sistema que está enviando, ID do componente que está enviando, tamanho da mensagem e código de validação (checksum). Ao enviar uma mensagem você pode alterar o modo de voo, enviar um comando para o drone pousar, voltar para o ponto inicial, waypoints e de retorno do drone você recebe todas as informações de telemetria que estão disponíveis e exibe na estação em solo, como o Mission Planner ou o QGroundControl, tais como altura, inclinação, número de satélites, coordenadas, voltagem da bateria, velocidade, entre outras informações [13]

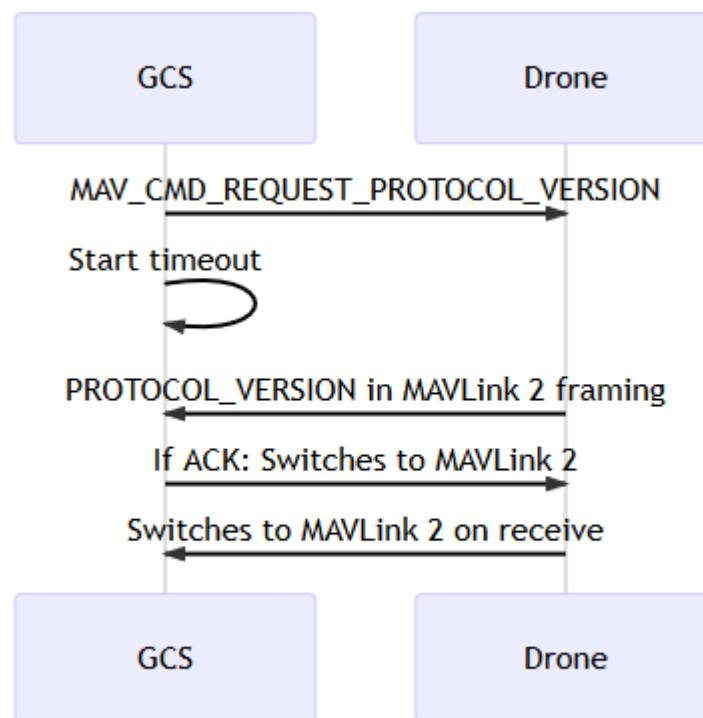


Figura 6 - MAVLink Funcionamento

2.5. Construindo uma VANT

Existem inúmeros quadros (frames), ESCs (Controlador eletrônico de velocidade) e motores disponíveis.

2.5.1. Frames / Quadro

A estrutura que contém todos os componentes, precisa ser projetado para ser forte, mas também leve. Esta estrutura, onde estão todos os componentes eletrônicos, podem ser feitos de diversos materiais, desde madeira, a alumínio e Carbono.

Fibra de Carbono - A fibra de carbono é o material mais utilizado, é leve e tem uma grande resistência, mais rígida e absorve as vibrações dos rotores, tem igualmente a propriedade de bloquear os sinais de Radio Frequência.

Alumínio - Placas ocas de alumínio é o mais popular para os braços dos QuadCopters devido ao seu peso relativamente leve, rigidez e acessibilidade. Em caso de problema de vibração intensa, pode atrapalhar as leituras dos sensores.



Figura 7 - Frame HexaCopter

Plástico - O plástico deve ser usado com precaução, não deve ser usado para os braços devido à sua flexão, pois fará os motores mudar o ângulo de rotação e isso pode originar leituras erradas do Controlador de Voo.



Figura 8 - Frame QuadCopter

Os frames podem ser configurados com diversos tipos, **Tricopter**, **Y4**, **QuadCopter**, **HexaCopter**, **Y6**, **OctaCopter**, **X8**.

PCB ou PDB (Descrever)

2.5.2. RC Transmissor e Recetor de canais

Você precisará de um transmissor de controle de rádio para controlar manualmente seu helicóptero e ativar seus modos de voo. Você pode usar qualquer sistema de transmissor e receptor RC.



Figura 9 - RC Transmissor e Recetor

PPM Encoder

O codificador PPM permite codificar até 8 sinais PWM (pulso modulado por largura) em um sinal PPM (modulação de posição de pulso).

O PPM Encoder pode ser comprado em lojas especializadas ou pode ser produzido através do Arduino ATmega328p ou através do esquemático de circuito eletrônicos mostrado abaixo.

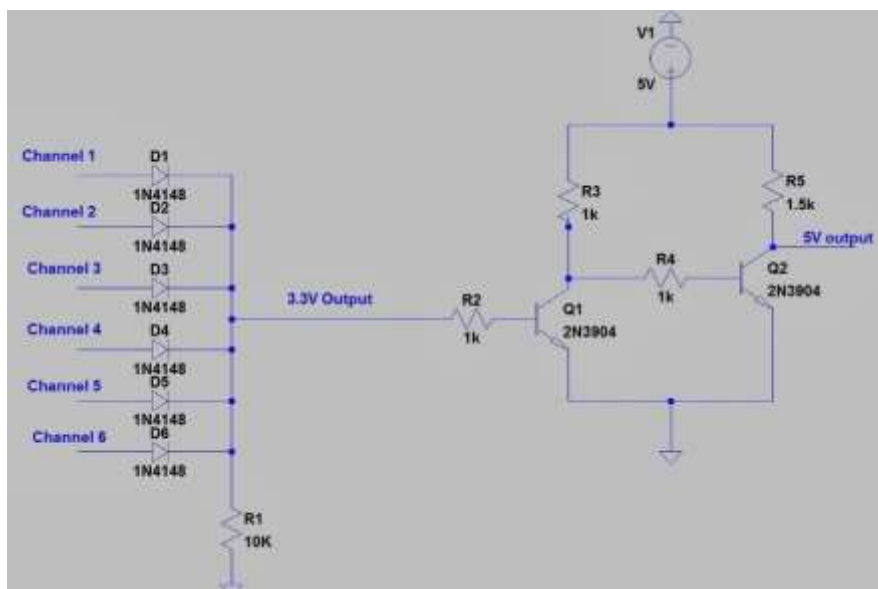


Figura 10 - Esquemático do Circuito eletrônico PPM Encoder

Tabela 1 - Material para fabricar PPM-Encoder "Home Made"

Item	Descrição	Quantidade
1	Placa de circuitos - Protoboard	1 (und)
2	Pino macho (Male Header)	9 (und)
3	Diodo 1N4148	6 (und)
4	Transistor 2N3904 NPN	2 (und)
5	Resistor 1k Ohm	3 (und)
6	Resistor 10k Ohm	1 (und)
7	Resistor 1,5k Ohm	1 (und)

2.5.3. Placa controladora de voo

A placa de piloto automático da VANT que determina sua capacidade de voo.

Conceitos já abordados no tópico 2.1 deste relatório.

2.5.4. Estação em Solo

O Mission Planner é necessário para o carregamento de novas versões do helicóptero no controlador de voo e para ajuste e calibração de primeiro voo. Ele é executado em um PC e também pode ser usado para planejar missões.

Conceitos já abordados no tópico 2.3 deste relatório.

2.5.5. GPS + Compass

O módulo (UBlox GPS + Compass) é o GPS mais usado para controladores de voo compatíveis com ArduPilot firmware.

O ArduPilot configura automaticamente o GPS logo após a inicialização, portanto não há necessidade de nenhuma calibração relacionada ao GPS. No entanto a bússola deve ser calibrada.

Este módulo permite que o GPS seja montado separadamente do módulo de controle de voo para que ele possa ter a melhor visibilidade do céu e permitir que a bússola fique distante dos campos magnéticos interferentes.



Figura 11 - VANT com GPS externo

2.5.6. Baterias de LiPo e carregador

A VANT requer uma bateria de Polímero de Lítio recarregável (LiPo). Uma boa regra é usar 1.000 mAH (miliampere-horas) por motor. Para um helicóptero Quad, uma LiPo de 4.000 mAh.

As baterias de LiPo possuem uma placa de circuito e são quase 100% inteligentes. De acordo com o fabricante das baterias, elas aguentam até 200 ciclos de carga, ou seja, você carregou e usou 200 vezes, porém quando uma célula danifica essa bateria não fica totalmente inteligente, perdendo a capacidade de se auto conservar.

Quando usa um carregador inteligente de baterias, o mesmo informa se existe alguma das células com defeito. É recomendado que não seja utilizado baterias com células danificadas para voos, podendo ocasionar queda da VANT.

Um carregador inteligente de baterias tem algumas características que ajudam a proteger a vida útil da bateria.

1 – Monitora o tempo máximo e normal de carga da bateria. Quando o tempo máximo é atingido o carregador encerra a carga como medida de segurança;

2 – Monitora a carga de cada célula individualmente;

3 – Oscila a amperagem de carga durante o ciclo, especialmente ao término, quando a bateria atinge cerca de 80 a 90% de sua capacidade, a fim de manter a voltagem constante (CV);

4 – Não inicia a carga se a voltagem da bateria não for superior a voltagem mínima (3V). Quando é esse o caso, significa que uma ou mais células foi danificada, pois baterias LiPo, diferentes de outros tipos de baterias, não podem ter sua carga zerada ou reduzida abaixo de um limite mínimo que é de 3.30v por célula;

5 – Possui funções de armazenamento, carga, descarga e balanceamento de carga;

6 – Emite sinais sonoros que orientarão o processo de carga;

7 – Possui display para monitoração em tempo real;

8 – Possui mecanismos de prevenção contra curto-circuitos;

Além de possuir diversas outras funções relacionadas a segurança, já que tais baterias exigem um maior cuidado.



Figura 12 - Carregador inteligente de Baterias LiPo

Devem ser tomadas algumas medidas de segurança com as baterias de LiPo.

Quando carregada, descarregada, armazenada, manuseada incorretamente, as baterias de LiPo podem se tornar muito perigosas.



Figura 13 - Bateria LiPo 5000 mAh

O site **DOCTORDRONE** listou 15 dicas de segurança para com as baterias de LiPo.

1. Nunca carregue, descarregue, use ou armazene uma bateria LiPo danificada ou inchada. Siga imediatamente protocolos de descarte adequados.
2. Evite a compra de baterias LiPo usadas.

3. Use sempre um carregador de LiPo com balanceador ao carregar e descarregar a sua LiPo. É essencial que todas as células de uma bateria LiPo mantenham a mesma voltagem através de todas as células em todos os momentos. Se as tensões entre as células se diferenciarem muito uma da outra ($5\text{ mV} \sim 10\text{ mV}$), a bateria pode se tornar instável e perigosa. (A menos que seja uma única célula LiPo, caso em que você não precisa se preocupar com o equilíbrio da célula).

4. Utilize sempre um saco de segurança LiPo à prova de fogo ou algum outro recipiente à prova de fogo enquanto você está carregando, descarregando ou armazenando sua bateria LiPo. Nunca sobrecarregue sua bateria, siga sempre as recomendações do fabricante sobre a sua LiPo, sobre quantos mAh ela pode ser carregada com segurança.

5. Não utilize case de mão para o armazenamento permanente da LiPo. A espuma e o plástico, nestes casos, podem ajudar a espalhar o fogo da LiPo em caso de problemas. Sempre use um recipiente à prova de fogo.

6. Nunca deixe suas baterias LiPo carregando sozinhas. Se a bateria começa a ficar inchada, fumaçar ou pega fogo você precisa ser capaz de lidar com a situação imediatamente. Apenas 5 minutos de descuido podem significar um desastre.



Figura 14 - Bateria LiPo danificada

7. O fogo da LiPo é um fogo químico. Mantenha sempre um extintor de incêndio Classe D nas proximidades das suas baterias. O local aonde você realiza a carga e descarga deve estar livre de quaisquer materiais que possam pegar fogo, como mesas de madeira, carpete, ou recipientes com gasolina. A superfície ideal para carregar e armazenar baterias LiPo é concreto ou cerâmica.

8. Nunca sobrecarregue uma bateria LiPo. Normalmente uma carga completa é 4.2v por célula.

9. Nunca descarregue uma bateria LiPo abaixo 3.0v por célula. O ideal é que você nunca deixe ir abaixo de 3.2V por célula para manter uma bateria saudável. 2.9V por célula e menos está causando danos permanentes.

10. Nunca deixe suas baterias LiPo paradas com uma carga completa por mais de 2 ou 3 dias. Se pelo 3º dia você percebe que não vai usar a sua bateria, você precisa de descarregar a bateria até 3.6V-3.8V por célula para o armazenamento seguro até que você esteja pronto para voar com seu drone novamente.

11. Guarde sempre as suas baterias LiPo em temperatura ambiente. Não armazene em um lugar muito quente, ou muito frio.

12. Lembre-se sempre que o calor é o inimigo das baterias de LiPo. Quanto mais quente suas baterias ficarem, menor sua vida útil será. Nunca carregue uma bateria que ainda está quente de uso, e nunca use uma bateria que ainda está quente.

13. Dependendo de como elas são usados, a maioria das baterias LiPo normalmente não duram mais do que 300 ciclos de carga. Deixá-las com uma carga completa ou esgotadas o tempo todo, deixar acabar completamente a carga ou expô-las a altas temperaturas irá encurtar este tempo de vida drasticamente.

14. Baterias LiPo não funcionam bem em clima frio. Quanto mais frio está, menor seu tempo de execução será, devido à desaceleração da atividade química dentro da bateria. Se estiver abaixo 14F (-10C), o uso de LiPo não é recomendado a todos. Sua bateria pode causar a falha repentina do seu drone, sem aviso, nestas temperaturas.

15. Sempre embale suas baterias LiPo em sua bagagem de mão e nunca em sua bagagem despachada ao viajar em um avião.



Figura 15 - Case bateria LiPo

2.5.7. Radio de Telemetria

Um rádio de telemetria permite que a VANT se comunique com a sua estação terrestre a partir do ar usando o protocolo MAVLink. Isso permite que você interaja com suas missões em tempo real e receba dados de streaming de câmeras e outros componentes.



Figura 16 - Radio de Telemetria

A comunicação entre VANT e GCS pode ser realizada através de outros meios (VANTs que possuem suporte a internet), por exemplo: Internet (uso dos módulos de internet 3G / 4G), comunicação entre rádios wireless (menor distância) entre outros que permitam a comunicação ponto-a-ponto da VANT e a GCS.

2.5.8. ESC (Controlador eletrônico de velocidade)

Tem a função de regular a velocidade do motor elétrico de acordo com o sinal do controle. Converte o sinal PWM do controlador de voo ou do recetor de rádio e aciona o motor fornecendo o nível adequado de energia elétrica.

Um exemplo simples do funcionamento dos ESC dar-se a: Quando é enviado o comando para a VANT subir, então a ESC entenderá que para subir terá de que se aumentar a velocidade de todos os motores simultaneamente, assim a VANT subirá de forma nivelada. A falta do ESC faria com que o piloto não tivesse controle sobre a força aplicada aos motores e então os mesmos funcionariam de forma desordenada.



Figura 17 - ESC (Controlador eletrônico de velocidade)

Existem alguns tipos de firmware de ESCs, SimonK, BLHeli, BLHeli_S, BLHeli32.

Simonk e BLHeli:

Dois dos mais antigos firmwares de ESC para multirotores são SimonK e BLHeli. Originalmente estes eram ambos OpenSource. A maioria dos ESCs vem com o firmware BLHeli ou SimonK pré-instalado [15].

O BLHeli é mais escolhido por causa de sua gama de recursos, interface amigável e por que o SimonK se tornou obsoleto.

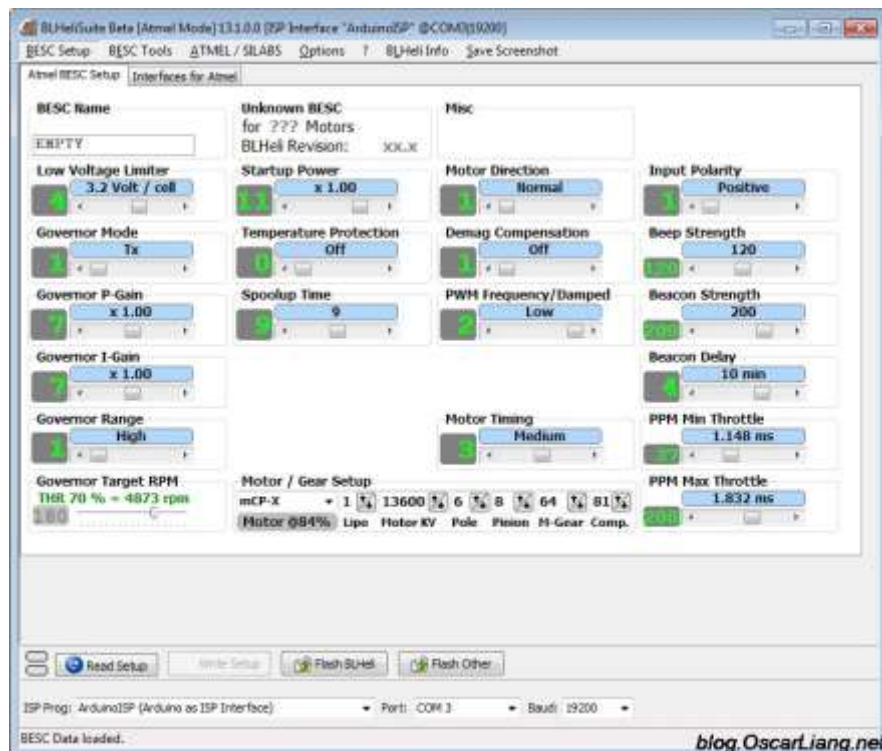


Figura 18 - BLHeli Interface

BLHeli_S:

O firmware BLHeli_S é a segunda geração do firmware BLHeli, desenvolvido especificamente para ESCs que possuem processadores Busybee com hardware PWM (Pulse Width Modulation). Também possui uma interface de usuário muito mais simplificada [15].

BLHeli_32:

O firmware BLHeli_32 ESC é a terceira e mais recente geração do BLHeli. Ele foi escrito especificamente para ESCs de 32 bits e foi fechado para essa iteração. Esses processadores mais potentes permitem um desempenho mais suave, preciso e confiável do que os do ESC anterior [15].

Protocolos ESC

Os protocolos ESC determinam a rapidez com que os sinais podem ser enviados do controlador de voo para o ESC, o que pode ter um grande impacto no desempenho da sua VANT. O protocolo original (mais antigo) é o PWM (Pulse Width Modulation) padrão, tem um atraso de até 2ms, enquanto o Multishot atualmente mais rápido reduziu a latência para apenas cerca de 5-25uS [15].

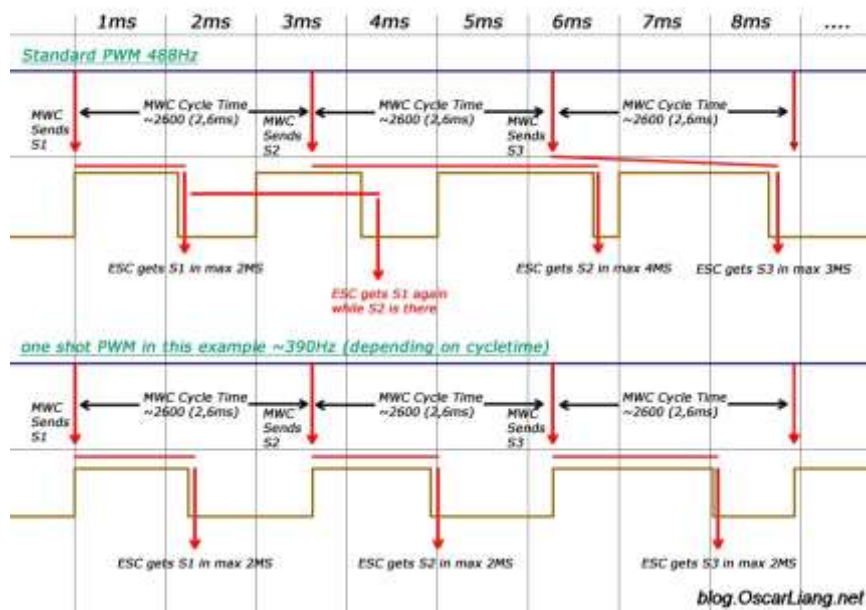


Figura 19 - ESC Protocolo

Aqui há uma lista dos protocolos existentes usados em QuadCopters, podendo ser verificadas direto no site de (OSCARLING - <https://oscarliang.com>).

- Standard PWM
- Oneshot125
- Oneshot42
- Multishot
- DShot (DSHOT150, DSHOT300, DSHOT600, DSHOT1200)
- ProShot

As ESC devem ser calibradas inicialmente. As configurações podem ser feitas através de cartões de programação de fabricação proprietária, através de software proprietário, ou utilizando o RC Transmissor e Recetor, seguindo as instruções do manual do utilizador do equipamento.



Figura 20 - Cartão de programação de ESC

2.5.9. BEC – Circuito eliminador de Bateria

É um regulador de tensão, que converte a tensão principal da bateria LiPo em uma voltagem mais baixa (por exemplo, 2S, 7,4V, 3S, 11,1V ou 4S, 14,8V a 5V). O BEC é normalmente integrado ao ESC, elimina a necessidade de uma bateria separada para alimentar os dispositivos eletrônicos de 5V.

UBEC significa BEC universal ou BEC final. É usado quando o ESC não tem BEC incorporado ou é necessário um sistema de energia independente. Eles geralmente são mais eficientes, mais confiáveis e capazes de fornecer mais corrente do que a BEC. O UBEC é conectado diretamente à bateria principal do multirotor, da mesma forma que um ESC.



Figura 21 - UBEC

- UBEC são mais eficientes no consumo de energia;

- BEC tende a superaquecer com grande diferença de tensão de entrada / saída ou carga grande; A UBEC não tem esse problema e, portanto, é mais confiável;
- UBEC geralmente pode fornecer mais segurança com segurança.

A razão por trás disso é devido à maneira como a tensão é regulada. A maioria das BECs são do tipo linear e as UBEC são do tipo de comutação.

Se seu ESC não tiver BEC, você pode usar um UBEC externo para alimentar seu FC e RX. O cabo de entrada da UBEC deve ser conectado à bateria LiPo e o cabo de saída ao RX e FC. Nenhuma alteração é necessária na conexão ESC.

Mas se você quiser alimentar seu FC e RX com um UBEC, enquanto seus ESCs tiverem BECs integrados, esses BECs primeiro precisam ser desabilitados / desconectados de seu sistema. Simplesmente remova o fio vermelho (5V) do fio do servo de saída do ESC.

Existem dois tipos de BEC, lineares e de comutação.

Lineares:

O BEC linear reduz a voltagem do LiPo principal para 5V, convertendo o excesso de voltagem em calor. Esta não é uma maneira muito eficiente de conversão de voltagem. À medida que a tensão de entrada aumenta ou o consumo de corrente aumenta, mais energia será desperdiçada e convertida em calor. É por isso que este tipo de regulador de tensão não é ideal para alta diferença de tensão de entrada / saída ou aplicação de alta corrente. Geralmente é usado apenas em 3S ou abaixo.

O BEC superaquecido entrará no desligamento térmico e causará perda de energia para o controlador de voo e o recetor de rádio, eventualmente causando uma falha.

Quando a bateria principal é razoavelmente baixa (por exemplo, 7.4v 2S), a energia desperdiçada é relativamente pequena porque não há muita diferença de tensão, portanto, a eficiência é melhor. Mas à medida que você usa uma maior quantidade de LiPo, a eficiência vai a baixo. Muita energia é desperdiçada e convertida em calor.

Comutação:

Os BECs de comutação reduzem a tensão de saída, ligando e desligando a alimentação por milhares de vezes por segundo. Eles não se aquecem como BEC linear, e geralmente lidam com voltagens de entrada mais altas e correntes mais altas muito melhor.

Eles têm uma eficiência muito consistente em uma ampla gama de tensões de entrada / saída, que é de cerca de 85%. Esta é também a escolha para rodar em sistema 4S ou acima se você está atrás de confiabilidade.

Uma desvantagem dos reguladores de comutação é o ruído que eles produzem devido à natureza da regulação de voltagem.

Tabela 2 – Comparação de ESC Linear e ESC de Comutação

ESC. Linear						
Voltagem	Corrente Saída	Corrente Entrada	Força Total	Força Útil	Força Desperdiçada	Eficiência
2S(7.4V)	1A	1A	$7.4V * 1A = 7.4W$	$5V * 1A = 5W$	$7.4 - 5 = 2.4W$	68%
3S(11.1V)	1A	1A	$11.1V * 1A = 11.1W$	$5V * 1A = 5W$	$11.1 - 5 = 6.1W$	45%
4S(14.8V)	1A	1A	$14.8V * 1A = 14.8W$	$5V * 1A = 5W$	$14.8 - 5 = 9.8W$	34%
ESC de Comutação						
2S(7.4V)	1A	0.79A	$7.4V * 0.79A = 5.88W$	$5V * 1A = 5W$	$5.88 - 5 = 1.25W$	85%
3S(11.1V)	1A	0.53A	$11.1V * 0.53A = 5.88W$	$5V * 1A = 5W$	$5.88 - 5 = 1.25W$	85%
4S(14.8V)	1A	0.40A	$14.8V * 0.40A = 5.88W$	$5V * 1A = 5W$	$5.88 - 5 = 1.25W$	85%

2.5.10. Motores e Hélices

Os motores das VANTs são geralmente sem escovas (Brushless), apenas os modelos de tamanho mini possuem motores com escovas (Brushed). Os motores brushless são síncronos e alimentados por um inversor através de corrente contínua de baixa tensão (DC).

Os motores brushless necessitam de um circuito integrado (ESC), para oferecer um melhor controle. São considerados mais eficientes que os motores escovados.



Figura 22 – Motor Brushless

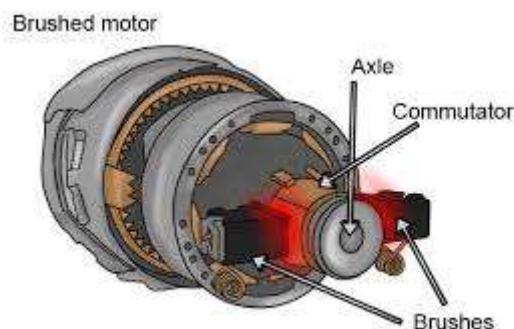


Figura 23 - Motor Brushed

Na maioria dos motores, vemos uma numeração muito importante a primeira que é do estilo 3536/6 ou 2306 indica o tamanho do estator em milímetros, os dois primeiros dígitos indicam o diâmetro do estator e os próximos dois a altura do estator.

Isso sem conhecer mais dados do motor, pode nos dar uma ideia do tamanho da hélice que irá suportar. Um modelo normal para uma VANT de 450 pode ser um motor 3536. Se a VANT for utilizada com GIMBAL para câmeras, deve ser utilizado um modelo superior a este citado anteriormente, pois tem que obter potência extra para o peso da câmera.

O KV é o número que informa as Rotações por Minuto (RPM) de um motor. Para obter o RPM através do KV, tem que se multiplicar esse número pela tensão utilizada. Por exemplo, um motor de 1250KV alimentado por uma bateria de 11.1V giraria a 13.875 rpm (1250×11.1) sem carga. Uma mudança na voltagem mudará a rotação e exigirá a troca da hélice para evitar sobrecarregar o motor.

Quanto mais KV, mais rotações, porém menos torque terá o motor. As VANTs de modelos profissionais geralmente utilizam motores de baixo KV e hélices maiores, com isso conseguem um menor consumo e, portanto, mais autonomia de energia. As VANTs maiores devem utilizar motores de modelos com menor altura, os chamados discos. Esse tipo de motor dá maior torque em baixas rotações.



Figura 24 - Motor Disco 3508

Como selecionar o melhor motor para VANT?

Para escolher o melhor motor a ser utilizado, primeiramente deve-se determinar o peso global da VANT. Por regra, o motor deve ser capaz de fornecer o dobro de impulso que o peso da VANT. Se o impulso fornecido pelos motores é muito pouco, a VANT não responderá adequadamente ao controle e causando dificuldades para levantar. Portanto, se o impulso for alto demais, a VANT poderá se tornar muito ágil e ocasionalmente difícil de controlar.

O cálculo usado para o impulso de cada motor (Thrust per motor) é: $(\text{Peso Global da VANT} * 2) / \text{Número de motores}$

Além do KV do Motor e impulso devem ser consideradas algumas outras características na hora de escolher o melhor motor, essas são: Watt e eficiência, ambas são tão importante quanto as outras.

Um motor de eficiência 70%, produz 70% de energia e 30% de calor. Com motores mais eficientes deixará de desperdiçar uma grande quantidade de energia, assim aumentando o tempo de voo. Quando os motores funcionam de maneira ineficiente, isso pode mudar o RPM do motor e acaba por influenciar na estabilidade da VANT.

Hélices

Existem dois tipos de hélices (CW e CCW) e de diversos tipos de material, ambas são classificadas por tamanho e densidade.



Figura 25 - Hélices Grouper 9x4" / 8mm

As hélices CW são as que giram em sentido horário e as CCW são as que giram em sentido anti-horário.

Geralmente, o aumento da inclinação da hélice (pitch) e comprimento vai aumentar o consumo. Quando se tem mais "pitch" significa rotação mais lenta, mas irá aumentar a velocidade da VANT, aumentando o consumo.

Hélices com menor “pitch” geram mais torque e exige menos energia dos motores, além de melhorar a estabilidade da VANT.

Hélices com maior “pitch”, movem uma maior quantidade de ar, o que pode causar problemas durante o “Loiter”, fazendo com que a VANT não tenha estabilidade.

Hélices de tamanho menor é mais fácil de parar ou acelerar pois possui menor inércia. Hélices menores também significa menor consumo, por esse motivo que as VANTs HexaCopter e OctaCopter tendem a utilizar hélices menores.

2.5.11. Power Module

O PM (Módulo de Força), fornece uma fonte de alimentação estável de 5.37V e 2.25Amp que reduz as chances de sobrecarga nos circuitos. Além de permitir o monitoramento da tensão e corrente da bateria, quando ocorre uma baixa de tensão ou a energia total consumida durante o voo se aproxima da capacidade da bateria, então é ativado um gatilho de RTL (Return-To-Launch), a VANT automaticamente retorna para o local de pouso.

Permite que o firmware do autopilot compense com mais precisão a interferência na bússola por outros componentes.

O PM aceita uma tensão de entrada máxima de 18V (bateria de até 4S LiPo) e corrente máxima de 90Amps.

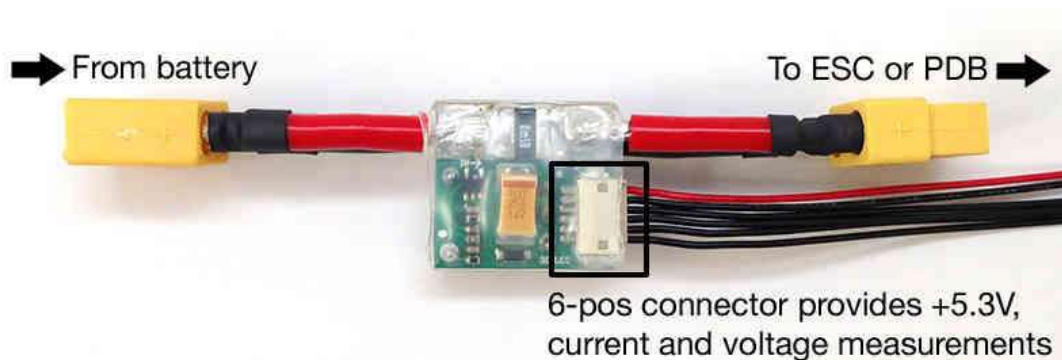


Figura 26 - Power Module (Conector XT60)

2.6. Módulos Opcionais

Os módulos opcionais são equipamentos que podem auxiliar ao piloto em tarefas diversificadas, como por exemplo, usar a VANT como um aerográfico quando a nave é utilizada para fazer imagens aéreas de uma determinada área, topográfico quando é utilizado para fazer topografia de terrenos, entre outras aplicações.

Outros módulos podem ser diretamente investigados no site da ArduPilot.

2.6.1. FPV – First Person View

O First Person View oferece a visão de um verdadeiro piloto durante o voo, colocando uma câmera de vídeo e um transmissor em seu veículo emparelhados com um recetor e um LCD ou óculos de FPV. Um OSD (On Screen Display) opcional ajuda a manter a orientação, fornecendo uma sobreposição de instrumento da aeronave em seu monitor FPV.



Figura 27 - Kit FPV

O sistema de FPV pode ser feito através de IP, com a utilização de 2 Raspberry PI, 1 câmera PI e duas internetes remotas para a comunicação entre os equipamentos. Os requisitos de instalação e configuração do sistema de FPV IP estão descritos no manual de configurações do projeto.

2.6.2. Módulo de medição de distância

O módulo de medição de distância ultrassônico emite um pequeno comprimento de onda quadrada de 40Khz e então ao serem refletidas em outros objetos, as mesmas podem ser captadas por um outro sensor colocado em posição apropriada. Ao detetar a vibração refletida o recetor pode medir a distância em que o objeto se encontra do mesmo.

O modelo atualmente mais usado de forma não profissional é o HC-SR04, pois apresenta boas características e tem uma alta precisão de medição, além de ter uma área cega de aproximadamente de até 2cm. Este modelo em si é compatível com Arduino e pode ser usado diretamente com os GPIO do Raspberry Pi.



Figura 28 - Módulo de medição HC-SR04

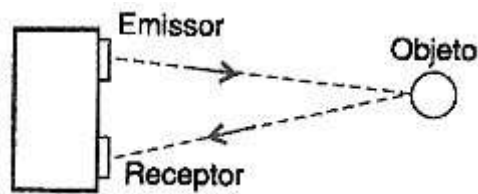


Figura 29 - Funcionamento do módulo de distância

2.6.3. Trens de aterragem

Equipamento que da maior altura da base da VANT ao chão, permitindo que possam ser instalados outros módulos nesse espaço, Ex: Gimbal + câmera, bateria maior na parte inferior da base, EPM *gripper*. Existe a possibilidade de ser instalado trens de pouso de característica retrátil, onde os mesmos são controlados por servo via sinal PWM, diminuindo a aerodinâmica da VANT e consequentemente melhorando no rendimento de energia.



Figura 30 - F450 com trem de aterragem

2.6.4. EPM gripper

O EPM (*Electro Permanent Magnet – Ímã eletromagnético semipermanente*) *gripper* é um módulo dotado de um ímã com aderência suficiente para suportar 15 kg de peso necessitando de reduzidas quantidades de corrente. O dispositivo deve ser colocado sob o multirrotor o mais próximo possível do centro de massa deste. O equipamento vem com chapas metálicas quadradas que devem ser colocadas em objetos que se pretende que o multirrotor transporte. O EMP pode ser ativado mediante sinal PWM do módulo de controlo, ou seja, via aplicação de controlo Ardupilot ou comando de controlo. Não é aconselhável a utilização deste equipamento em multirrotor de pequeno porte, pois quando mais peso maior é o consumo energético.



Figura 31 - EPM Gripper

2.6.5. Joystick / Game Pad

Equipamento utilizado para controlar a VANT com um joystick através do Mission Planner, que envia mensagens “RC Override” para o veículo.



Figura 32 - EPM Gripper

O joystick é utilizado através do Mission Planner, apenas através do GCS é capaz de utilizar este tipo de equipamento, logo pois o GCS será o meio de telemetria entre o joystick e a VANT. A utilização de joystick deve ser feita de forma cuidadosa, já que o tipo de equipamento não te dá maior controle sobre o tamanho de PWM enviado à VANT, como as AXIS do joystick possuem um tamanho menor então o intervalo entre o mínimo e o máximo é bastante pequena, podendo causar comandos indesejados.

Para utilizar um joystick, o mesmo deve ser habilitado e configurado no painel configurações.



Figura 33 - Configuração do Joystick

Deve ser feita a calibração do joystick da mesma maneira como se estivesse utilizando um comando remoto (RC Control).

Quando um joystick é utilizado como alternativa para controle remoto da VANT, deve ser feita uma configuração que reduz o atraso do envio dos dados entre o veículo e o GCS (Mission Planner).



Figura 34 - Diminuir atraso do Joystick

2.7. ROS – Robot Operating System

Os recursos do ArduPilot podem ser estendidos com o ROS (também conhecido como Robot Operating System). [18]

O ROS fornece bibliotecas, ferramentas, abstração de hardware, drivers de dispositivos, visualizadores, transmissão de mensagens, gerenciamento de pacotes e muito mais para ajudar os desenvolvedores de software a criar aplicativos de robô.

O ROS foi projetado para ser o mais distribuído e modular possível, para que os usuários possam usar o ROS que quiserem. Vamos cobrir os componentes que compõem o ROS em outro lugar, mas a modularidade do ROS permite que você escolha quais partes são úteis para você e quais partes você prefere implementar.

O ROS é uma estrutura flexível para escrever software robótico. É uma coleção de ferramentas, bibliotecas e convenções que visam simplificar a tarefa de criar um comportamento de robô complexo e robusto em uma ampla variedade de plataformas robóticas.

3. Execução

3.1. Planeamento

O projeto foi apresentado como um dos mais difíceis a ser executado, pois, haviam uma certa resistência de outros companheiros de turma em relação ao mesmo, devido algumas tentativas anteriores que não foi concluída com sucesso. Tendo em vista que este projeto é um dos mais completos em relação a colocar em prática os conhecimentos adquiridos durante o curso, então foi decidido escolher tal projeto pelo seu grau de dificuldade.

Inicialmente foi investigado sobre qual seria o tipo de controladora de voo a ser utilizada, logo apareceram algumas opções como tais, MultiWii, ArduPilot 2.8, NAZA e PXFMini. Com pouca experiência no assunto, foi investigado sobre utilização futura de tal equipamento, se o escolhido poderia ser utilizado em futuros projetos utilizando VANTs, ou até mesmo projetos voltados a robótica. Das controladoras investigadas a que mais aproximou-se do objetivo foi a PXFMini, pois trata-se de uma controladora baseada em Linux onde é acoplada diretamente em um Raspberry Pi e que tem a característica “Ready-to-Fly”, além de já vir com o ROS (Robot Operating System) instalado.

Depois que foi decidido qual controladora utilizar, foi possível contar com a ajuda do orientador do projeto para a aquisição do equipamento, juntamente com o departamento responsável pela compra. Não demorou muito e já tinha em mãos a controladora de voo PXFMini, com GPS e Power Module. Com tudo o principal já adquirido, pode ser dado início aos trabalhos.

3.2. Materiais

3.2.1. Materiais disponíveis

Tabela 3 - Materiais disponíveis para o projeto

Item	Descrição	Quantidade
1	Transmissor Turnigy 9X com recetor de 8 canais	1 (und)
2	Fuselagem H.A.L. (Heavy Aerial Lift) 775 mm de alumínio (com 6 braços)	1 (und)
3	ArduPilot Mega 2.8	1 (und)
4	GPS + Compass para ArduPilot Mega (Conector DF13)	1 (und)
5	Power Módulo ArduPilot Mega	1 (und)
6	BEC 5v	2 (und)

7	Conjunto de Rádio Telemetria 915 MHz (Transmissor + Recetor)	1 (und)
8	Placa de distribuição de energia OctoCopter	1 (und)
9	ESC Turnigy Trust 45A com BEC	6 (und)
10	Motores Turnigy D3536/6 1250 KV Brushless Outrunner	6 (und)
11	Hélices CW Slow Fly Glow in the Dark 10x3,8	4 (und)
12	Hélices CCW Slow Fly Glow in the Dark 10x3,8	4 (und)
13	Bateria Turnigy 5000 mAh, 3S, 40C	2 (und)
14	Carregador Max 80w 7A	1 (und)
15	Raspberry Pi 2	1 (und)
16	Câmara Pi com case	1 (und)
17	Modem Huawei E8278	1 (und)
18	Sensor HC-SR04 modificado para PWM	1 (und)
19	Óculos FPV Quanam	1 (und)
20	Bateria 1500 mAh 25C 7,4V (FPV)	1 (und)
21	Câmara FPV com transmissor Skyzone	1 (und)
22	Recetor de vídeo RC832 com 40 canais	1 (und)
23	Micro SD Kingston 32 Gb Class 10	1 (und)
24	Cartão GSM com internet móvel tarifada ao tempo Vodafone	1 (und)

Tabela 4 - Materiais adquiridos

Item	Descrição	Quantidade
1	PXFMMini (Erle Robotics)	1 (und)
2	Power Módulo PXFMMini (Conector JST GH 6 pinos)	1 (und)
3	GPS + Compass PXFMMini (Conector JST GH 6 pinos)	1 (und)
4	Header - Female socket SAM1193-20-ND (40 posições)	1 (und)
5	Hélices CW Graupner 9x4" com furo de 8mm	3 (und)
6	Hélices CCW Graupner 9x4" com furo de 8mm	3 (und)
7	Raspberry Pi case com altura extendida	1 (und)
8	Raspberry Pi 2 + vídeo-in RCA	1 (und)
9	Router móvel Telenor 4G	1 (und)
10	Frame F450 DJI + PDB	1 (und)
11	Trem de aterragem (plástico - Chinês)	2 (und)
12	Bateria Turnigy 5000 mAh, 3S, 40C (Conector XT90)	1 (und)
13	Conectores XT90 (par)	1 (und)
14	Conectores XT60 (par)	1 (und)
15	Suporte para GPS	1 (und)
16	LiPo Checker (chegar a carga da bateria geral e por célula)	1 (und)

3.2.2. Testes realizados nos materiais

Foram realizados testes de uso nos materiais disponíveis, para que pudesse aferir o funcionamento de todo material.

3.2.2.1. Transmissor Turnigy 9X com recetor de 8 canais

Foram realizados testes de transmissão e recepção de sinal, os testes foram realizados nas duas placas controladoras de voo existentes, a PXFMini e a ArduPilot Mega 2.8. Primeiramente os testes foram realizados na ArduPilot Mega e o equipamento mostrou está funcionando de maneira adequada. Logo em seguida foi iniciado os testes na PXFMini, onde de primeira não funcionou, então foram realizadas investigações sobre o funcionamento do mesmo e foi visto que era necessário adicionar o equipamento PPM Encoder (descrito no item 2.5.2), a controladora de voo PXFMini utiliza o PPM-SUM para controle de sinal PWM.

PPM-SUM é a sumarização dos 8 sinais PWM emitidos pelo comando sem fios.

Existem três possibilidades de obter o equipamento PPM Encoder, seriam: Comprar o equipamento que tem um custo de 8,00 € + 25,00 € de postagem, totalizando 33,00 € com 15 dias de entrega.

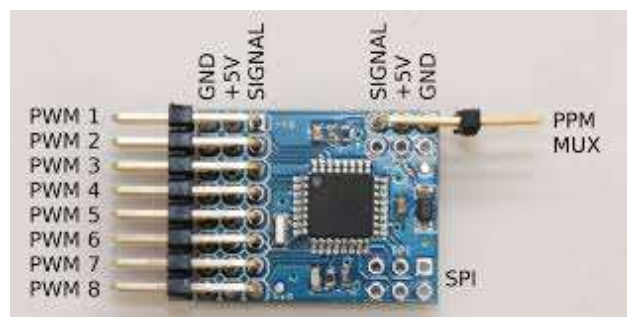


Figura 35 - PPM Encoder

A outra possibilidade seria fabricar o equipamento utilizando o esquema de circuitos eletrônicos exibido anteriormente na Figura 10, através de uma empresa que produz circuitos impressos, porém, os custos para produzir tal componente em baixa quantidade seriam demasiado alto em torno de 60,00 €, outra maneira seria utilizando um Arduino AT328P com o firmware da referência [14].

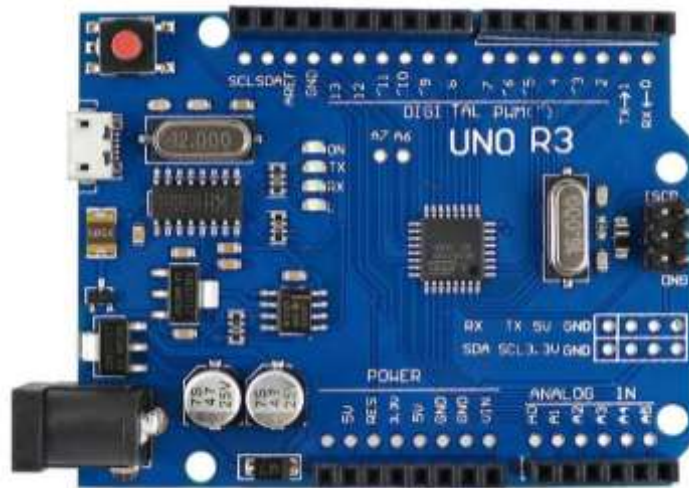


Figura 36 - Arduíno AT328P



Figura 37 - Circuito impresso

Foi decidido fabricar o PPM Encoder de modo “Home Made” já que o componente feito em fábrica de circuitos impressos oneraria o valor final do mesmo. O mesmo componente feito em modo “Home Made” tem um valor final de produção de aproximadamente 5,00 €.

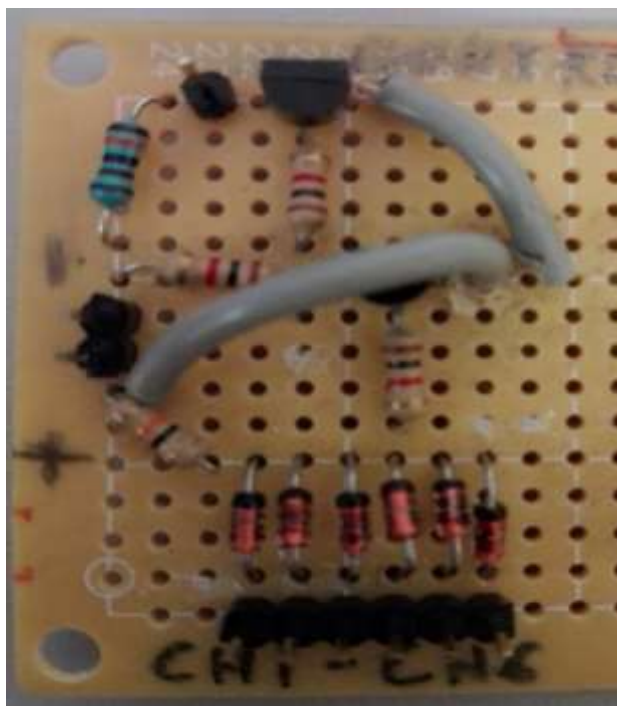


Figura 38 – PPM Encoder HomeMade

3.2.2.2. ArduPilot Mega 2.8

Os testes realizados na ArduPilot Mega 2.8, foram o de funcionamento com o Mission Planner que inclui a atualização de firmware e funcionamento dos dados de telemetria obtidos da VANT. Após a atualização de firmware foi verificado que o equipamento se encontra com problemas na memória flash, pois a conexão com o computador através do cabo usb é perdida após desconectar o dispositivo, voltando a funcionar após uma nova conexão e nova instalação de firmware.

Depois do firmware instalado e configurado, o equipamento funciona adequadamente, e com o sistema de telemetria o mesmo consegue emitir os dados referentes a VANT.

Também em conjunto com as controladoras de voo, foram testados os seus respectivos Power Modules (Módulos de Força), equipamento descrito no item 2.5.11 e mostrou que funcionaram de maneira adequada.

3.2.2.3. GPS + Compass

Existem 2 módulos de GPS + Compass, sendo um que funciona diretamente com o ArduPilot Mega e outro que funciona com o PXFMini, o que muda entre os dois é apenas o tipo de conector, o referente ao ArduPilot Mega tem dois conectores do tipo DF13, sendo um para o GPS e outro para o Compass, já o do modelo para o PXFMini vem com um conector do tipo

JST GH (Japan Solderless Terminal) para ser conectado na porta UART da board, ambos funcionam adequadamente realizando o 3D FIX em ambiente externo e livre de impedimentos para o sinal do GPS.

É aconselhável a utilização deste equipamento em ambiente externo sem impedimentos, sem paredes ao redor e sem telhados, para que possa ter um maior sucesso em sua captura de sinal, já que se trata de um módulo GPS de baixo custo.

O Compass (Bússola) embutida nos módulos também funcionam de maneira adequada, mas somente se o equipamento estiver isolado de interferências eletromagnéticas causadas pelos motores das VANT's, e para diminuir essas interferências é utilizado um suporte para GPS de mais ou menos 10cm de altura.



Figura 39 - Suporte GPS

3.2.2.4. Rádio Telemetria 915 MHz

O conjunto de rádio responsáveis por transmitir e receber os dados de telemetria da VANT, está em situação de falha, um dos rádios tem uma falha na conexão USB e somente consegue parrear com o outro rádio através de muita manipulação no cabo, até que se ache uma posição que funcione, além de o equipamento ser de frequência não permitida para uso em território europeu.

Tabela 5 - Diferenciação de frequência de banda por território

Região	Modelo de Sinal	Padrão
Europa	433 MHz	ETSI EN300 220 7.2.3
USA	915 MHz	FCC 15.247
Canadá	915 MHz	RSS-210 Annex 8.1
Austrália	443 MHz	LIPD-2000 item 17
	915 MHz	LIPD-2000 item 52
Reino Unido	443 MHz	IR2030/1/10
Nova Zelândia	443 MHz	Notice 2007, Schedule 1
	915 MHz	
Brasil	443 MHz	Anatel - Resolução nº 506/2008
	915 MHz	
Argentina	915 MHz	Comisión Nacional de Comunicaciones
África do Sul	443 MHz	2008 RR 5.138. Government Gazette Nº 31127, Notice Nº 713 of 2008 and Government Gazette Nº 31290, Notice Nº 926 of 2008 refer

Para que não fosse tempo perdido com esse equipamento, foram investigadas duas soluções, onde seriam: Conexão via WIFI onde utiliza um adaptador wireless no Raspberry Pi caso o mesmo já não o possua que é o caso do Raspberry Pi 2 e então configurar para que o adaptador wireless funcione como AP (Access Point) então a GCS e a VANT estariam na mesma rede e comunicando entre si, com esta solução a distância se tornaria um problema já que os adaptadores wireless utilizados têm uma limitação de distância, dependendo da marca e modelo. Entretanto foi investigado uma segunda solução que utiliza a comunicação remota através de rede móvel, assim as barreiras entre as distâncias seriam superadas. A solução de rede móvel é feita através da utilização de dois routers com suporte a conexão remota 3G ou 4G e suporte a configuração de redirecionamento de portas e também a utilização DNS (por exemplo DNS gratuito do No-IP), caso o router não possua a configuração de DNS a mesma pode ser feita diretamente no GCS (PC portátil) pelo aplicativo DUC do próprio fabricante NOIP e no Raspberry Pi instalando-o e configurando-o para iniciar juntamente com o kernel e configurando a aplicação do NOIP para atualização automática a cada 1 minuto.



Figura 40 – Router 4G

O teste de comunicação remota foi feito juntamente com a equipa que está a desenvolver o projeto de comunicação 4G de Drone, o teste foi realizado com sucesso.

3.2.2.5. ESC Turnigy Trust 45A com BEC

Os testes nos controladores eletrónicos de velocidade foram realizados com apoio de um Raspberry Pi 2, o código desenvolvido em python para controle da ESC e uma bateria de LiPo de 5000 mAh. O código em python está apresentado no Anexo A.



Figura 41 – Motor Turnigy D3536/6 1250kv

Seguindo o manual do fabricante dos controladores eletrônicos de velocidade, foi realizado calibração nos equipamentos.

Para que os testes de motores e ESC's sejam realizados, primeiramente devem-se ser seguidos alguns passos:

- Ter a biblioteca PIGPIO instalada no Raspberry Pi;
- Retirar as hélices dos motores;
- Ter o motor que será testado bem preso a uma base;
- Ter cuidado para não inverter a polaridade da bateria quando conectar a ESC;

Então depois de tudo verificado, deve-se fazer as conexões de acordo com a Figura 37. Quando estiver a realizar as conexões deve-se ser bem atencioso, pois o fio vermelho da ESC não deve ser conectado ao Reaspberry Pi, este fio vermelho tem a função de alimentar e ser alimentado com 5V, se o mesmo for conectado no Raspberry Pi, poderá queimar o processador do mesmo. O lado da conexão com o motor pode ser ligado em qualquer ordem, logo se quiseres inverter a rotação o fio vermelho ou amarelo deve ser trocado com o preto. Não deve ser conectado à bateria logo de início, o programa a ser “interpretado” do Anexo A irá solicitar o momento certo de conexão com a bateria, para evitar algum curto circuito por conectar e

desconectar e acabar enganando-se na conexão é aconselhável a utilização de uma chave interruptora, assim podendo apenas ligar ou desligar a energia pela chave.

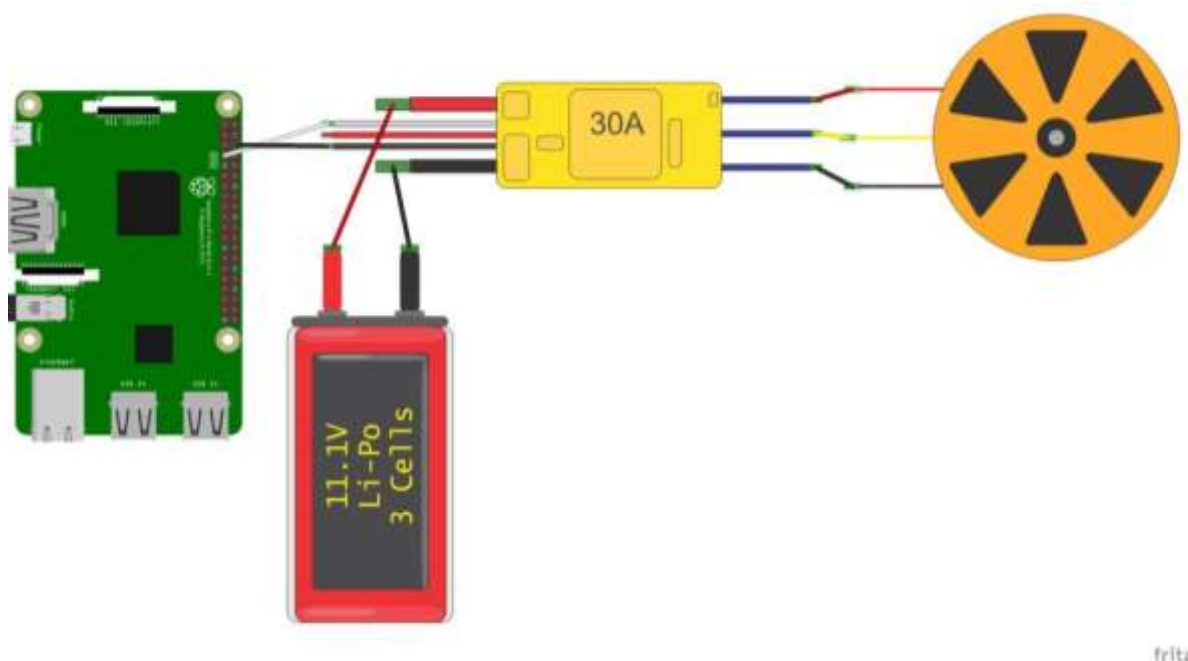


Figura 42 - Teste de motor e ESC

Quando o programa é “interpretado/executado”, o mesmo apresentará algumas opções. “CALIBRAR OU MANUAL OU CONTROLE OU ARM OU STOP”. É aconselhável que seja feita a calibração das ESC’s antes do primeiro uso, com a calibração das ESC’s, automaticamente é feita a calibração da aceleração dos motores, acionar freios, verificar a conexão para a rotação correta de cada motor em sua determinada posição. Usando a função de “CONTROLE” é possível que possa controlar a velocidade aplicada ao sistema com telhas de atalho “a, d, q, e”, para utilizar as teclas de atalho é necessário que entre no modo de “CONTROLE” e depois no modo “ARM”.

Os testes realizados não demonstraram falhas com os equipamentos, pelo menos nenhuma falha visível. Os freios e rotação aplicada ao sistema (ESC e motor), foram atendidos de maneira esperada.

3.2.2.6. Motores Turnigy D3536/6 1250 KV Brushless

Os testes realizados nos motores, tanto de velocidade como de orientação de rotação foram feitos em conjunto com os testes dos ESC’s, então teoricamente e fisicamente funcionaram de maneira adequada.

Tais motores foram desenvolvidos para serem utilizados com baterias de 2 até 4 células e com um máximo impulso de 1.270g.

Os motores disponíveis exigem hélices de tamanhos específicos estes são:

Carga de 7.4V devem ser utilizadas hélices de tamanho 10x5;

Carga de 14,8V devem ser utilizadas hélices de tamanho 9x4.7;

3.2.2.7. Bateria Turnigy 5000 mAh, 3S, 40C

Foram disponibilizadas 2 (duas) unidades da bateria de 5000 mAh 3S do fabricante Turnigy e já visivelmente se via o inchaço das duas unidades, foi investigado o motivo de certa característica apresentadas.

O próprio fabricante das baterias diz que o inchaço na bateria se dá devido a má utilização, ou seja, a bateria pode ter sido armazenada de maneira inadequada, pode ter excedido a carga máxima com o uso de um carregador de baterias inapropriado, uma descarga elétrica causada por um curto circuito onde o polo positivo e negativo tenham se tocado gerando este curto circuito.



Figura 43 - Bateria inchada

As baterias de LiPo (Polímero de Lítio) requer alguns cuidados com o seu manuseamento, no item 2.5.6 deste relatório são apresentadas 15 dicas de como estas devem ser manuseadas.

3.2.2.8. Carregador Max 80W 7A

O carregador inteligente de baterias que foi disponibilizado, não vinha com a fonte de energia original então foi verificado junto ao fabricante que o carregador pode ser utilizado com qualquer outra fonte que tenha uma saída de energia de 12v e com a corrente de até 7A então foi utilizado uma fonte alternativa que tem uma saída de 12v e uma corrente de 1,2A.

As baterias disponíveis foram dispostas em local seguro para serem carregadas pelo carregador inteligente, tal carga se deu por meio de balanceamento e com uma corrente de 0,7A para garantir que as baterias fossem carregadas lentamente e garantir uma maior vida útil da mesma e quando foi acionado o modo de verificação de células das baterias o carregador mostrava que a célula de Nº 1 em ambas as baterias estava com a voltagem oscilando, o mínimo deveria ser de 3,30v e o máximo de 4,20v e o carregador mostrava uma oscilação de 1,20v e quando “totalmente” carregada nunca mostrava a carga máxima da célula. Na primeira ocasião foi achado que as baterias poderiam está com problemas, no entanto foi realizado o teste com uma nova bateria adquirida posteriormente e o mesmo continuou apresentando o mesmo comportamento, isto não quer dizer que o carregador não faça o que tem que fazer “carregar baterias”, mas o monitoramento da carga pode-se dizer que deixa a desejar já que apresenta este defeito.

3.2.2.9. Kit FPV (First Person View)

Com o objetivo de assistir diretamente as missões planeadas com a VANT, foi instalado um sistema de FPV no mesmo. O kit de FPV dispõe de uma câmera FPV com transmissor skyzone, um par de óculos Quantum com ecrã de 5 polegadas, bateria 2S de 1500 mAh e um recetor de vídeo RC832 com 40 canais, o teste realizado visava apenas transmitir e rececionar a imagem utilizando o equipamento de FPV, no entanto a câmera apresentou problemas, pois, a mesma não ligou na primeira tentativa, foi admitido que o problema seria a alimentação da mesma que estava sendo feita por um BEC 5V, logo a dúvida de que o problema fosse na alimentação foi sanado devido ao teste realizado por uma fonte externa com alimentação de 5V. Em uma outra tentativa aleatória foi visto que a câmera ligou e até chegou a transmitir imagem, mas 30 segundos depois a mesma para, desliga e não volta a funcionar.

Uma outra solução foi investigada para que o sistema de emissão e receção de vídeo viesse a funcionar de maneira adequada sem depender do módulo existente e também rompendo a barreira da distância. Utilizando um Raspberry Pi com saída de vídeo do tipo RCA (neste caso a saída de vídeo do tipo RCA foi escolhida por conta de que os óculos Quantum disponíveis

tenham apenas entrada de vídeo do tipo RCA, mas se forem utilizadas uma tela com qualidade HD pode ser utilizada a saída de vídeo HDMI), uma câmera RPi (Câmera própria do Raspberry Pi), 2 (dois) routers de conexão móvel 3G ou 4G com disponibilidade de configuração DNS e redirecionamento de portas, sendo um router para o Raspberry Pi da VANT que terá a função de streamer de vídeo e um router para o Raspberry Pi com a função de receptor de vídeo. O sistema de FPV IP funcionou de maneira satisfatória.

As configurações de uso do FPV IP estão descritas no manual de configuração do projeto.

3.2.2.10. Fuselagem H.A.L. (Heavy Aerial Lift)

A fuselagem de uma VANT possui algumas particularidades que devem ser verificadas, entre elas estão:

O corpo da fuselagem, o tipo de trem de pouso, as conexões realizadas na PDB (Power Distributor Board) para receber as ESC's, motores e outros equipamentos. A fuselagem disponível para o projeto possui o corpo de alumínio relativamente leve, alguns dos braços encontram-se com folgas, tais folgas podem causar vibrações intensas e interferir na leitura dos sensores.

Uma nova frame no formato QUAD (F450 DJI – para quatro motores, apresentado na Figura 45) foi adquirida para que pudessem ser realizados testes usando os motores, ESC's, e os outros equipamentos disponíveis. Uma frame de menor tamanho possibilita um melhor manuseio do veículo já que é necessário fazer a calibração do Compass e tal calibração se dá por fazer rotações manuais com o veículo.

3.2.3. Montagem da VANT

Após a desmontagem do veículo existente, testes foram realizados em todo material conforme o item 3.2.2, foi realizada a montagem dos mesmos sobre o frame H.A.L (Heavy Aerial Lift) HexaCopter, para que os testes de confiabilidade fossem executados e tentar eliminar quaisquer dúvidas sobre os problemas encontrados durante os testes de materiais.

Antes de iniciar a montagem, os motores foram lubrificados com óleo WD40, indicado por profissionais em pilotagem e montagem de VANTs (HP Modelismo). Foram verificadas as soldas já realizadas na PDB, os fios condutores passaram por testes de continuidade, foi

necessário refazer as conexões presas com manga térmica. Após estes ajustes concluídos, foi dado início e termino à montagem do veículo.



Figura 44 - HexaCopter montado

Logo com a conclusão dos testes de voo foi visto que seria necessário montar uma segunda VANT (Frame 450 – QuadCopter) com as partes principais já existentes do HexaCopter e com mais outras adquiridas posteriormente.



Figura 45 - F450 Montado / Nova VANT

Os motores foram fixados aos braços da VANT através de parafusos devidamente apertados para não exceder a pressão e muito menos permitir que com a vibração do veículo os mesmos venham a soltar-se. As ESCs foram fixadas utilizando abraçadeiras do tipo “ZipTie” permitindo um maior controle e fixação no posicionamento do equipamento. Para fixar o Raspberry Pi + PXFMini junto ao corpo, foi utilizado uma fita adesiva de dupla face juntamente com uma abraçadeira em velcro, assim faz com que o controlador de voo fique em uma posição estável, fixa e evitando que perca o nível do acelerômetro durante a utilização. O controlador de voo foi acomodado em uma case para Raspberry Pi de acrílico resistente a quedas e com uma altura estendida para que pudesse acomodar o RPI juntamente com o PXFMini. Por não ser uma tarefa fácil encontrar uma caixa de acrílico de tal qualidade e também existe uma dificuldade em cortar esse tipo de material, foi feita de maneira abrupta aberturas na parte superior e parte frontal da case do RPI, para que os cabos de força, GPS, câmera, e controle de ESC pudessem ser conectados em seus respectivos locais.

A bateria de 5000 mAH fixada na parte inferior do veículo esta presa com duas abraçadeiras de velcro, ambas com uma boa qualidade de fixação, já que a bateria deve está bem fixa e também deve ter a possibilidade de remoção da mesma para manutenção (carregar, armazenar, trocar). A bateria adquirida posteriormente veio por defeito com um conector do tipo XT90, então foi necessário a fabricação de um adaptador XT90 para XT60, já que o power module da PXFMini vem por padrão de fábrica com conector XT60.

Com a troca da fuselagem, foi necessário a compra de um trem de aterragem para que o veículo possuísse altura suficiente para acomodar a bateria e possivelmente uma câmera. O trem de aterragem adquirido foi do mais simples possível e sem muita qualidade já que era apenas para uso de testes e que o veículo original já possui um trem de aterragem com melhor qualidade.

3.3. Testes de confiabilidade

Antes de iniciar os testes de voo, o Comando Remoto, ESCs, Bússola, Acelerômetro, Bateria, devem ser calibrados conforme mostrado no manual de configurações.

3.3.1. Testes de voo guiado

Logo após o veículo devidamente montado, deu-se por ligar a energia e esperar o mesmo iniciar todos os sensores e apresentar seus comportamentos iniciais. Inesperadamente os motores começaram a emitir um “beep” continuamente, foram realizadas investigações diante o fórum da “Erle Robotics”, outros fóruns de entusiastas em drones, mas nenhuma resposta concreta era obtida, então, por meio de tentativas em meio ao desespero, já que nos testes diretamente com o Raspberry Pi os motores funcionaram de maneira adequada, foi iniciada uma investigação sobre as configurações do Mission Planner, ate que se chegou a conclusão que com a mudança do parâmetro RC_SPEED (velocidade em Hertz com que a ESC recebe atualizações) em configurações de 450 Hz padrão para 120 Hz, a controladora de voo aceitaria as ESCs, já que foi aceito um valor, outras tentativas de encontrar o valo máximo da frequência das ESCs foram realizadas até que foi encontrado para esse tipo de ESC no qual o fabricante e muito menos o distribuidor anuncia o seu firmware, os valores de 110 Hz mínimos até 180 Hz máximos.

Para que a VANT pudesse “armar” (está pronta para decolar), foi necessário realizar um “preFlight”, o Mission Planner precisa checar alguns parâmetros antes de “armar” a VANT. Esse “preFlight” dar-se pela checagem dos seguintes parâmetros.

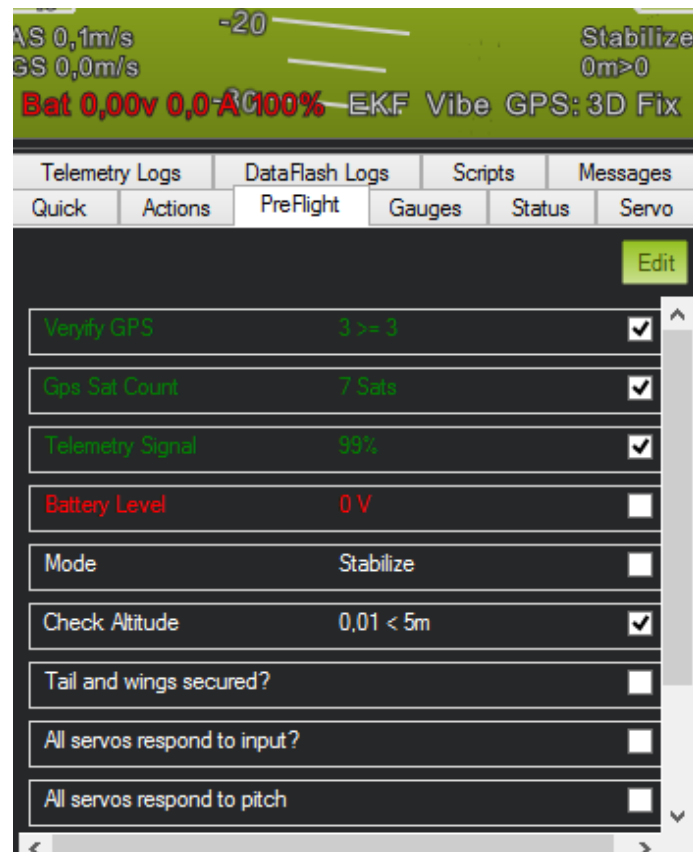


Figura 46 – Parâmetros do "preFlight"

Os parâmetros mostrados acima são os básicos para garantir que a VANT possa levantar voo sem nenhum problema, além desses parâmetros existem outros mais que podem ser verificados no painel de configurações. O "preFlight" pode ser desativado caso o piloto garanta que tudo está de acordo, forçando assim que o Mission Planner permita que a VANT seja "armada", mas sem garantir sucesso na decolagem e no voo.



Figura 47 - Parâmetros de Pre-Arm

No caso de utilizar a VANT sem um GCS, a controlar o voo apenas com o comando remoto, seria necessário colocar o veículo em modo “Arm” então foi investigado junto ao site da ARDUPILOT, para que entre em modo “Arm” é necessário obedecer alguns comandos, tal que faz com que o barômetro e o giroscópio sejam também reinicializados, esses comandos dão-se em deixar o Throttle (acelerador) no mínimo e o leme (Rudder) no máximo por cinco segundos, logo após os cinco segundos o veículo é armado.



Figura 48 – Comando para “Armar” a VANT

Todas as configurações a serem seguidas para um primeiro voo estão descritas no manual de configurações do projeto.

Com todos os parâmetros iniciais checados e o veículo armado, pôde-se dar início a tentativa do primeiro voo. Inicialmente não aconteceu como esperado, pois, a insegurança e o medo de que o veículo pudesse cair e quebrar alguma peça importante era muito grande. Como para realizar um voo utilizando o comando remoto Turnigy 9x poderia ser uma alternativa mais segura pois assim daria mais estabilidade no voo, mas a necessidade da utilização de um PPM Encoder se tornava um pouco difícil, logo o equipamento que existia era um “Home Made” apresentado na Figura 38. Por conta do problema em manter as conexões, foi decidido utilizar um Joystick / Game Pad para o primeiro voo.

A utilização do joystick não ocorreu como esperado, como citado anteriormente no item 2.6.5 o joystick possui um AXIS de tamanho menor que os dos comandos remoto (RC Control) e não foi possível manter o controle do veículo durante a tentativa de voo, então a mesma levantou e em seguida virou totalmente para o lado, causando uma queda. Com a queda ocorrida a insegurança em levantar voo tornou-se ainda maior, o medo de quebrar hélices, motores, ou até mesmo o módulo controlador de voo.

Os testes realizados para o primeiro voo deram-se com a fuselagem existente H.A.L (Heavy Aerial Lift), a mesma que apresenta problemas de fixação em dois dos seus seis braços e possui um corpo feito de alumínio que também pode causar interferência eletromagnética. Por esses motivos acreditou-se que o corpo poderia estar atrapalhando na calibração do compass (bússola), as folgas nos braços poderiam estar atrapalhando os sensores, as hélices existentes e incompatíveis com o modelo de motor utilizado poderiam estar causando uma falha na estabilidade durante a tentativa de decolagem, além da falta de experiência com drones e os controles do mesmo não foi previamente verificado o modo de voo escolhido para um primeiro teste, o modo que estava pré-definido era o “Stabilize” onde os comandos são muito mais difíceis de ser controlados, o modo recomendado para uma pessoa sem muita ou nenhuma experiência é o “Alt Hold”, onde o sistema encarrega-se de manter o veículo estabilizado durante o voo.

Foram realizados outros testes de voo, desde já testes com mais segurança pois segurava o veículo pela cúpula com as mãos para que a mesma não virasse e durante algumas tentativas desordenadas foi verificado que realmente o problema encontrava-se em manter a estabilidade

do veículo. A estabilidade dar-se por ter o compass bem calibrado, o acelerómetro e o nível da altura calibrados.

Foi solicitada ajuda para os colegas do outro grupo do projeto de “Comunicação 4G de drone”, já que o mesmo possuía um veículo HexaCopter nova e aparentemente sem problemas, em troca desta ajuda, foi oferecido ajuda com o projeto que o grupo estava a desenvolver. Durante os testes com o veículo alternativo, foi verificado que a placa controladora de voo PXFMini realmente não apresentava problemas, já que desta vez o veículo não virou, porém, também não levantou voo, acreditando ser por motivo de falta de força já que a estrutura é um pouco demasiada para o tamanho da bateria que é de 2000 mAH.



Figura 49 – Drone alternativo (Grupo 4G)

Deu por início os testes com o novo modelo (QuadCopter), foi decidido realizar novamente um teste utilizando o joystick, foram realizadas 3 tentativas de decolagem, devido a falta de controle com a força adicionada nos AXIS do comando. A primeira tentativa, o veículo levantou por cerca de 0,5m andou para a direita e por um momento de medo e descontrolo do piloto, voltou a cair em uma área descampada. A segunda tentativa, já foi mais controlada, o veículo levantou os mesmos 0,5m e inconscientemente o piloto por não ter notado que o “Pitch” não estava invertido, ao invés de o mandar para frente, acabou por o mandar para trás, causando um pequeno acidente entre o piloto e o veículo. Depois do ocorrido foi verificado que um dos pares das novas hélices havia sido quebrado, o piloto obteve um corte

na perna, então foi decidido trocar as hélices novas e testar novamente as hélices antigas que por mais uma vez não deu estabilidade ao veículo causando uma queda antes da decolagem, apenas por comprovar que as mesmas não são compatíveis para o tipo de motor. Na terceira tentativa, foram tomadas algumas precauções, o veículo foi levado para uma área maior e mais aberta, depois do veículo armado foi lentamente adicionado aceleração ao mesmo para que pudesse lentamente levantar voo e assim foi realizado com sucesso, o veículo levantou a uma altura de dois metros e pairando no ar, logo em seguida foi também lentamente diminuindo a aceleração para que o veículo pousasse em segurança. Poucos momentos antes do veículo terminar o pouso em segurança, o piloto soltou o acelerador repentinamente causando um desequilíbrio do veículo e ocasionando em uma nova queda, porém mais leve do que as anteriores e sem nenhum acedente ou estrago.

3.3.2. Testes de voo autónomo

Os testes foram realizados em campo aberto, pois seria de certa forma a área mais segura para os realizar. Para atingir o objetivo do projeto que seria realizar missões autónomas através do Mission Planner, foi criada uma missão com 3 pontos onde seriam, Takeoff (decolar), Loiter Time por 10s (Pairar no ar), Land (Pousar), executados a uma altura de 1,00m (altura no qual teria o controle do veículo com as mãos, evitando de uma possível queda), após a missão carregada na controladora de voo, foi acionado o modo de voo “Auto” e solicitado que se iniciasse a missão (através do botão Resume mission em Flight Data - > Actions), o veículo de início não armou e apresentou o erro de “BAD COMPASS”, onde estava a falhar com a calibração do compass e também por motivos de localização não estava com o “GPS Fix” parâmetro essencial para um voo autónomo. Após a calibração do compass ser realizada com sucesso e encontrado um local com melhor visibilidade de GPS e assim por consegui um “GPS Fix”, os procedimentos anteriores para gerar, carregar e iniciar uma missão foram reexecutados, mas agora “armado” com sucesso, decolando a uma altura de aproximadamente 1,30m o veículo rapidamente dirigiu-se para o segundo ponto onde estava marcado para o “Loiter” e o executou por 10s, então imediatamente foi para a área de “Land”, o pouso não foi tão suave como imaginava, pois deve ser feita uma compensação da altura do solo, a altura na qual se indica que o veículo irá pousar e o local para não acabar pousando em uma mata de difícil acesso.

Executar uma missão autónoma exige uma maior noção e controlo na utilização da VANT, logo que nunca se deve realizar uma missão autónoma sem um comando remoto já

conectado e por perto para que no momento em que perca o sinal de GPS o piloto possa tomar o controlo do veículo e que sempre tenha os parâmetros de Failsafe bem configurados. Hoje em dia os módulos de GPS vendidos para este tipo de veículos “domésticos / baixo custo” não são tão confiáveis o quanto precisamos para executar uma missão autónoma, já que para o mesmo é necessário que o GPS esteja em modo fixo independente da localização e vemos que isso não é o que acontece com esses módulos mais básicos.



Figura 50 - Primeira missão autónoma

4.Considerações finais

4.1. Conclusões

Este trabalho consistiu na investigação acerca da possibilidade de utilizar uma VANT para realizar missões autónomas, abrangendo as etapas de Estado do conhecimento e execução das tarefas programadas para obtenção do resultado.

Para o desenvolvimento completo do projeto, não foram o suficiente os conhecimentos inicialmente previstos, foi necessária muita investigação, leitura de documentos, auxílio de alguns profissionais do ramo de VANT além do orientador que sempre esteve disposto a ajudar quer seja com alguma dificuldade ou com assuntos financeiros relacionados ao projeto, tal como a aquisição de uma nova placa controladora de voo, um módulo GPS e um Power Module. O conhecimento adquirido com as investigações realizadas durante o Estado do conhecimento vem para provar que um Engenheiro Informático está plenamente capacitado para realizar tarefas no qual não está apenas ligado a desenvolvimento de softwares ou análise de sistemas.

Apesar dos problemas iniciais, o medo, a insegurança ou até mesmo a possível fragilidade dos materiais já disponíveis, além da falta de estabilidade do veículo ao tentar decolar, as quedas sofridas, os acessórios perdidos, o projeto demonstrou-se bem-sucedido a partir do momento em que se deu sucesso ao primeiro voo guiado, desde este momento a insegurança e o medo já haviam desaparecidos, abrindo espaço para sucesso total do projeto.

Após a realização dessas considerações, pode-se afirmar que o projeto satisfaz com seu objetivo em realizar missões autônomas diretamente com a utilização de um GCS (Mission Planner), adicionados de comunicação via internet remota entre os dispositivos além de um sistema de FPV IP.

4.2. Contribuições

O processo de desenvolvimento do projeto abordou áreas tal como a eletrônica que trouxe conhecimento sobre a utilização e manutenção de recursos eletrônicos e até mesmo o conhecimento de como trabalhar em segurança com energia de baixa voltagem e a utilização de sensores através do Raspberry Pi. Além de conhecimento sobre o sistema operacional Linux, pois a controladora de voo utilizada no projeto tinha como característica principal, ser baseada em Linux. Outros benefícios trazidos pela realização deste trabalho foi o melhoramento da investigação sobre os assuntos que até mesmo eram desconhecidos por si e a capacidade de acreditar na intuição sem ter medo de errar.

As informações contidas neste relatório devem servir de ponto de partida para que trabalhos futuros possam ser realizados utilizando a controladora de voo PXFMini, a GCS – Mission Planner e comunicação remota via 4G.

4.3. Trabalhos Futuros

A realização deste projeto trouxe conhecimentos relacionados que também podem ser utilizados futuramente para realização de outros projetos, como por exemplo: Pode ser construído um “Hover” veículo terrestre utilizando a mesma controladora de voo que permite múltiplos tipos de veículos e utilizar esse “Hover” para realizar missões em áreas com difíceis acessos, ou até contruir um pequeno submarino que através de uma comunicação via cabos ou que o mesmo seja um submarino de baixa profundidade e que utilize uma antena externa para receber dados de telemetria.

Lista de Referências

- [1] ARDUPILOT – 2016, [Consul. 15 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/ardupilot/index.html>
- [2] ARDUPILOT. - Arming the motors. 2016, [Consul. 20 Mai. 2018]
Disponível em WWW: http://ardupilot.org/copter/docs/arming_the_motors.html
- [3] ARDUPILOT. - Failsafe. 2016, [Consul. 20 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/copter/docs/failsafe-landing-page.html>
- [4] ARDUPILOT. - Parâmetro FRAME_CLASS. 2016, [Consul. 20 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/copter/docs/parameters.html>
- [5] ARDUPILOT. - Radio Failsafe. 2016[Consul. 20 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/copter/docs/radio-failsafe.html>
- [6] ARDUPILOT.- Testing the failsafes. 2016, [Consul. 20 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/copter/docs/flying-with-a-joystickgamepad-instead-of-rc-controller.html#testing-the-failsafes>
- [7] MISSION PLANNER, [Consul. 30 Mai. 2018]
Disponível em WWW: <http://ardupilot.org/planner/>
- [8] MISSION PLANNER.- Repositório GITHUB Mission Planner , [Consul. 30 Mai. 2018]
Disponível em WWW: <https://github.com/ArduPilot/MissionPlanner>
- [9] ARDUPILOT MEGA, [Consul. 30 Mai. 2018]
Disponível em WWW: <https://www.ardupilot.co.uk/>
- [10] GITHUB. GitHub - ArduPilot - Firmware, [Consul. 10 Mai. 2018]
Disponível em WWW: <https://github.com/ArduPilot/ardupilot>
- [11] PX4 Flight Stacks – Firmware, [Consul. 02 Jun. 2018]
Disponível em WWW: <http://px4.io/>
- [12] MAVlink. 2009
Disponível em WWW: <https://mavlink.io/en/>
- [13] DOCTORDRONE. - O Protocolo MavLink e a Programação para Drones Open, [Consul. 10 Jun. 2018]
Disponível em WWW: <http://doctordrone.com.br/drones-e-programacao-a-combinacao-perfeita/>
- [14] ARDUPILOT 2016 – PPM ENCODER, [Consul. 01 Jun. 2018]
Disponível em WWW: <http://ardupilot.org/copter/docs/common-ppm-encoder-8-channel-standalone-encoder.html>
- [15] OSCAR LIANG. – 2013, [Consul. 01 Jun. 2018]

Disponível em WWW: <https://oscarliang.com>

[16] CAVOUKIAN, A. - Privacy and drones: Unmanned aerial vehicles.

Research Group of The Office Privacy Commission of Canada, 2012, [Em linha]. p.3 [Consult. 29 Jun. 2018]

Disponível na internet: <https://www.ipc.on.ca/wp-content/uploads/Resources/pbd-drones.pdf>

[17] NERIS, L. d. O. - Um piloto automático para as aeronaves do projeto ARARA. Tese (Doutorado) — Universidade de São Paulo, 2001 [Em linha]. p.1 [Consult. 30 Jun. 2018]

Disponível na internet: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12092002-130939/pt-br.php>

[18] ARDUPILOT 2016 – ROS, [Consul. 25 Jul. 2018]

Disponível em WWW: <http://ardupilot.org/dev/docs/ros.html>

Anexos

Anexo A.

```
# Este programa permite que você teste sua ESC e moto brushless.
# Siga as instruções apresentadas no relatório do projeto, item 3.2.2.5
# Ao rodar este programa estará por sua conta e risco.

import os
import time
os.system("sudo pigpiod") #GPIO library
time.sleep(1)
import pigpio #importando GPIO library

ESC=4 #Pino em que a ESC está conectada

pi = pigpio.pi();
pi.set_servo_pulsewidth(ESC, 0)

max_value = 2000 # PWM máximo para a ESC
min_value = 700 # PWM mínimo para a ESC

print("Para o primeiro uso, digite CALIBRAR")
print("CALIBRAR OU MANUAL OU CONTROLE OU ARM OU STOP")

def manual_drive():
    print ("Opção MANUAL selecionada, entre com os valores que gostaria de
dar à ESC, entre 0 e 2000.")
    while True:
        inp = raw_input()
        if inp == "STOP":
            stop()
            break
        elif inp == "CONTROLE":
            control()
            break
        elif inp == "ARM":
            arm()
            break
        else:
            pi.set_servo_pulsewidth(ESC,inp)

def calibrate(): #Procedimento para CALIBRAR a ESC
    pi.set_servo_pulsewidth(ESC, 0)
    print("Desconecte a bateria caso esteja conectada.")
    inp = raw_input()
    if inp == '':
        pi.set_servo_pulsewidth(ESC, max_value)
        print("Conecte a bateria, escutará dois beeps, logo após um tom
gradual será escutado, então pressione ENTER.")
        inp = raw_input()
        if inp == '':
            pi.set_servo_pulsewidth(ESC, min_value)
            print ("Tom especial!")
            time.sleep(7)
            print ("Aguarde ....")
            time.sleep (5)
            pi.set_servo_pulsewidth(ESC, 0)
            time.sleep(2)
```

```

        print ("Armando ESC...")
        pi.set_servo_pulsewidth(ESC, min_value)
        time.sleep(1)
        print ("!!!!!!")
        control() # Aciona o modo de controle

def control():
    print ("Iniciando o motor, se o mesmo não estiver calibrado e armado,
    pressione 'x' para reiniciar.")
    time.sleep(1)
    speed = 1500 # Velocidade que irá iniciar, pode ser alterada para a
    que quiser, (Por sua conta e risco) o valor alterado deverá ser entre 0 e
    2000.
    print ("Controles - Pressione 'a' para diminuir a velocidade em 10 & 'd'
    para aumentar a velocidade em 10 ou 'q' para diminuir a velocidade em 100 &
    'e' para aumentar a velocidade em 100. ")
    while True:
        pi.set_servo_pulsewidth(ESC, speed)
        inp = raw_input()

        if inp == "q":
            speed -= 100
            print ("speed = %d" % speed)
        elif inp == "e":
            speed += 100
            print ("speed = %d" % speed)
        elif inp == "d":
            speed += 10
            print ("speed = %d" % speed)
        elif inp == "a":
            speed -= 10
            print ("speed = %d" % speed)
        elif inp == "STOP":
            stop()
            break
        elif inp == "MANUAL":
            manual_drive()
            break
        elif inp == "ARM":
            arm()
            break
        else:
            print ("Os botões de controles são: _ a, q, d ou e")

def arm(): #Procedimento para Armar o sistema
    print ("Conecte a Bateria então pressione ENTER.")
    inp = raw_input()
    if inp == '':
        pi.set_servo_pulsewidth(ESC, 0)
        time.sleep(1)
        pi.set_servo_pulsewidth(ESC, max_value)
        time.sleep(1)
        pi.set_servo_pulsewidth(ESC, min_value)
        time.sleep(1)
        control()

def stop():
    pi.set_servo_pulsewidth(ESC, 0)
    pi.stop()

#Inicio do programa!.
```

```
inp = raw_input()
if inp == "MANUAL":
    manual_drive()
elif inp == "CALIBRAR":
    calibrate()
elif inp == "ARM":
    arm()
elif inp == "CONTROLE":
    control()
elif inp == "STOP":
    stop()
else :
    print ("Obrigado por seguir as instruções!")
```