

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
ESCOLA DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

ARTHUR BENEMANN

**ESTAÇÃO DE CONTROLE PARA VEÍCULOS AÉREOS NÃO
TRIPULADOS**

Porto Alegre

2013

ARTHUR BENEMANN

**ESTAÇÃO DE CONTROLE PARA VEÍCULOS AÉREOS NÃO
TRIPULADOS**

Trabalho de Conclusão apresentado como
requisito parcial para a obtenção do grau de
Engenheiro Elétrico

ORIENTADOR: Prof. Dr. Carlos Eduardo Pereira

Porto Alegre

2013

AGRADECIMENTOS

A minha família pelo suporte durante o período da graduação. Gostaria de agradecer também à Universidade Federal do Rio Grande do Sul que ofereceu uma das melhores graduações de engenharia elétrica do Brasil.

Aos colegas do LASCAR pelo seu auxílio nas tarefas desenvolvidas no desenvolvimento e revisão deste trabalho.

RESUMO

Os avanços tecnológicos relacionados a motores elétricos de pequeno porte e baterias de alta densidade energética possibilitaram que os micro veículos aéreos não tripulados (VANT) ganhassem foco na comunidade científica. Este trabalho propõe a criação de uma estação de controle (Ground Control Station), para facilitar a interação com os sistemas quando operados em campo. Com o objetivo de facilitar o planejamento de missões a serem executadas pelo VANT, e acompanhar os dados de voo enquanto esta é realizada.

Palavras-chaves: Engenharia Elétrica. VANT. GCS.

ABSTRACT

Technological advancements related to small electrical engines, and high-density batteries, made micro unmanned aerial vehicles (UAV) earn a focus on the scientific community. This paper proposes the creation of a ground control station (GCS) to facilitate interaction with the systems when operated in the field. In order to facilitate the planning of missions to be performed by UAVs, and monitor flight data while it is executed.

Keywords: Electrical Engineering. UAV. GCS.

SUMÁRIO

1 INTRODUÇÃO.....	10
2 REVISÃO DE CONCEITOS	11
1.1.1 3.1 Veículos aéreos não tripulados.....	11
1.Plataforma ArduCopter.....	11
1.1.2 3.2 Estação de controle de solo.....	12
1.Dados de voo.....	12
2.Controle.....	12
3.Planejamento.....	12
1.1.3 3.3 Redes sem fio.....	13
1.Xbee.....	13
2.Hope-RF.....	13
3.Bluetooth.....	14
4.WiFi.....	14
1.1.4 3.4 Plataforma Android.....	14
1.1.5 3.5 Protocolo de comunicação.....	15
1.Pacotes MAVLink.....	16
2.CRC.....	16
3.Mensagens.....	17
4.Transferência de missões.....	17
4 IMPLEMENTAÇÃO.....	19
1.1.6 4.1 Hardware.....	19
1.MAVBridge.....	19
1.1.7 4.2 Software.....	23
1.Arquitetura.....	23
2.Interface Homem Maquina.....	23
5 RESULTADOS.....	27
6 PROPOSTAS DE MELHORIAS.....	28
7 CONCLUSÕES.....	29

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

TABELA 1 ESTRUTURA DE UM PACOTE MAVLINK V1.0.....	17
--	-----------

LISTA DE ABREVIATURAS

UAV: *Unmanned Aircraft Vehicle*

VANT: *Veículo Aéreo Não Tripulado*

GCS: *Ground Control Station*

MAVLink: *Micro Air Vehicle Link*

ASCII: *American Standard Code for Information Interchange*

CRC: *Cyclic Redundancy Check*

LSB: *Least Significant Bits*

MSB: *Most Significant Bits*

XML: *Extensible Markup Languages*

LASCAR: Laboratório de Sistemas de Controle, Automação e Robótica

VARP: Veículo Aéreo Remotamente Pilotado

IHM: Interface Homem Máquina

2 INTRODUÇÃO

Nos últimos anos, o avanço de diversas tecnologias relacionadas a micro veículos aéreos não tripulados (VANT) fizeram com que o seu potencial para uso civil fosse elevado, considerando que a maior parcela de uso, atualmente, encontra-se em fins militares. Como são sistemas versáteis, as possibilidades de uso variam incluindo, principalmente, monitoramento de linhas de transmissão, apoio à segurança pública e agricultura de precisão.

Por serem sistemas complexos, existe grande dificuldade em sua operação. Este trabalho propõe a criação de uma estação de controle (*Ground Control Station*, GCS), para facilitar a interação com os VANTs quando operados em campo, facilitando o planejamento e execução de missões do sistema, além de acompanhar os dados de voo de forma simultânea. A implementação do projeto terá como base a plataforma *Android*, através da elaboração de um aplicativo.

Posteriormente, serão apresentados de forma mais detalhada o problema e a elaboração da solução, incluindo o protocolo de comunicação utilizado, a implementação do sistema e os resultados de experimentos práticos realizados para a validação do projeto.

3 REVISÃO DE CONCEITOS

3.1 VEÍCULOS AÉREOS NÃO TRIPULADOS

Um veículo aéreo não tripulado ou veículo aéreo remotamente pilotado (VARP) é todo e qualquer tipo de aeronave que não necessita de pilotos embarcados para ser guiada. Esses aviões são controlados a distância por meios eletrônicos e computacionais, sob a supervisão e governo humanos ou sem a sua intervenção.

A terminologia VANT se refere não somente a aeronave, mas a todos os equipamentos de suporte utilizados no sistema, incluindo sensores, microcontroladores, *software*, estações de controle de solo, *hardware* de comunicação. (BEARD;McLAIN, 2012)

Pelo fato de o veículo não ser tripulado todos os esforços de desenvolvimento de uma Interface Homem Máquina (IHM) são deslocados do interior da aeronave para a estação de controle de solo.

1. Plataforma ArduCopter

FIGURA 1. 3DROBOTICS ARDUCOPTER Y6



Fonte: 3DRobotics, 2013

3.2 ESTAÇÃO DE CONTROLE DE SOLO

A estação de controle é uma parte indispensável em um sistema de VANT. O computador é núcleo de uma GCS, e o *software* do computador, ou *software* da GCS, é o seu elemento mais crítico. (KANG;YUAN, 2009)

Os objetivos de uma GCS são divididos em visualização de dados de voo, controle da aeronave, planejamento de missões autônomas.

1. Dados de voo

A percepção da situação, isto é, a habilidade de perceber e monitorar o ambiente nas proximidades da aeronave, é de extrema importância na operação de veículos aéreos. A visualização dos dados de voo como, por exemplo, altitude atual, velocidade do ar, velocidade de solo, orientação espacial, posição espacial, localização geográfica, modo de controle/voo, altitude e velocidade desejados, situação da missão, consumo de energia, é necessária para a construção desta percepção de situação pelo operador do sistema na GCS, dado que não há piloto embarcado.

2. Controle

O controle da aeronave pode ser realizado de forma direta, modificando diretamente a orientação da aeronave, ou de forma indireta, através de modificações do plano de voo (o qual é seguido de forma autônoma pelo VANT).

3. Planejamento

Os VANTs são capazes de realizar, de forma autônoma, “missões”, isto é, planos de voo definidos pelo operador. Uma forma de definir uma missão, ou seja, planejar um

voo, é através da definição de *waypoints* - pontos tridimensionais os quais devem ser atingidos pela aeronave.

Existem diversos tipos de *waypoints*, dependendo do *software* de piloto automático utilizado, como por exemplo *waypoints* pontuais, de decolagem, de pouso, de espera, de retorno à casa (ponto de origem do voo, definido pelo operador).

3.3 REDES SEM FIO

1. XBEE

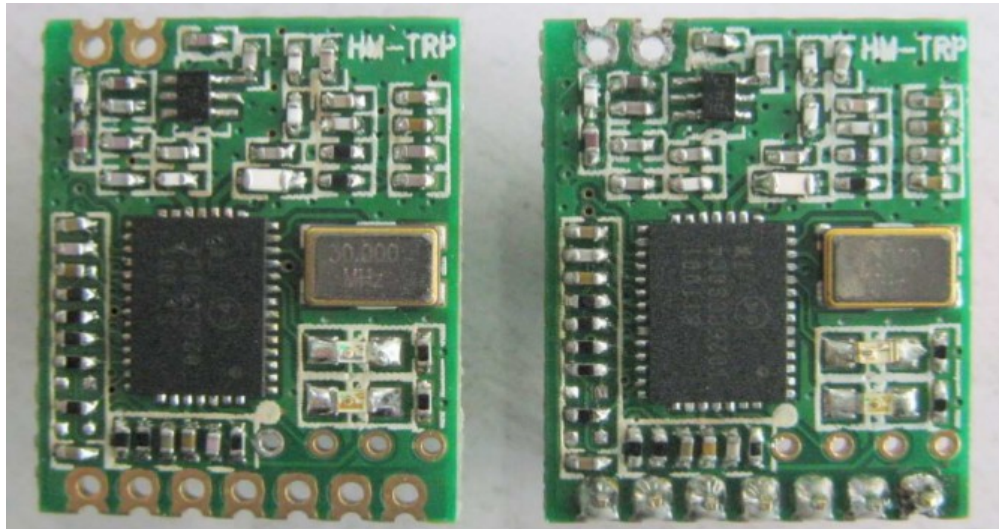
Figura 2. MODULO XBEE-XSC



FONTE: DIGI, 2013

2. HOPE-RF

FIGURA 3. MODULO HM-TRP



FONTE: HOPE MICROELECTRONICS, 2013

3. BLUETOOTH

FIGURA 4. MODULO RN-42



FONTE: MICROCHIP, 2013

4. WiFi

Figura 5. Modulo RN-171



3.4 PLATAFORMA ANDROID

Android é um sistema operacional, baseado no núcleo do Linux, mas destinado a dispositivos móveis. Desenvolvido pela Open Handset Alliance, o sistema teve grande popularização desde seu ano de lançamento público, 2008, dado que incorpora uma combinação de facilidade de uso com disponibilidade gratuita. Juntamente com o sistema operacional, observou-se o crescimento da utilização dos aplicativos, isto é, softwares destinados a uma tarefa específica, sendo, assim, “ferramentas” do sistema.

Majoritariamente, o desenvolvimento de aplicativos se dá através de programação em Java, uma linguagem de programação orientada ao objeto, com sintaxe similar às linguagens C/C++. Para o projeto descrito neste documento, a linguagem de programação utilizada foi Java.

3.5 PROTOCOLO DE COMUNICAÇÃO

O protocolo de comunicação utilizado foi o MAVLink v1.0. Este foi escolhido pois existem um grande suporte em VANTs já existentes, como: Ardupilot, PX4FMU, SmartAP, MatrixPilot. Isso possibilita a fácil integração com todos estes sistemas.

O protocolo foi projetado por Lorenz Meier [], e liberado para uso sobre a licença LGPL em 2009. O uso mais comum é para a comunicação de um VANT para a sua GCS, e na intercomunicação de subsistemas do veículo. Através deste protocolo, dados do VANT, como posição e orientação, são transferidos.

A comunicação dos dados é realizada através de pacotes, os quais contém uma estrutura parecida com o protocolo CAN. Cada pacote contém uma mensagem de tamanho variável, cujos os dados depende da mensagem a ser transmitida. A ordem dos dados contidos em cada mensagem é definida em um arquivo XML, que faz parte do protocolo. VANTs de fabricantes diferentes podem estender o conjunto de mensagens padrão, oferecendo uma certa flexibilidade sobre o custo de se tornar incompatível com as GCS disponíveis.

1. Pacotes MAVLink

A anatomia dos pacotes é apresentada na tabela 1, esta estrutura é baseada nos protocolos CAN e SAE AS-4. O foco do desenvolvimento deste protocolo foi a velocidade de transmissão e segurança. Com apenas seis bytes extras impostos nas mensagens é possível realizar a verificação dos dados da mensagem e detectar a perda de pacotes.

Tabela 1 Estrutura de um pacote MAVLink v1.0

Nome do Campo	Posição (Bytes)	Propósito
Início de pacote	0	Denota o início de um pacote (v1.0 0xFE)
Tamanho da mensagem	1	Tamanho em bytes da mensagem seguinte
Número de sequência	2	Número do pacote na sequência
ID do Sistema	3	Identificação do sistema que está enviando este pacote
ID do Componente	4	Identificação do componente que está enviando este pacote
ID da mensagem	5	Define o que a mensagem contém, e como ela deve ser decodificada
Mensagem	6 a (n+6)	Dados dessa mensagem, dependente do ID da mensagem
CRC	(n+7) a (n+8)	Check-sum do pacote, para detecção de erros de transmissão

2. CRC

Para garantir a integridade das mensagens um campo contendo o *checksum* é adicionado ao final de cada pacote. Na recepção o *checksum* é recalculado e comparado com o recebido, caso ocorra uma diferença a mensagem é descartada.

Outra função do campo de CRC é garantir que o transmissor e receptor concordam com o tipo de dado sendo transferido. Isso é feito adicionando uma semente ao gerador de checksum baseada em um hash da definição da mensagem. Normalmente um vetor estático é gerado com o valor de hash correspondente a cada mensagem, diminuindo o custo computacional para um sistema real-time.

3. Mensagens

Os dados transmitidos são contidos em mensagens estruturadas, as quais posteriormente são encapsuladas pelos pacotes já descritos na seção anterior. Cada mensagem é identificada pelo campo ID de mensagem do pacote, e tem os seus dados armazenados na seção de carga do pacote.

A decodificação/codificação de cada mensagem é feita de acordo com um arquivo XML definido pelo protocolo. Este arquivo contém a ordem dos campos a serem extraídos, o tamanho de cada campo e o tipo de dado armazenado.

Na ... abaixo é apresentada a definição de uma mensagem típica do protocolo, extraída do documento XML.

```
<message id="24" name="GPS_RAW_INT">
  <description>The global position, as returned by the Global Positioning
  System (GPS). This is NOT the global position estimate of the system, but rather a
  RAW sensor value. See message GLOBAL_POSITION for the global position estimate.
  Coordinate frame is right-handed, Z-axis up (GPS frame).</description>
  <field type="uint64_t" name="time_usec">Timestamp (microseconds since UNIX
  epoch or microseconds since system boot)</field>
  <field type="uint8_t" name="fix_type">0-1: no fix, 2: 2D fix, 3: 3D fix.
  Some applications will not use the value of this field unless it is at least two,
  so always correctly fill in the fix.</field>
  <field type="int32_t" name="lat">Latitude (WGS84), in degrees * 1E7</field>
  <field type="int32_t" name="lon">Longitude (WGS84), in degrees *
  1E7</field>
```

```
<field type="int32_t" name="alt">Altitude (WGS84), in meters * 1000
(positive for up)</field>
<field type="uint16_t" name="eph">GPS HDOP horizontal dilution of position
in cm (m*100). If unknown, set to: UINT16_MAX</field>
<field type="uint16_t" name="epv">GPS VDOP horizontal dilution of position
in cm (m*100). If unknown, set to: UINT16_MAX</field>
<field type="uint16_t" name="vel">GPS ground speed (m/s * 100). If unknown,
set to: UINT16_MAX</field>
<field type="uint16_t" name="cog">Course over ground (NOT heading, but
direction of movement) in degrees * 100, 0.0..359.99 degrees. If unknown, set to:
UINT16_MAX</field>
<field type="uint8_t" name="satellites_visible">Number of satellites
visible. If unknown, set to 255</field>
</message>
```

4. Transferência de missões

A transmissão de uma missão é realizada através de diversas transações, cada transação é validada podendo ser requerido pelo receptor uma retransmissão. As transferências podem ser realizadas no sentido GCS para VANT, e vice-versa.

4 IMPLEMENTAÇÃO

4.1 HARDWARE

1. Conexão direta via USB

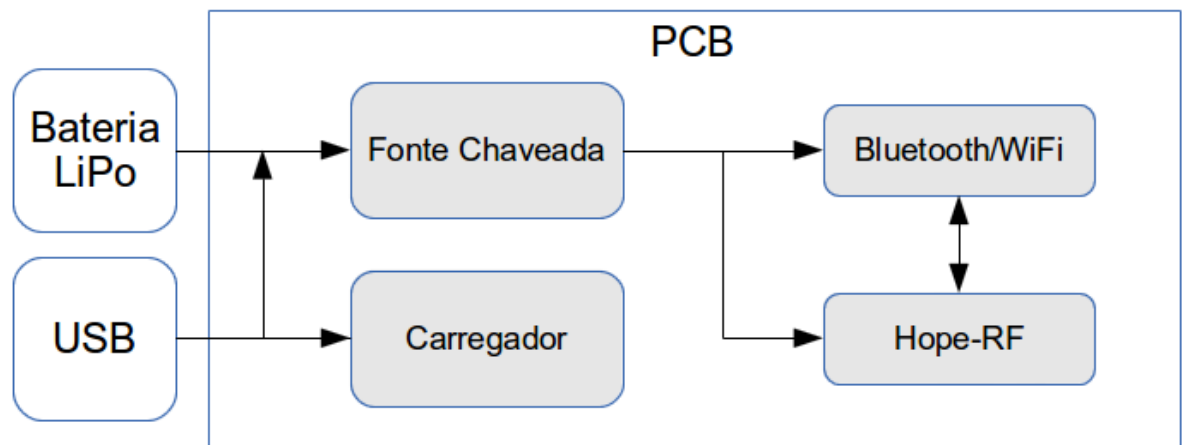
Figura 6. Modulo de telemetria 3DRobotics



Fonte: 3DRobotics, 2013

1. MAVBridge

Figura 7. MAVBridge – Diagrama de Blocos



1 Esquemático

Figura 8. MAVBridge – Modulo Bluetooth/Wifi

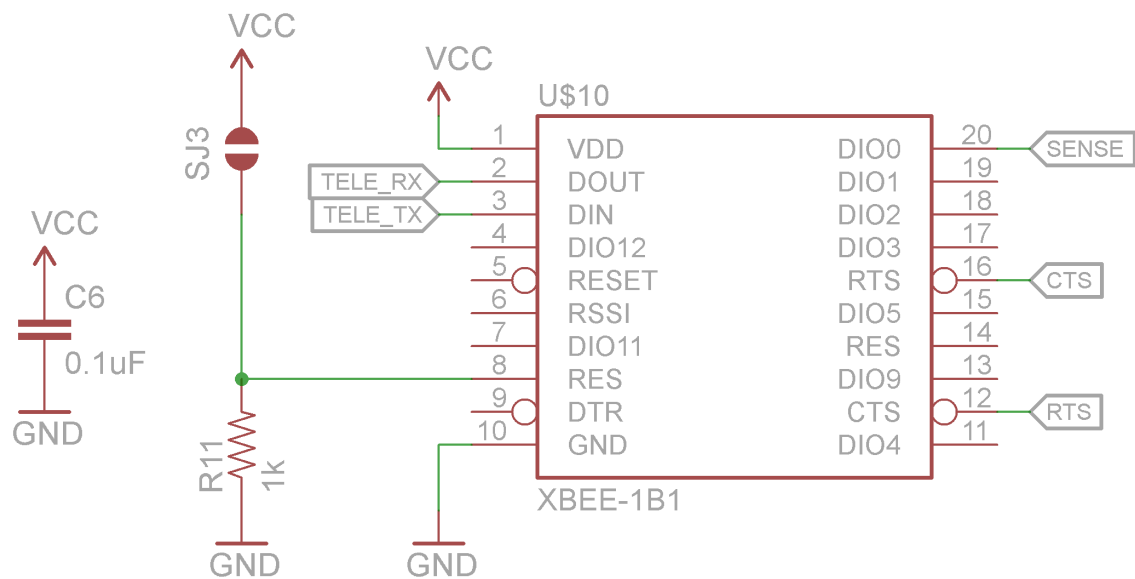


Figura 9. MAVBridge – Modulo HopeRF

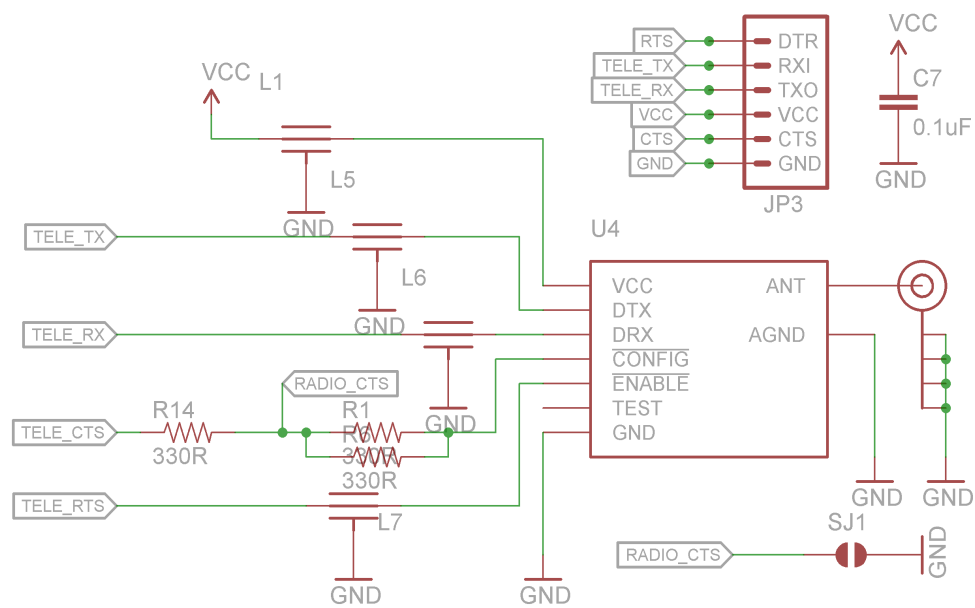


Figura 10. MAVBridge – Carregador de bateria LiPo

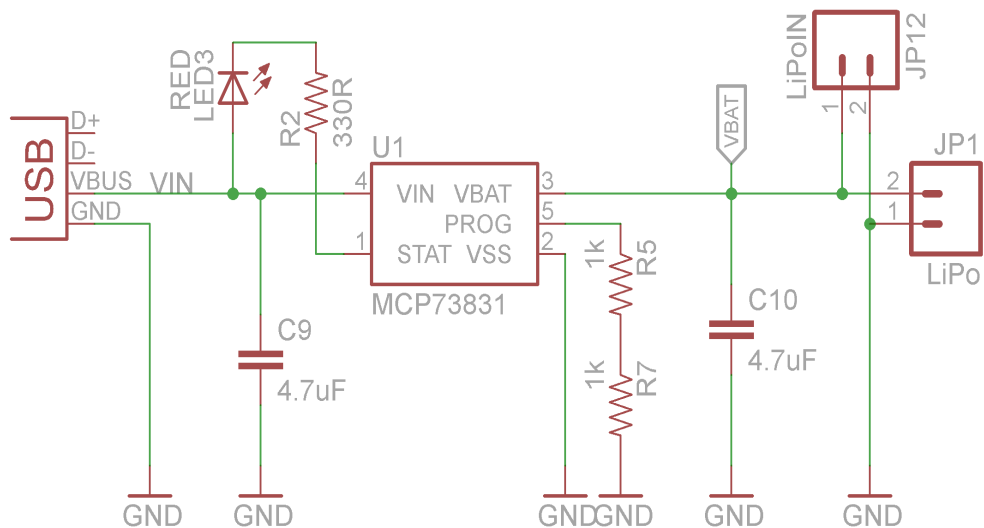


Figura 11. MAVBridge – Fonte Chaveada

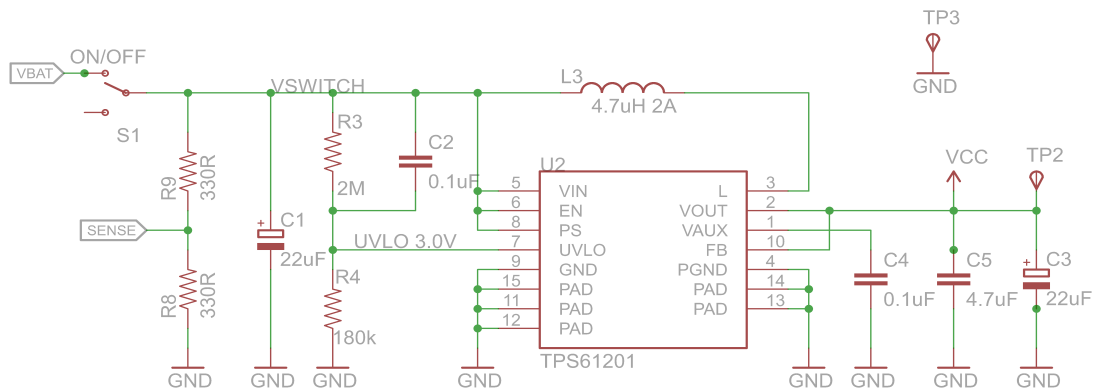
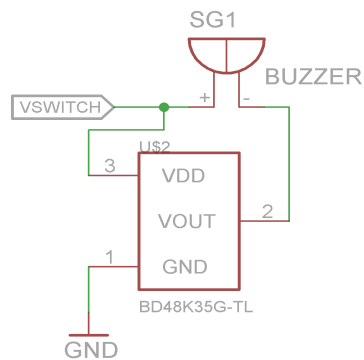


Figura 12. MAVBridge – Alarme de bateria baixa



2 Layout

Figura 13. MAVBridge – Layout dos componentes

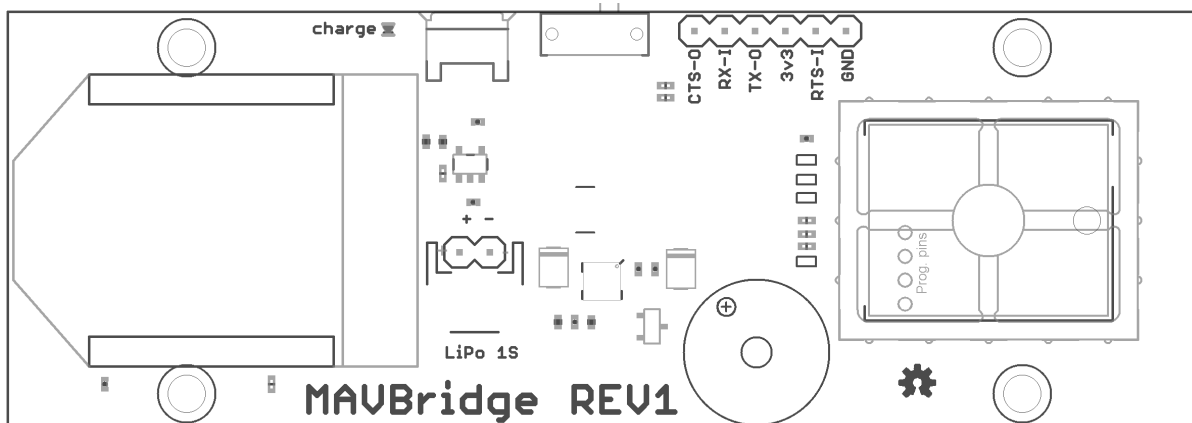


Figura 14. MAVBridge – Layout da camada superior

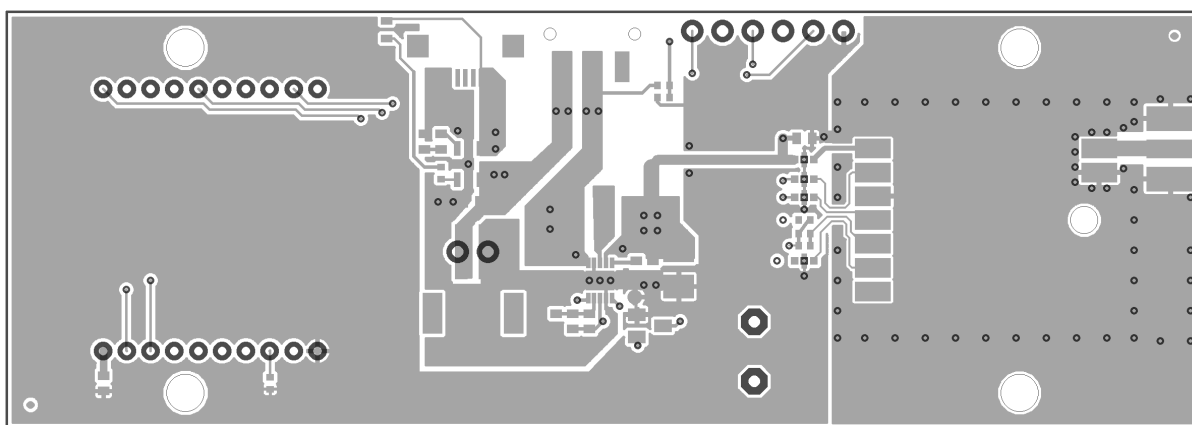
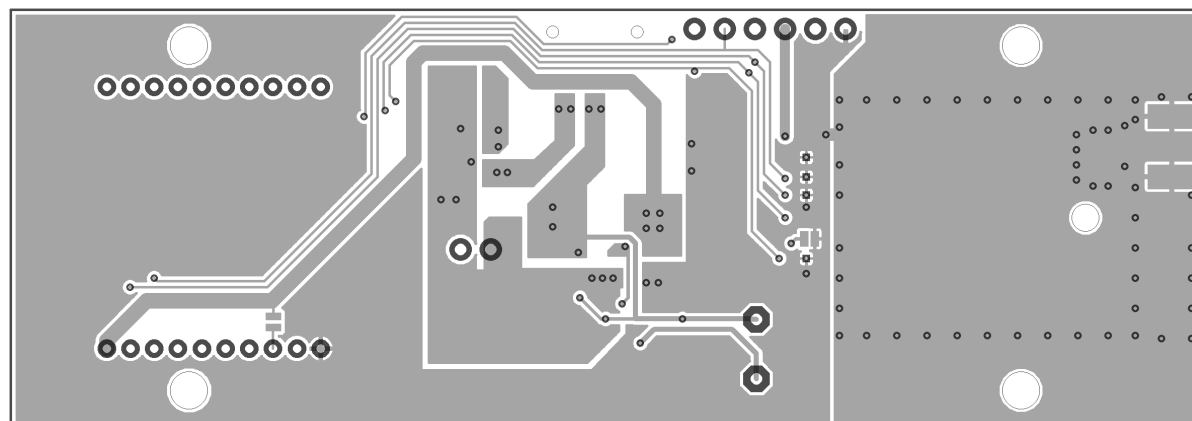


Figura 15. MAVBridge – Layout da camada inferior



4.2 SOFTWARE

1. Arquitetura

3 Modelo do veiculo

4 Serviços de *background*

5 Meios de Comunicação

USB

Bluetooth

TCP

UDP

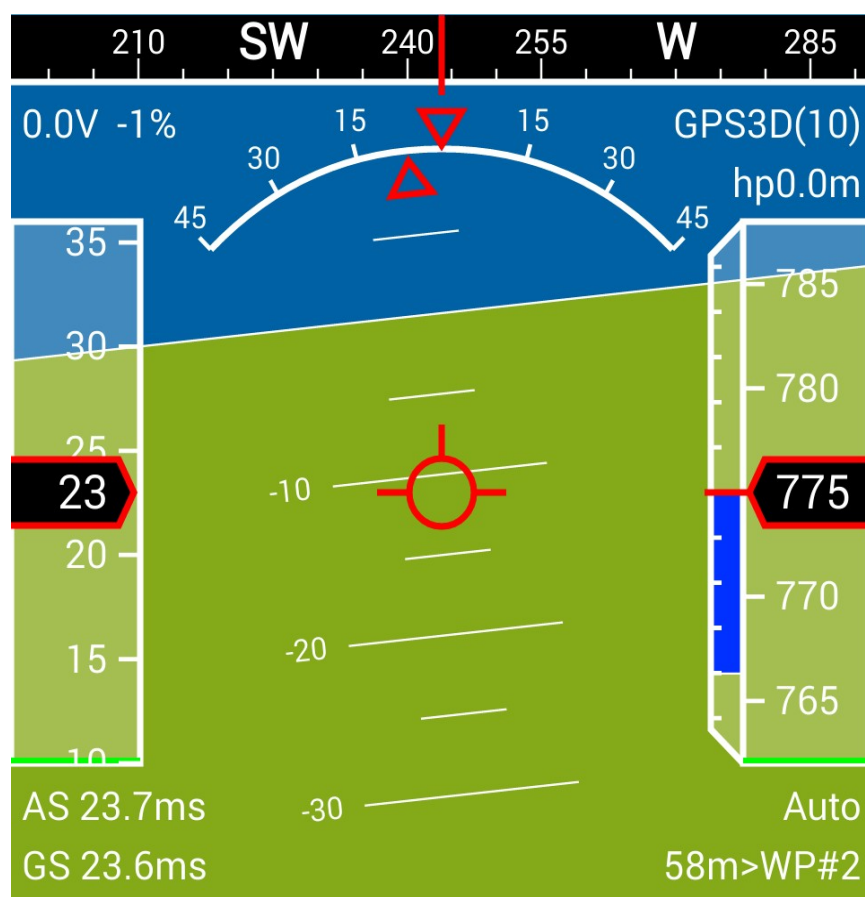
2. Interface Homem Maquina

1 Informações de voo

Figura 16. Tela de Informações de voo



Figura 17. Detalhes do Heads Up Display



2 Planejamento de missões

Figura 18. Tela de Planejamento de missões



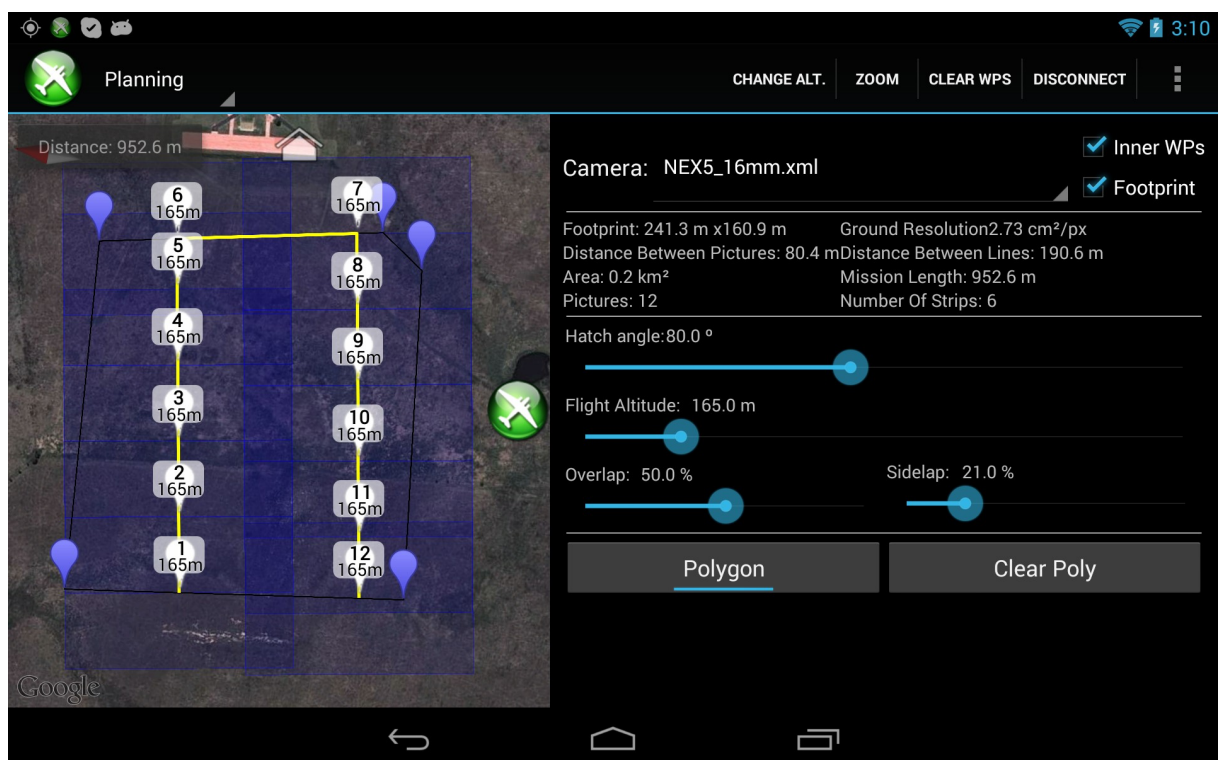
3 Planejamento aerofotogramétrico

Figura 19. Voo fotogramétrico



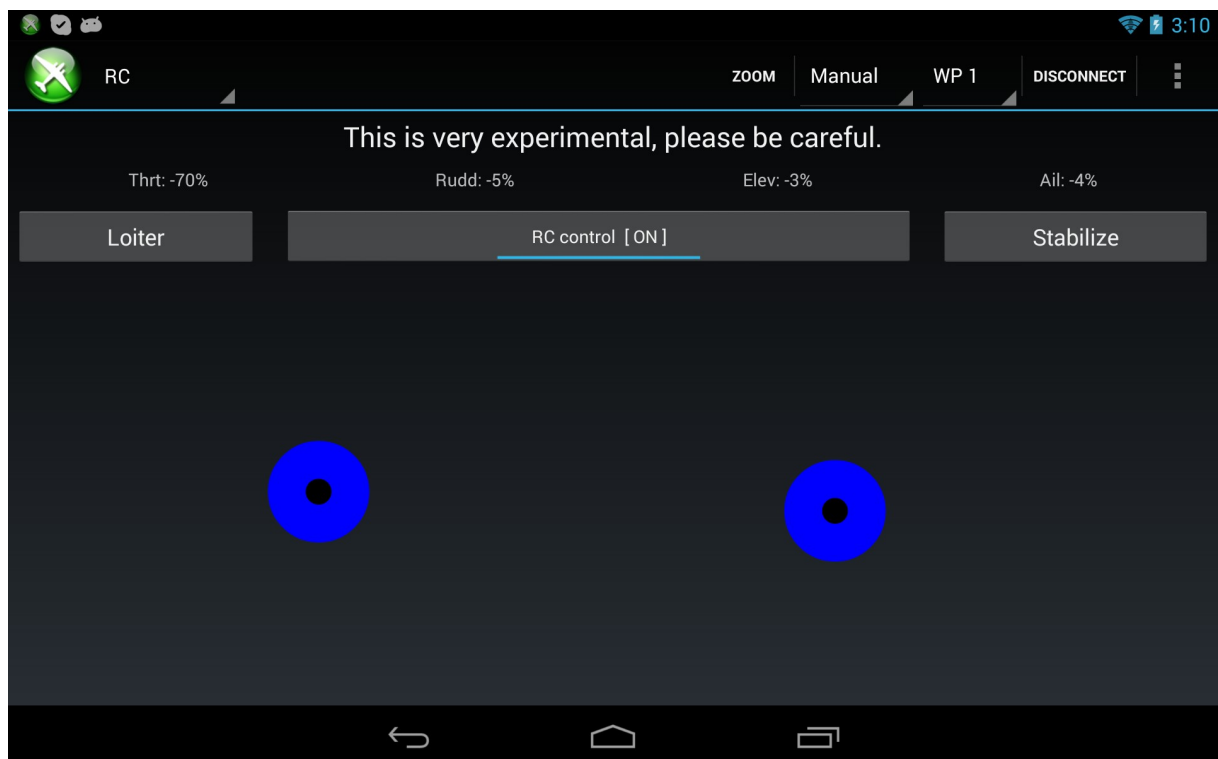
Fonte: UFF, 2013

Figura 20. Tela de Planejamento aerofotogramétrico



4 Controle

Figura 21. Tela de Controle



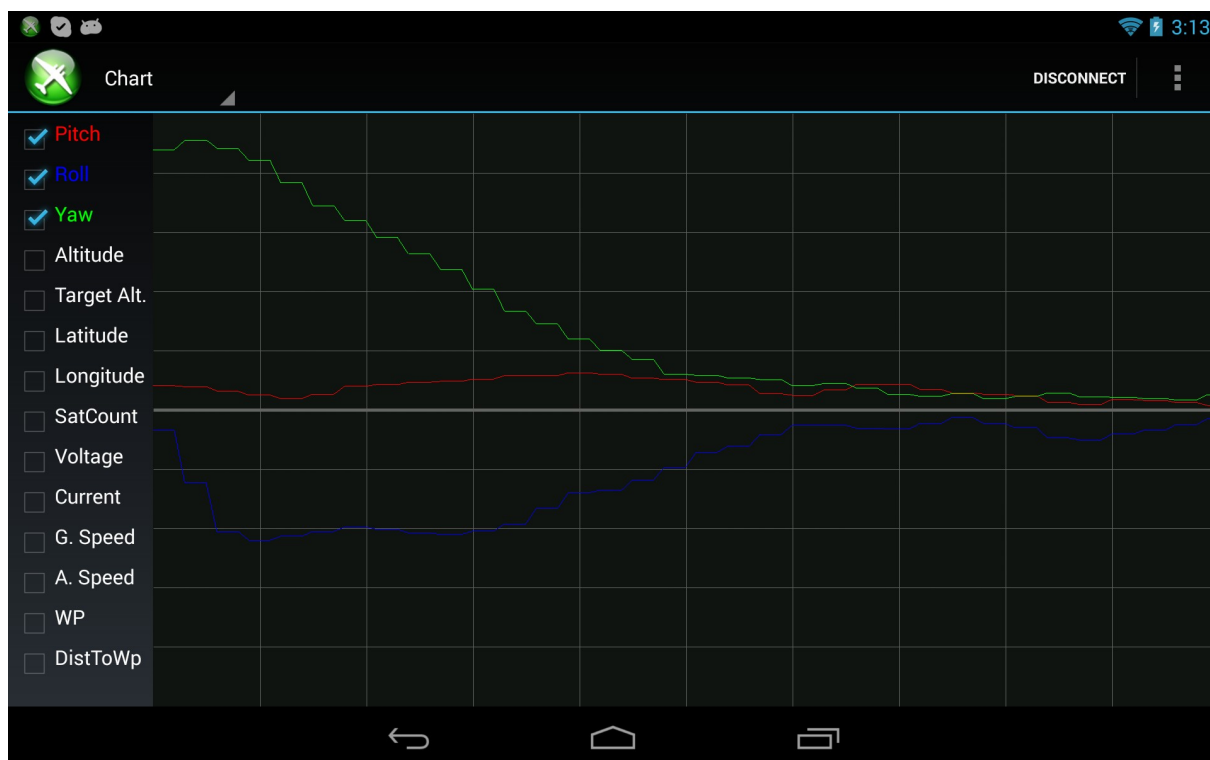
5 Parâmetros

Figura 22. Tela de Parâmetros

Parameter Name	Description	Value
ACRO_PITCH_RATE	ACRO mode pitch rate (degrees/second)	180
ACRO_ROLL_RATE	ACRO mode roll rate (degrees/second)	180
AHRS_COMP_BETA	AHRS Velocity Complimentary Filter Beta Coefficient	0.1
AHRS_GPS_GAIN	AHRS GPS gain	1
AHRS_GPS_MINSATS	AHRS GPS Minimum satellites	6
AHRS_GPS_USE	AHRS use GPS for navigation	1
AHRS_ORIENTATION	Board Orientation	0
AHRS_RP_P	AHRS RP_P	0.3
AHRS_TRIM_X	AHRS Trim Roll (Radians)	0
AHRS_TRIM_Y	AHRS Trim Pitch (Radians)	0
AHRS_TRIM_Z	AHRS Trim Yaw (Radians)	0
AHRS_WIND_MAX	Maximum wind (m/s)	0

6 Gráficos

Figura 23. Tela de Gráficos



5 RESULTADOS

6 PROPOSTAS DE MELHORIAS

7 CONCLUSÕES

REFERÊNCIAS

BEARD, R. W.; McLAIN, T. W. **Small Unmanned Aircraft**. First edition. Princeton University Press, 2012.

KANG, W.; YUAN, M. **Software Design for Mini-type Ground Control Station of UAV**. School of Automation Science and Electrical Engineering, Beijing University of Aeronautics & Astronautics, 2009

Datasheet. **HM-TRP Series 100mW Transceiver modules V1.0**. HOPE MICROELECTRONICS CO., LTD, 2006

Edison Pignaton de Freitas. Cooperative Context-Aware Setup and Performance of Surveillance Missions Using Static and Mobile Wireless Sensor Networks. PhD thesis, 2011.

Athos Alexandre Lima Fontanari, Flavio Rech, and Carlos Eduardo. Sistema de planejamento e controle de missao de um veiculo aereo nao-tripulado. 2011.

Carlos Eduardo Pereira. Integracao de redes de sensores sem fio com veiculos aereos nao-tripulados (vants). Premio Santander 2010, 2010.

3DRobotics. <http://www.3drobotics.com>

Skydrones. <http://www.skydrones.com.br>