

Universidade Federal de São Carlos
Pró-Reitoria de Pesquisa
Coordenadoria de Iniciação Científica e Tecnológica

Relatório parcial 2016

Título do Projeto de Pesquisa:
VANT SEMIAUTÔNOMO UTILIZANDO ARDUINO

Nome do Aluno:
CAIO CESAR ALMEIDA PEGORARO

Nome do Curso de Graduação:
ENGENHARIA DE COMPUTAÇÃO

Nome do Orientador:
EDILSON REIS RODRIGUES KATO

Departamento/Centro:
DEPARTAMENTO DE COMPUTAÇÃO

São Carlos / 2016

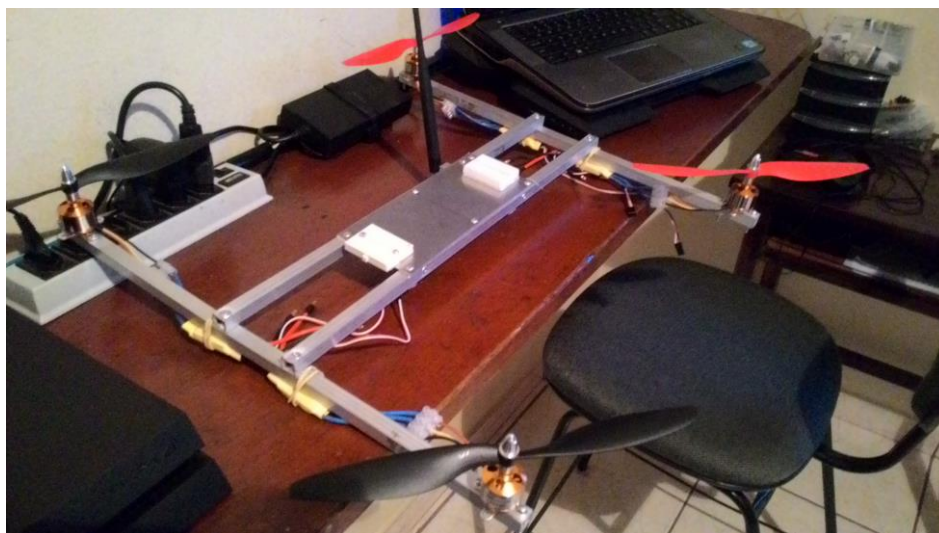
1 Resultados obtidos <parcialmente>

A construção da estrutura (em alumínio) seguiu como esperado, apesar de o projeto esquemático ter sido feito em papel (após a execução de algumas atividades mais críticas <apresentadas nos itens seguintes> será feito a conversão da estrutura completa para sua versão digital no software SketchUp 2015).

A fixação dos componentes e motores foi realizada com parafusos e *velcro* (para sustentação dos 4 ESC's e da bateria na parte inferior do Vant).

Os conectores tiveram que ser estendidos utilizando solda nas conexões fixas e conector bullet para conexões móveis (dos ESC's a bateria), todas as conexões foram isoladas utilizando um termo retrátil.

Como esperado após o todo processo de análise, o conjunto motores, esc's, bateria e hélices foram capazes de erguer a estrutura com grande facilidade (os motores foram controlados manualmente por um potenciômetro, por questões de segurança, uma vez que o módulo de controle autônomo ainda não foi implementado).

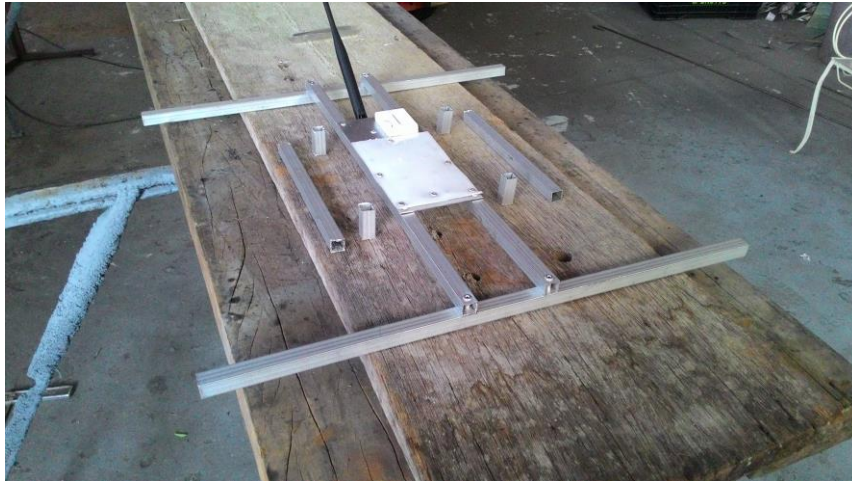




2 Dificuldades encontradas

Nas etapas que envolveram a aquisição de componentes a dificuldade foi aliar custo com eficiência e eficácia, um dos problemas de um Vant é fazer com que a somatória dos pesos não exceda um teto limite, na mesma medida em que certos componentes são essenciais e os motores devem ser capazes de suportar a carga.

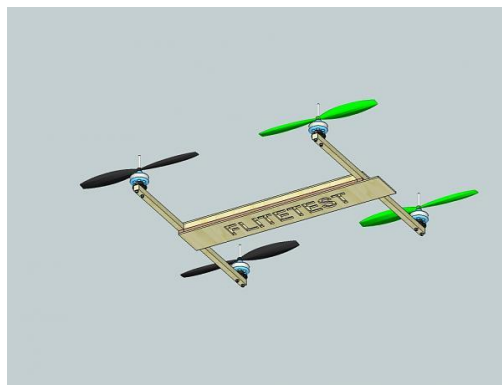
A utilização do primeiro sensor de rádio frequência (emissor/receptor) não foi satisfatória, devido à baixa eficácia (sim, algumas operações não eram concluídas e não existia forma de garantir isso) e a incompatibilidade com o chip Attiny85 (o qual seria o operador de comunicação).



3 Cronograma de Trabalho **Atualizado**

A1. Finalização da estrutura física

A fixação das barras e da chapa central de alumínio utilizou parafusos e arrebitos, toda estrutura foi completada atingindo um peso total de 418 gramas. Foi adotado um modelo de arquitetura conhecido como H-Quad (justamente porque o formato se assemelha à letra “H”), isso possibilitou a utilização da estrutura central como apoio ao micro controlador e aos demais sensores.



h-quadr (<http://forum.flitetest.com/attachment.php?attachmentid=5151&stc=1&thumb=1&d=1342134855>)

A2. Seleção dos motores e ESC.

A adoção de motores *brushless* (motor de corrente contínua sem escovas) se deve a melhor vida útil devido a um menor desgaste em contra partida do maior custo. O modelo de 1000kv (rpm/volts implicando em $1000 \times 11,1v = 11.100rpm$) em conjunto com ESC de 30A (será referenciado na parte da bateria) e as hélices de 10 polegadas foram suficientes para prover empuxo suficiente para erguer a estrutura.

A3. Bateria

A bateria selecionada é uma do tipo *Lipo* (explicar o que é?), modelo 3s (3 células de $3.7v = 11.1v$) com capacidade de 2200mah (na prática em pleno funcionamento, provendo energia para 5~10 minutos de operação) com capacidade de alta descarga (40c), no caso poderia prover até 22A de corrente para cada motor (que será, na prática, limitada a 12A).

A4. Implementação da interface de comunicação com a base receptora

Essa etapa sofreu alguns ajustes e precisa ser finalizada, inicialmente era previsto a utilização de um módulo de comunicação via rádio (FS1000A 315 MHz) no qual o receptor era processado por um chip Attiny85 que se comunicaria com um micro controlador arduino mega/uno; Alguns aspectos técnicos forçaram uma mudança: primeiro, a biblioteca de comunicação convencional não é compatível com a do micro chip, segundo, o sensor não se mostrou eficiente o suficiente para ser o principal meio de comunicação.

Solução encontrada: substituição por dois *transceiver wifi* (modulo nrf24l01), no momento, funcionaram como esperado (após adaptações e correções na biblioteca e nas ligações), apenas precisando construir

uma tabela de códigos para interpretação e execução dos comandos.

A5. Interface de comunicação do PC com arduino

Está sendo utilizado a plataforma Windows para desenvolvimento de uma interface em C# para comunicação serial entre o PC e o arduino emissor, nessa etapa falta a definição das grandezas que são de interesse para serem obtidas e exibidas, tal como a padronização dos códigos de controle que serão enviados ao receptor.

A6. Definição dos microcontroladores

Inicialmente esperava trabalhar com unidades iguais, mas atualmente o projeto é desenvolvido com um total de três modelos de arduino (uno, mega e due), o último em questão adquirido justamente para ser acoplado ao Vant como controlador principal dos motores, devido a sua capacidade de *multitasking*, enquanto o arduino mega/uno seriam utilizados na base do emissor (normalmente seria o uno, mas ainda há a possibilidade de este ser integrado ao Vant como uma fonte de execução auxiliar).

A7. Testes de campo

Os testes realizados se enquadram em uma escala de execução reduzida, isto é, serviram pra certificar que os componentes cumpriram com as suas especificações técnicas e operacionais (hélices, motores, ESC, bateria, etc), mostrando que o peso e o empuxo estão em uma relação que permite a decolagem.

A8. Análise dos resultados

A única etapa que precisou ser prolongada no cronograma se refere a comunicação sem fio, até por conta da troca dos sensores e micro controladores, as demais seguiram como planejado e estão de acordo

com as especificações de desempenho e eficiência iniciais.

Uma etapa não prevista inicialmente deve ser implementada: utilizando um sensor de ultra som voltado para baixo (eixo vertical Z negativo) para compor uma função de pouso controlado e automático.

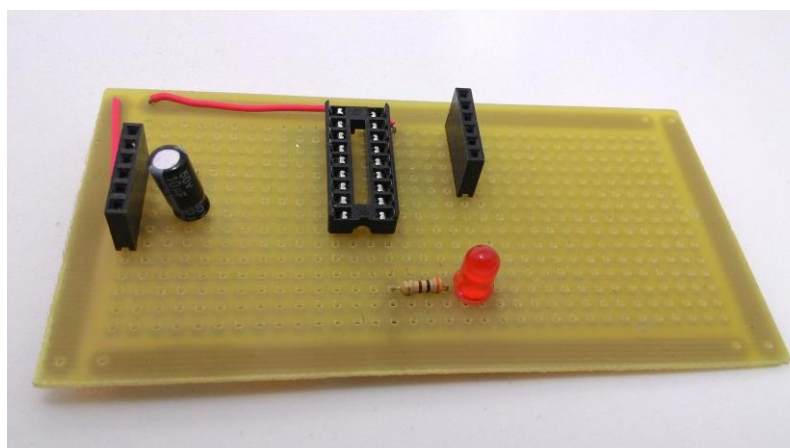
A9. Próximas etapas

Finalizando o módulo de comunicação sem fio, a etapa seguinte se concentrará na interface C# (já que nesse caso teremos que enviar um sinal via USB que será enviado pelo módulo ao Vant, finalizando o setor de comunicação).

A execução vai ocorrer em paralelo a implementação do controle autônomo baseado em PID (já especificado).

Os módulos acelerômetro e giroscópio já foram programados para fornecer a inclinação nos eixos horizontais X e Y, dados de entrada para o controle de equilíbrio.

Uma etapa complementar ao projeto seria integrar o chip Attiny85 como um controlador secundário (dos LED's e do Buzzer <já implementado>), uma vez que esses não são setores prioritários para implementar no controlador principal e já foi construído todo mecanismo de programação para o chip.



Cronograma atualizado 2016.

Atividades	Meses (2015 e 2016)												
	8	9	10	11	12	1	2	3	4	5	6	7	
Levantamento bibliográfico e planejamento	X	X	X	X	X	X	X	X	X	X	X		
Seleção de componentes a serem utilizados	X	X	X	X				X					
Desenvolvimento do algoritmo do micro controlador			X	X	X	X	X	X	X	X	X		
Implementação da interface de comunicação com a base receptora					X	X	X	X	X				
Estabelecimento da comunicação entre o VANT e o controlador remoto			X	X	X			X	X	X	X		
Desenvolvimento da arquitetura física do VANT			X	X	X	X	X	X	X	X	X		
Testes de campo				X	X	X	X	X	X	X	X		
Análise dos resultados									X	X	X		
Elaboração da documentação final											X	X	

4 Referências Bibliográficas

- [1] MARGOLIS, Michael. **Arduino Cookbook**.ed. O'Reilly Media, 2011.
- [2] McROBERTS, Michel, **Arduino Básico**, Ed. Novatec, 2011.
- [3] MONK, Simon. **Programação com Arduino: Começando com Sketches**, Ed. Bookan, 2012.
- [4] Arduino, <<http://www.arduino.cc/>>, acesso em acesso em 25 de Abril de 2015.
- [5] “Veículos aéreos não tripulados prometem revolucionar mercado de geotecnologia”, Massa Cinzenta, <<http://www.cimentoitambe.com.br/veiculos-aereos-nao-tripulados-prometem-revolucionar-mercado-de-geotecnologia/>>, acesso em 26 de Abril de 2015.

[6] VANTs e RPA, <<http://www.ebc.com.br/tecnologia/2015/02/drones-vants-ou-rpas-entenda-mais-sobre-essas-aeronaves-nao-tripuladas#1>>, acesso em 26 de Abril de 2015.

[7] Visual Studio, <<https://www.visualstudio.com/>>, acessado em 26 de Abril de 2015.

[8] Bluetooth, <<http://www.bluetooth.com/Pages/Bluetooth-Home.aspx>>, acessado em 26 de Abril de 2015.

[9] FPV, <<http://fpvbrasil.com.br/page/o-que-e>>, acessado em 26 de Abril de 2015.

[10] ANAC, <<http://www.anac.gov.br/>>, acessado em 27 de Abril de 2015.

[11] WIRELESS, <http://pt.wikipedia.org/wiki/Rede_sem_fio>, acessado em 27 de Abril de 2015.

Referências complementadas em 2016:

[12] “Quadcopter Design”, <<http://quadcopterdesign.blogspot.com.br/>>, acessado em 10 de março de 2016.

[13] “RF24 Driver release”, Maniacal Bits, <<http://maniacalbits.blogspot.com.br/2013/04/new-rf24-driver-release-fork.html>>, acessado em 10 de março de 2016.

[14] “Giroscópio GY-521”, O mundo da programação, <<http://omundodaprogramacao.blogspot.com.br/2015/08/giroscopio-gy-521.html>>, acessado em 10 de março de 2016.

[15] “Controle PID em sistemas embarcados”, Embarcados, <<http://www.embarcados.com.br/controle-pid-em-sistemas-embarcados/>>, acessado em 10 de março de 2016.

[16] “Quadcopter PID Explained and Tuning”, OscarLiang, <<http://blog.oscarliang.net/quadcopter-pid-explained-tuning/>>, acessado em 10 de março de 2016.

[17] “Estabilizador de voo”, Drones personalizados, <<http://dronespersonalizados.blogspot.com.br/2013/11/estabilizador-de-voo.html>>, acessado em 10 de março de 2016.