

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CAMPUS ARARAQUARA**

**SISTEMA DE ESTABILIZAÇÃO DE VÔO DE UMA
AERONAVE NÃO TRIPULADA**

Adriano A. Camargo
Ana Carla Barbosa
Eduardo Eugenio
Paulo Sérgio F. Dorm
Valdecir Valerian Junior

Araraquara, SP

2014

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DE SÃO PAULO
CAMPUS ARARAQUARA**

**SISTEMA DE ESTABILIZAÇÃO DE VÔODE UMA
AERONAVE NÃO TRIPULADA**

Adriano A. Camargo
Ana Carla Barbosa
Eduardo Eugenio
Paulo Sérgio F. Dorm
Valdecir Valerian Junior

Relatório final apresentado ao Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Araraquara, submetido à avaliação, como requisito para conclusão do curso Técnico em Mecatrônica.

Araraquara, SP

2014

Ficha Catalográfica elaborada pela Biblioteca do IFSP – Campus Araraquara

S636 Sistema de estabilização de vôo de uma aeronave não tripulada / Adriano Adalberto Camargo et al. - Araraquara: [s.n.], 2014. – 65 f.

Orientador: Prof. Me. Fernando de Haro Moraes

Relatório Técnico (Trabalho de Conclusão de Curso) - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Araraquara.

1. Veículo aéreo não-tripulado. 2. Controle de vôo. 3. Arduíno. 4. Sistema de Estabilização. I. Camargo, Adriano Adalberto. II. Eugenio, Eduardo. III. Dorm, Paulo Sérgio Frederico. IV. Valerian Junior, Valdecir. V. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Araraquara. VI. Título.

CDD 629.132

SISTEMA DE ESTABILIZAÇÃO DE VÔO DE UMA AERONAVE NÃO TRIPULADA

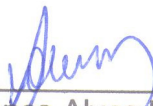
Relatório Final de Curso, apresentado ao Instituto Federal de São Paulo (IFSP), Campus de Araraquara, como parte das exigências para a conclusão do curso técnico em Mecatrônica.

Araraquara, 03 de junho de 2014.

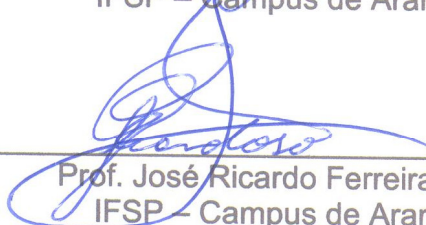
BANCA EXAMINADORA



Prof. Me. Fernando de Haro Moraes
IFSP – Campus de Araraquara



Prof. Me. Lourenço Alves Pereira Júnior
IFSP – Campus de Araraquara



Prof. José Ricardo Ferreira Cardoso
IFSP – Campus de Araraquara



Alexandre Machado Ferraz
Coordenador de Área - Industrial
Instituto Federal de São Paulo
Campus Araraquara

*"As dificuldades são como as montanhas.
Elas só se aplainam quando avançamos sobre elas. "*
Provérbio japonês

*Dedicamos este trabalho a todos que
contribuíram direta ou indiretamente em
nossa formação acadêmica.*

AGRADECIMENTOS

Agradecemos cada um de nós, aos outros membros do grupo por terem realizado o trabalho com afinco e produzido resultados de qualidade.

A Deus, a quem devemos nossa vida.

A nossa família que sempre nos apoiaram nos estudos e nas escolhas tomadas.

Ao orientador Prof. Me. Fernando de Haro Moraes que teve papel fundamental na elaboração deste trabalho.

Aos professores do campus que nos auxiliaram em momentos de dúvidas.

RESUMO

CAMARGO, Adriano A.; BARBOSA, A.C., EUGÊNIO, E. ; DORM, P.S.F. e VALERIAN JR, V. (2014). Sistema de Estabilização de um Aeromodelo, Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus de Araraquara, 63 pág.

Atualmente serviços de vigilância militar, análise de topografia de terrenos, fotografias aéreas, controle de desmatamento são realizados por Veículos Aéreos Não Tripulados (VANT), o qual pode ser qualquer tipo de aeronave que não necessita de pilotos embarcados para ser guiada. Esses aviões são controlados à distância, por meios eletrônicos e computacionais, sob a supervisão e governo humanos, ou sem a sua intervenção, por meio de Controladores Lógicos Programáveis (CLP). Um dos grandes desafios desses protótipos é manter se em altitude constante em vôo nivelado em relação aos três eixos de rotações, guinada, rolagem e arfagem, devido tanto as intempéries climáticas como correntes ar, ventos, tendências de mudança de rota relativas ao projeto da aeronave. Foi utilizada uma plataforma experimental de um veículo aéreo não tripulado, construída com a configuração de asa alta, fuselagem de isopor de alta densidade (P3), trem de pouso do tipo quilha. O aeromodelo possui motor elétrico com configuração trator. O conjunto, micro controlador ArduinoMega 2560 (hardware), algoritmo(software) e os acelerômetros de três eixos (sensores GY-80) calculam o comportamento da aeronave nos três eixos de movimentação e detectam as mudanças de posição do modelo em relação ao plano horizontal. Caso o comportamento da aeronave se altere, sinais de controle serão enviados as superfícies de comando, ailerons, profundor e leme de maneira a compensar as mudanças. E realizando esse processo constantemente, estabiliza a aeronave, mantendo-a em um estado de equilíbrio contínuo, o que resulta num vôo suavemente nivelado. O sistema, quando ativado, é possível retomar o controle do vôo a qualquer momento, bastando movimentar as hastes do rádio transmissor e soltá-la até a posição neutra do controle, assim, a aeronave retorna imediatamente ao modo de vôo nivelado. O vôo nivelado pode ser ativado ou desativado com um simples toque no botão do radio transmissor.

Palavras-chave: Veículo aéreo não-tripulado; Controle de vôo; Arduino.

ABSTRACT

Currently military surveillance, analysis of the topography of land, aerial photographs, deforestation control services are performed by Unmanned Aerial Vehicles (UAV), which can be any type of aircraft that does not require drivers to be guided shipped. These planes are controlled remotely, by electronic and computer facilities, under the supervision and human government, or without his intervention, through Programmable Logic Controllers (PLC). A major challenge of these prototypes is to keep in constant altitude in level flight for the three axes of rotation, yaw, roll and pitch, due both to climatic conditions such as air currents, winds, changing trends regarding the route of the aircraft design. An experimental platform for an unmanned aerial vehicle built with the high wing configuration, the fuselage of high density polystyrene (P3), landing gear keel type was used. The model airplane electric motor with own tractor configuration. The set, microcontroller Arduino Mega 2560 (hardware) algorithm (software) and three-axis accelerometers (sensors GY-80) calculate the behavior of the aircraft in the three axes of movement and detect changes in position of the model relative to the plane horizontal. If the behavior of the aircraft change, control signals are sent to control surfaces, ailerons, elevator and rudder so as to compensate for changes. And performing this process consistently, stabilizes the aircraft, maintaining it in a state of continuous equilibrium, resulting in level flight smoothly. The system, when activated, you can regain control of the flight at any time by simply moving the stalks radio transmitter and release it to the neutral position of the control, so the aircraft immediately returns to level flight mode. The level flight can be turned on or off with a push button on the radio transmitter.

Keywords: unmanned aerial vehicle; Flight control; Arduino.

LISTA DE FIGURAS

Pág.

Figura.1 – Esquema simplificado de um túnel de circulação fechada.....	20
Figura.2 – Espectro aerodinâmico de um perfil alar.....	21
Figura.3 – Deslocamento de um disco de cartão no ar.....	21
Figura.4 – Movimento do ar sobre o disco.....	22
Figura.5 – Espectro aerodinâmico do disco.....	22
Figura.6 – Distribuição e resultante das pressões sobre o disco.....	23
Figura.7 – Valor relativo da resistência para diversos sólidos do mesmo diâmetro.....	24
Figura.8 – Planos sustentadores.....	25
Figura.9 – Movimento do avião.....	26
Figura.10 – Ângulo de perda.....	27
Figura.11 – Ângulos do perfil.....	27
Figura.12 – Cento de pressão de um perfil biconvexo simétrico.....	28
Figura.13 – Cento de pressão de um perfil biconvexo assimétrico.....	28
Figura.14 – Cento de pressão de um perfil plano convexo.....	29
Figura.15 – Cento de pressão de um perfil côncavo convexo.....	29
Figura.16 – Cento de pressão de um perfil de dupla curvatura.....	29
Figura.17 – Eixos de rotação.....	30
Figura.18 – Diedro das asas.....	32
Figura.19 – Eixo de rolamento.....	33
Figura.20 – A hélice.....	36
Figura.21 – Tipos de hélices.....	37
Figura.22 – Sentidos de rotação do hélice.....	37
Figura.23 – Afinação e centragem do modelo.....	38
Figura.24 – Lançamento do aeromodelo.....	39
Figura.25 – Configuração inicial da aeronave pré fabricada.....	46
Figura.26 – Aeronave modificada com o sistema de estabilização de vôo.....	46
Figura.27 – Micro Servo.....	48

Figura.28 –Placa de sensores Gy-80.....	49
Figura.29 –Micro controlador ArduinoATMega 2560.....	50
Figura.30 –Bateria do Aeromodelo.....	50
Figura.31 – Motor do Aeromodelo.....	51
Figura.32 –Receptor do aeromodelo.....	51
Figura.33 –MULTIWII 2.2.....	52
Figura.34 –WinGUI.....	52
Figura.35 – Aeromodelo Danificado.....	53
Figura.36 – Perfil da asa do modelo atual.....	53
Figura.37– Fuselagem do modelo atual.....	54
Figura.38–Aeromodelo montado com os equipamentos.....	54
Figura.39 –Teste do aileron.....	55
Figura. 40 –Teste do profundor.....	55
Figura. 41 –Teste do leme.....	56

LISTA DE SIGLAS E ABREVIATURAS

IFSP	Instituto Federal de Educação, Ciência e Tecnologia do Estado de São Paulo
TCC	Trabalho de Conclusão de Curso
C.P.	Centro de pressão
C.G.	Centro de Gravidade
C.A.L.	Centro de área lateral
RC	Rádio controlado
LIPO	Polímero de Lítio
LiFePo	Polímero de Lítio – Ferro
VCC	Vôo circular controlado
A.M	Amplitude Modulada
F.M	Frequência Modulada
PCM	Pulso Codificado Modulado
GHz	Gigahertz
COBRA	Confederação Brasileira de Aerodelismo
ABA	Associação Brasileira de Aerodelismo
VANT	Veículo Aéreo Não Tripulado
VARP	Veículo Aéreo Remotamente Pilotado
CLP	Controladores Lógicos Programáveis
BIJ	Jornal Investigativo de Bureau
CIA	Agência Central de Inteligência
ONU	Organização das Nações Unidas
INCT	Instituto Nacional de Ciência e Tecnologia
SEC	Sistemas Embarcados Críticos
ICMC	Instituto de Ciências Matemáticas e Computação

USP	Universidade de São Paulo
IBAMA	Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis
APPs	Áreas de Preservação Permanente
E/S	Entrada e Saída
USB	Porta Universal
CPU	Unidade Central de Processamento
AVR	Micro Controlador fabricado pela empresa Atmel
MHz	Megahertz
RS,TTL, FTDI, FT,MLP	Uma família de circuitos digitais
ABNT	Associação Brasileira de Normas Técnicas
NBR	Normas Brasileiras
DC	Corrente Direta
AC	Corrente Alternada
CC	Corrente Contínua
Gy	Tipo de placa de sensor
RPM	Rotações Por Minuto
V	Volt
UARTs	Portas seriais de hardware
PWM	Modulador de largura de pulso
mA	MiliAmpère

LISTA DE SÍMBOLOS

A	Área, m ²
\uparrow -, \uparrow +	Girofrequência para o próton e para o elétron, respectivamente
C/C++.	Linguagem de Programação
N	Newton
R	Reação do ar
R_x	Resistência ao Avanço
R_z	Sustentação

SUMÁRIO

	<u>Pág.</u>
1. INTRODUÇÃO.....	17
1.1 Objetivos	18
1.2 Justificativa.....	18
2. AEROMODELISMO	18
2.1 Categorias de Aeromodelos.....	19
2.2 Regulamentação	19
2.3 Túneis aerodinâmicos.....	20
2.4 Resistencia do ar.....	21
2.5 Sustentação.....	25
2.6 Fenômeno de Perda	26
2.7 Forma dos Perfis.....	27
2.8 Estabilidade	30
2.9 Estabilidade Longitudinal.....	31
2.10 Estabilidade Lateral	31
2.11 Estabilidade de Rota.....	32
2.12 Centro de Area Lateral	33
2.13 Eixo de Rolamento	33
3. TIPOS DE MOTORES.....	34
3.1 Motores a explosão.....	34
3.2 Motores elétricos	35
3.3 A Hélice.....	36
4. AFINAÇÃO E CENTRAGEM	38
4.1 Afinação	38
4.2 Centragem	39
5. VEÍCULO AÉREO NÃO TRIPULADO	40
6. ARDUINO.....	41
6.1 Hardware	42
6.2 Software	43
6.3 Aplicações	43
6.4 Acessórios	44
7. PARÂMETROS DO PROJETO	45
8. MATERIAIS E MÉTODOS.....	46

8.1 Isopor.....	47
8.2 Depron.....	47
8.3 Cola Quente.....	48
8.4 Micro Servos.....	48
8.5 Placa de sensores GY-80.....	49
8.6 Micro Controlador Arduino.....	49
8.7 Bateria.....	50
8.8 Motor.....	51
8.9 Receptor.....	51
8.10 Programa utilizado na configuração.....	52
9. MODIFICAÇÕES FINAL DO PROJETO.....	53
10. RESULTADOS E DISCUSSÕES.....	54
11. CONSIDERAÇÕES FINAIS.....	56
12. REFERÊNCIAS BIBLIOGRÁFICAS.....	58
13. ANEXO I.....	60

1. INTRODUÇÃO

Dotados de grande capacidade de geração de imagens e informações de inteligência, esses aviões, controlados por um sistema autônomo de navegação ou por controle remoto, fornecem aos policiais da fronteira imagens em tempo real, potencializando o combate ao crime organizado e ao tráfico de drogas. O VANT mostra que a tecnologia embarcada em pequenos e médios aviões pode ser usada para fins de defesa e segurança, utilizando as imagens para agilizar nas abordagens. Os VANTs também surgem como grandes aliados da defesa do meio ambiente. É um apoio de peso ao trabalho das patrulhas terrestres que percorrem regiões imensas à procura de agressões ao meio ambiente. Estima-se que o país tenha hoje entre 65 a 70 milhões de hectares de áreas agrícolas e para fiscalizar essa área, o IBAMA conta com um contingente de apenas 1.319 fiscais espalhados por todo o território nacional. Verifica-se também um alto índice de impunidade, uma vez que flagrar a agressão ao meio ambiente é uma tarefa difícil e exige muitas vezes que os fiscais arrisquem suas vidas no exercício de sua função. A alta qualidade das imagens captadas permite aos aviões não-tripulados detectar com precisão e eficiência áreas desmatadas, diminuição de áreas de preservação permanentes, atividades irregulares de mineração em leitos de rios, além de identificar crimes como, queimadas, pesca ilegal e corte de vegetação nativa. Se adaptado com sensores termais e multi espectrais, os VANTs podem revelar dados impossíveis de serem vistos a olho nu. Apenas para exemplificar, o uso de tecnologia infravermelha pode revelar, entre outros, a utilização de substâncias químicas empregadas em lavouras, a contaminação de cursos d'água por agentes químicos, o estresse hídrico de plantações e doenças em áreas agrícolas; é a tecnologia usada a favor da proteção do meio ambiente e da agricultura.

Foi utilizado um protótipo de um aeromodelo para a implantação de um sistema de estabilização de vôo autônomo. Assim como micro servos, um motor brushless muito usado na área de aeromodelismo, uma placa de arduino mega utilizado para controlar o sistema do protótipo e uma placa de sensores GY-80 com três eixos dotado de giroscópios, barômetro, magnetômetro, acelerômetro e termômetro. O aeromodelo foi feito com o material depron, muito usado em aeromodelos por ser leve e barato.

1.1 Objetivos

O objetivo dessa pesquisa é implementar um sistema de controle de estabilização de vôo (hardware, software e sensores) em um veículo aéreo não tripulado. A aeronave deve manter - se em altitude constante em vôo nivelado em relação aos três eixos de rotações, guinada, rolagem e arfagem, devido.

1.2 Justificativa

Atualmente serviços de vigilância militar, análise de topografia de terrenos, fotografias aéreas, controle de desmatamento são realizados por Veículos Aéreos Não Tripulados (VANT), o qual pode ser qualquer tipo de aeronave que não necessita de pilotos embarcados para ser guiada. Esses aviões são controlados à distância, por meios eletrônicos e computacionais, sob a supervisão e governo humanos, ou sem a sua intervenção, por meio de Controladores Lógicos Programáveis (CLP). Um dos grandes desafios desses protótipos é manter se em altitude constante em vôo nivelado em relação aos três eixos de rotações, guinada, rolagem e arfagem, devido tanto às intempéries climáticas como correntes ar, ventos, tendências de mudança de rota relativas ao projeto da aeronave. O sistema de controle deve manter um estado de equilíbrio contínuo, o que resultará num vôo suavemente nivelado.

No projeto atual a aeronave é controlada manualmente através de uma rádio controle. Ao automatizar o sistema nivelado de vôo, será possível traçar rotas de vôo autônomas no sistema de GPS.

2. AEROMODELISMO

É a ciência que estuda o aeromodelo [1] , ou seja, aparelho voador de dimensões reduzidas e não susceptíveis de transportar um ser humano.

O Aeromodelismo desenvolve a habilidade manual e o interesse para os assuntos voltados a aviação, através da aplicação pratica de idéias e teorias baseadas na técnica aeronáutica mundial. É ainda excelente atividade educativa ao exigir perseverança, precisão, método, raciocínio e muita disciplina.

2.1 Categorias de Aerodelismo

- **VCC ou U/Control** - vôo circular controlado, no qual o aerodelo fica ligado ao aerodelista por meio de cabos, que podem variar de 15 a 18 metros de comprimento.
- **Radio controlado**- o aerodelo é controlado por meio de um transmissor de radiofrequências, das quais podem ser FM, AM, PCM para sistemas mais antigos, porém ainda muito utilizados e para os sistemas mais modernos são utilizados sistema 2.4GHz que são mais seguros não correndo risco de interferência.
- **Vôo livre** - o aerodelo, depois de lançado, não sofre mais nenhuma interferência por parte do aerodelista. Pode ser aerodelo com motor, com elástico ou sem propulsão própria.

2.2 Regulamentação

A Confederação Brasileira de Aerodelismo (COBRA) tem sua sede em São Paulo e foi fundada em 1959, reúne os clubes e associações dedicadas à prática da atividade no território brasileiro. Sua antiga denominação é Associação Brasileira de Aerodelismo (ABA) [2] .

A COBRA é reconhecida pelo Departamento de Aviação Civil do Ministério da Defesa e pelo Conselho Nacional de Desportos do Ministério do Esporte, como órgão máximo de direção do aerodelismo no Brasil. Entre as suas atribuições, incluem-se difundir, coordenar, organizar e dirigir o aerodelismo no Brasil, representando-o internacionalmente. A Confederação é ainda responsável por organizar periodicamente os Campeonatos Brasileiros de Aerodelismo.

A COBRA é integrada pelas federações, associações e clubes filiados que, através de seus presidentes e vice-presidentes, formam a sua Assembléia Geral.

2.3 Túneis Aerodinâmicos

Todos os objetos, que se deslocam inteiramente na atmosfera, estão sujeitos, a sofrer atrito e reação do ar, a algumas forças que determinam o seu comportamento [3] . Em grande parte, as experiências relativas a esse comportamento são efetuadas em túneis aerodinâmicos, considerados autênticos laboratórios do ar, onde se reproduzem as condições de vôo e se estuda a ação do ar sobre os perfis das asas, hélices, aeromodelos e até sobre aviões reais.

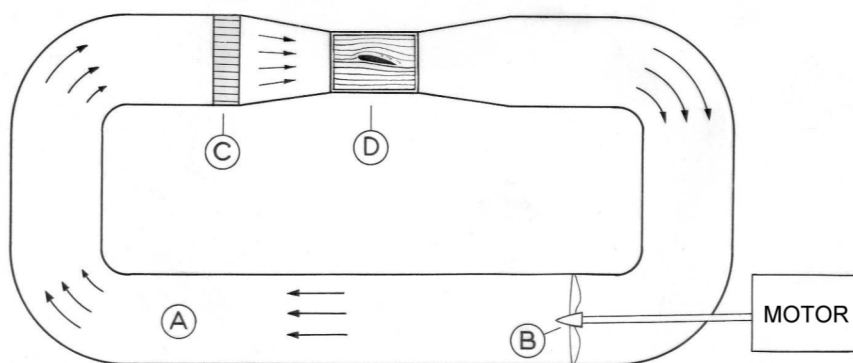


Figura 1 – Esquema simplificado de um túnel de circulação fechada.

Fonte: RODRIGUES, 2004

Os túneis aerodinâmicos (Figura 1) se constituem por uma condução do ar (A), no interior da qual está instalado um compressor (B), ligados por um motor, que produz uma determinada velocidade do ar. O filtro celular (C), que tem a missão de fornecer à câmara de provas (D), que se encontra o corpo para o estudo, uma corrente de ar, livre de turbilhões, ou seja, um jato certo, constituído por camadas de trajetórias retilíneas e paralelas, a que se denominam linhas de corrente ou veia fluída. As linhas de corrente são as representações gráfica do caminho das moléculas do ar, em movimento.

À imagem mostra as linhas de corrente em torno de um perfil de asa é conhecida como espectro aerodinâmico.

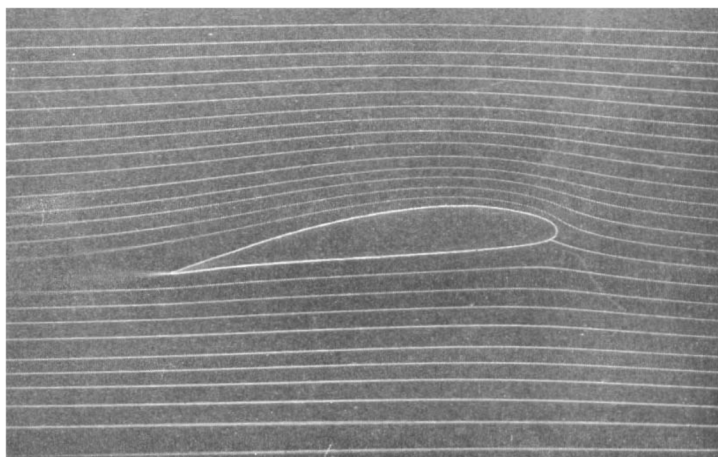


Figura 2 – Espectro aerodinâmico de um perfil alar (foto Lippisch).

Fonte: RODRIGUES, 2004

2.4 Resistência do Ar

O ar é um gás e como todos os gases, compõe-se de um agregado de pequenas partículas, a que se chamam moléculas [3] .

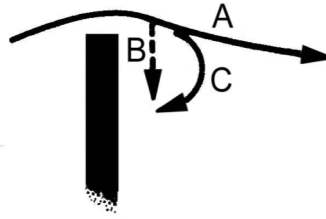
Quando se desloca no ar qualquer objeto, as moléculas do ar têm de desviar afastando-se umas das outras para dar passagem ao corpo, visto que não podem passar através dele. Assim ao deslocar no espaço um disco de cartão como mostra a (figura 3), nota-se imediatamente que, para o fazer avançar, terá de empregar uma determinada força. A pressão que notamos no disco não é mais do que a força que foi necessário gastar para desviar o ar à passagem do corpo, separando as suas moléculas. O disco, no seu movimento separa, portanto as partículas do ar, obrigando-as a contornar o obstáculo até aos bordos e a juntarem-se de novo na face posterior.



Figura 3 deslocamento de um disco de cartão no ar.

Fonte: RODRIGUES, 2004

Assim, geram-se, atrás do disco, movimentos desordenados de ar que se chamam turbilhões, a fim de efetuar a necessária mistura do ar, que corre à volta do disco, com o que se encontra na face posterior (Figura 4).



A – Movimento do ar

B – Movimento tendente a encher o vazio

C – Movimento resultante

Figura. 4 – Movimento do ar sobre o disco

Fonte: RODRIGUES, 2004

Considerando o disco na câmara de provas de um túnel, sujeito a uma corrente de ar, observa-se o seu espectro aerodinâmico (figura 5).

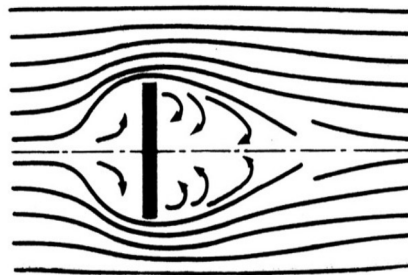


Figura 5 – Espectro aerodinâmico do disco

Fonte: RODRIGUES, 2004

As partículas de ar que, tendem a percorrer trajetórias retilíneas e paralelas, ao encontrarem o obstáculo são obrigadas a contorná-lo até aos bordos e a acelerar na face posterior em movimentos turbulentos, criando reações no disco.

À frente, forma-se uma pressão, porque, para obrigar o ar a desviar-se do seu caminho normal, é preciso que o disco exerça sobre ele certa força, que se traduz num aumento de pressão no ar. Entretanto, na face posterior, o ar tem tendência a seguir em frente, pelo que é necessário forçá-lo a contornar o disco. Isto é, torna-se também necessário exercer sobre o ar uma força que, neste caso é fácil de compreender, determina uma diminuição de pressão ou, simplesmente, uma pressão insuficiente.

Sendo assim, o ar exerce sobre o disco, à frente, forças de pressão e, atrás, forças de depressão, cuja resultante está aplicada no centro do disco e dirigida no sentido da corrente de ar. Esta força que o ar exerce sobre o disco é igual à força que o disco exerce sobre o ar, embora de sentidos opostos. À primeira – à reação do ar sobre o disco – chama-se resistência do ar. A segunda – a ação do disco sobre a veia fluida – representa a força que é necessária aplicar no disco para fazê-lo deslocar no ar

Na (Figura 6), entende-se por distribuição de pressões a representação gráfica das pressões negativas (forças de depressão); pressões positivas (forças de pressão), que atuam no corpo. A disposição das forças faz-se equitativamente em volta do centro C , e compreende-se por que, dado a vista simétrica da passagem do ar em torno do disco.

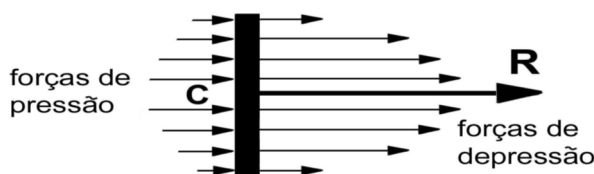


Figura 6 – Distribuição e resultante das pressões sobre o disco

Fonte: RODRIGUES, 2004

Portanto, a resultante R está aplicada no ponto C , com a mesma direção do deslocamento e de sentido inverso. A forma da imagem aerodinâmica de um disco é semelhante, quer seja de grandes dimensões, ou de proporções reduzidas. Assim, a distribuição de pressões será equivalente para ambos.

Sabe-se que a força R é igual à diferença de pressões vezes a área do disco, a resistência do ar será, por isso, proporcional a essa área.

Assim a resistência do ar para um corpo qualquer, varia com a área frontal que é a superfície da maior seção perpendicular à direção das linhas de corrente desse corpo.

Entretanto, para a mesma área frontal – partindo ainda do princípio de que a resistência é a força igual ao produto da diferença de pressões pela área frontal – quanto menor for essa diferença, menor será também a resistência.

Portanto, do mesmo modo que o automóvel vira mais facilmente ao descrever uma curva de grande raio, também o ar ficará sujeito a uma pressão menor quanto mais suave for a curvatura das linhas de corrente.

Segundo RODRIGUES, 2004, as formas que determinarem a menor curvatura da veia fluida serão as mais aerodinâmicas, porque, para a mesma área frontal e para a mesma velocidade, estabelecem a menor diferença de pressões. Sabendo-se que as forças

de inércia variam com o quadrado da velocidade, sendo que a resistência do ar é proporcional ao quadrado da velocidade.

Resumindo, a resistência do ar varia:

- 1) Com a área frontal;
- 2) Com a distribuição de pressões que, por sua vez, varia com a forma e com a velocidade.

Baseando se na forma dos corpos, na (Figura 7) se encontra o valor relativo da resistência para diversos sólidos do mesmo diâmetro, com as respectivas formas aerodinâmicas.

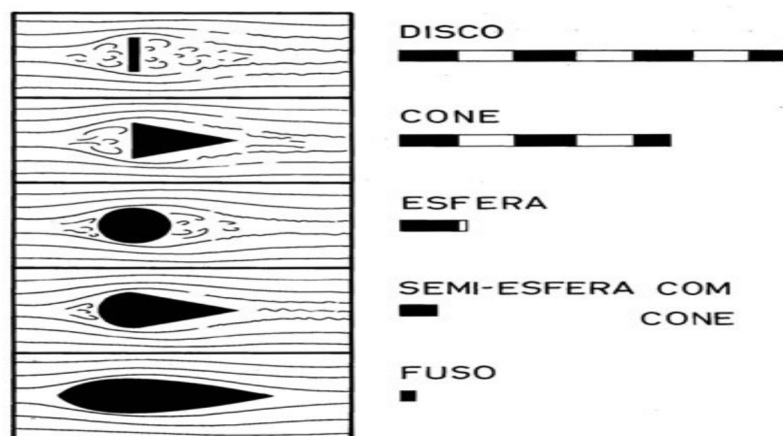


Figura 7 – Valor relativo da resistência para diversos sólidos do mesmo diâmetro

Fonte: RODRIGUES, 2004

À medida que os corpos vão sendo alongado, o comportamento das partículas de ar se torna menos turbulento, criando trajetórias quase regulares da veia fluida no caso da forma fuselada. Aliás, se observar a imagem aerodinâmica do disco, verifica-se que a zona turbulenta tem sensivelmente a forma de um corpo fuselado. Na verdade, esta forma, muito característica nos aviões, por ser a que cria menor resistência ao avanço, foi obtida pelo preenchimento da zona de turbulência criada em redor do disco. Ela permite que as partículas de ar se desviem à frente, sem virarem violentamente, ao aproximarem-se do corpo, e o contornem de uma forma suave, unindo-se de novo sem movimentos desordenados. A forma fuselada é, pois, a mais lógica a dar aos corpos que se desloquem inteiramente na atmosfera. Adaptado de (RODRIGUES)

2.5 Sustentação

É a asa que assegura a sustentação. É exatamente a ação do ar sobre a asa em movimento que criam determinadas forças, que chegam para suportar o peso do avião [3].

Na maior parte dos aeromodelos, a sustentação é assegurada por mais um elemento: o estabilizador ou empenagem horizontal. Assim, a asa e o estabilizador tomam a ação específica de planos sustentadores. Pode, dizer-se que a seção de uma placa curva representa a linha média do perfil semelhante à asa.

A asa comporta-se de forma similar à lâmina curva; logo a distribuição de pressões é também semelhante e a força \mathbf{R} (reação do ar) estará, situada junto do bordo anterior do perfil e aplicada no centro de pressão. No perfil semelhante à asa este centro encontra-se, mais ou menos, a 1/3 do bordo da frente, para ângulos de ataque reduzidos.

Ao decompor a força \mathbf{R} obtem-se um vetor \mathbf{R}_x , paralelo à direção da corrente de ar, e outro perpendicular, representado por \mathbf{R}_z .

A componente \mathbf{R}_z é a sustentação, que se opõe ao peso do aeromodelo, e a componente \mathbf{R}_x a resistência ao avanço, que se torna necessária vencer para assegurar o movimento.

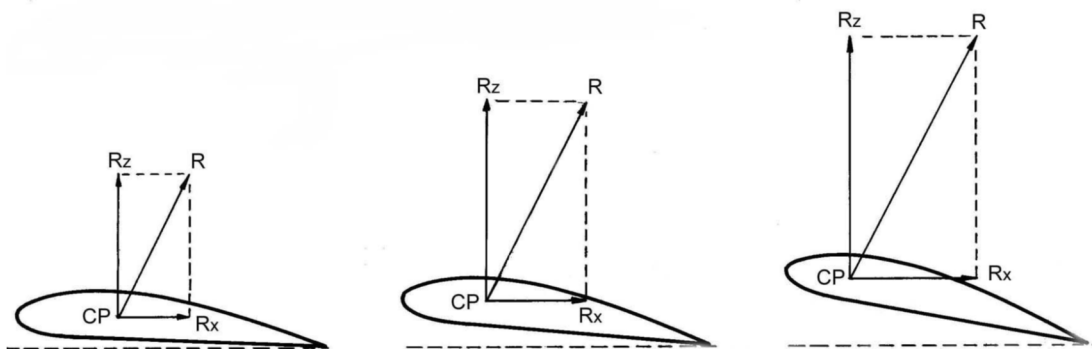


Figura 8 Planos sustentadores

Fonte: RODRIGUES, 2004

Ao fazer variar o ângulo de ataque do perfil, todos os restantes elementos se modificam com a alteração do ataque obtemos uma forma aerodinâmica diferente e, conseqüentemente, uma variação na distribuição de pressões e respectiva resultante.

Assim, todos os outros elementos estão dependentes da força \mathbf{R} , as suas grandezas e posições são naturalmente alteradas. Pelo exame das figuras verifica-se que, em virtude do aumento do ângulo de ataque, a reação aerodinâmica \mathbf{R} inclina-se para

trás e atinge um valor mais elevado. A grandeza das forças de sustentação e resistência aumenta também e o centro de pressão desloca-se para frente.

A sustentação é sempre a componente perpendicular à direção do deslocamento e a resistência ao avanço a componente que tem o mesmo sentido da corrente de ar relativa que é determinada pela direção do ar, em relação ao avião. Se o avião voa horizontalmente, a corrente de ar relativa é horizontal. Se o avião “pica” ou “cabra”, a direção do vento relativo é oblíqua. Assim, a direção da corrente de ar relativa confundir-se-á com a direção da deslocação e o sentido será exatamente oposto ao do movimento do avião (Figura 9).



Figura 9 Movimento do avião

Fonte: RODRIGUES, 2004

2.6 Fenômenos da Perda

Sobre a sustentação, sabe-se que esta força, a de maior utilidade ao vôo, cresce muito rapidamente com o aumento do ângulo de ataque e, nos aeromodelos, atinge o máximo a ângulos de cerca de 8 a 10 graus, conforme os perfis. A partir daí, diminui rapidamente [3].

Se dermos ao perfil uma inclinação exagerada, verifica-se que as linhas de corrente, cujo escoamento se fazia, até aí, regularmente, se desprendem no extradorso do perfil, criando uma zona de violentos turbilhões. É a própria inércia das partículas do ar que provoca esse descolamento, ou insuficiente aderência da veia fluida.

Nestas circunstâncias, a sustentação decresce rapidamente, enquanto a resistência aumenta de uma forma considerável. Àquele ângulo de ataque crítico, que determina a destruição da saída laminar e dá lugar à formação de turbilhões, chama-se ângulo de perda.

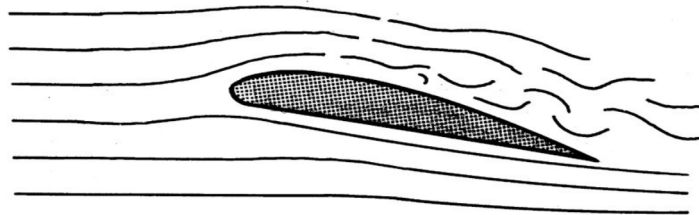


Figura 10 Ângulo de perda

Fonte: RODRIGUES, 2004

A diminuição da velocidade referente, ocasionada por esse aumento anormal da resistência, cria, pois, em determinada altura, o fenômeno da perda, em que os efeitos aerodinâmicos são reduzidos a tal ponto que o valor da sustentação não será suficiente para equilibrar o peso do aparelho, e este cai. A essa velocidade limite chama-se velocidade de perda. O ângulo de perda varia com a forma do perfil da asa e, no mesmo perfil, com a extensão da asa. Quanto maior for o alongamento, menor o ângulo de perda.

2.7 Formas dos Perfis

Os perfis classificam-se em biconvexos (simétricos e assimétricos), plano-convexos, côncavo-convexos e de dupla curvatura [3].

PERFIS BICONVEXOS SIMÉTRICOS – São os perfis cujo dorso e ventre são convexos e iguais. Estes perfis oferecem uma resistência mínima e não criam sustentação à 0° . Têm ainda a particularidade de, abaixo dos 0° , criarem sustentação negativa, de valor idêntico – contudo de sentido contrário – à sustentação criada no ângulo positivo correspondente. Este perfil emprega-se em modelos de acrobacia que, em virtude das figuras que têm de efetuar, necessitam de comportar-se de igual modo em posições invertidas e direitas.

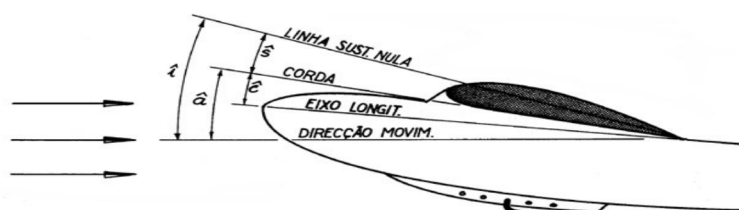


Figura 11 Ângulos do perfil
 \hat{i} – ângulo de incidência
 \hat{a} – ângulo de ataque
 s^\wedge – ângulo de sustentação nula
 c^\wedge – ângulo de calado

Fonte: RODRIGUES, 2004

A posição do centro de pressão destes perfis é muito pouco variável e por isso se denominam estáveis. O centro de pressão encontra-se a cerca de 25 % da corda, isto é, a 1/4 da profundidade (Figura 12).

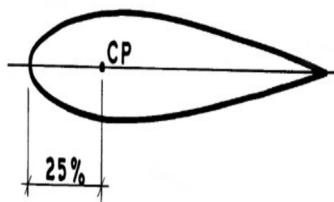


Figura 12 Centro de pressão de um perfil biconvexo simétrico

Fonte: RODRIGUES, 2004

PERIFIS BICONVEXOS ASSIMÉTRICOS – Nestes perfis, tanto o dorso como o ventre são convexos, apresentando, todavia, o intradorso uma curvatura menos pronunciada. As incidências reduzidas oferecem boa sustentação e mínima resistência, razão por que se empregam, com bons resultados, em modelos de velocidade pura. O seu centro de pressões encontra-se sensivelmente a 28 % da corda (Figura 13).

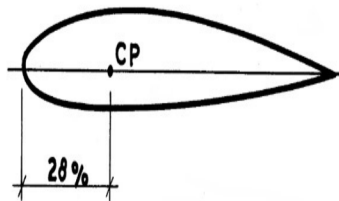


Figura 13 Centro de pressão de um perfil biconvexo assimétrico

Fonte: RODRIGUES, 2004

PERFIS PLANO-CONVEXOS – Perfis em que o intradorso é plano e o extradorso convexo.

São muito sustentadores e usam-se em modelos de treino, de rádio-controle e empenagens horizontais sustentadoras. O seu centro de pressões encontra-se, aproximadamente, a 30 % da corda (Figura 14).

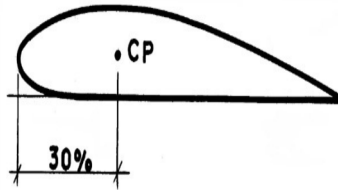


Figura 14 Centro de pressão de um perfil plano convexo

Fonte: RODRIGUES, 2004

PERFIS CÔNCAVO-CONVEXOS – Perfis cujo ventre e dorso são de forma côncava e convexa, respectivamente. Os perfis côncavo-convexos criam maior sustentação, mas oferece, em contrapartida, mais resistência ao avanço do que quaisquer outros.

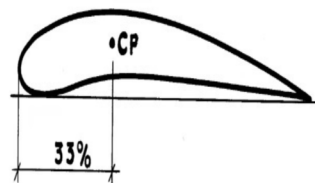
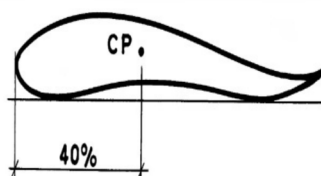


Figura 15 Centro de pressão de um perfil côncavo convexo

Fonte: RODRIGUES, 2004

PERFIS DE DUPLA CURVATURA – Perfis que, tanto no dorso como no ventre, têm formas côncava e convexa. Estes perfis são empregados exclusivamente em asas voadoras (modelos destituídos de empenagem horizontal) pela particularidade, já apontada, onde seu centro de pressão se desloca ao contrário dos restantes perfis alares: recuando com o aumento do ângulo de ataque, o *C.P.* restabelece o equilíbrio sem o auxílio do plano de cauda (Figura 16).

Por este motivo, também se designam por perfis auto-estáveis.



2.8 Estabilidade

Estabilidade é a propriedade que um aeromodelo possui de poder retomar a sua linha de vôo normal logo que cesse a causa que provocou o desequilíbrio [3].

Um aeromodelo que, voando segundo uma trajetória regular, baixa inesperadamente, por exemplo, uma das asas ou levanta o nariz, em virtude de qualquer interferência estranha, e logo em seguida volta à posição normal de vôo, significa que ele é estável. Se, ao contrário, o modelo, desviado da sua posição de equilíbrio, tende a desviar-se mais dessa posição, dizemos que é instável.

Um aeromodelo, em vôo, está sujeito a movimentos que se realizam em torno de três eixos de rotação principais, que são: eixo transversal, eixo longitudinal e eixo vertical.

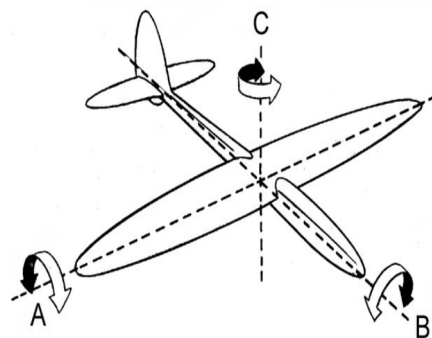


Figura 17 Eixos de rotação

A – eixo transversal

B – eixo longitudinal

C – eixo vertical

Fonte: RODRIGUES, 2004

Em torno do eixo transversal, realizam-se os movimentos de picar ou cabrar. Ao eixo longitudinal correspondem os movimentos de inclinação da asa. Em torno do eixo vertical, realizam-se os movimentos de deriva, à esquerda e à direita.

Um aeromodelo deve, para ser estável, corrigir em torno dos seus eixos as perdas de equilíbrio acidentais, de modo que a linha de vôo coincida aproximadamente

com o eixo longitudinal do aeromodelo. O aeromodelo terá de realizar em torno desses eixos a estabilidade correspondente, para voar em estado de equilíbrio.

2.9 Estabilidade Longitudinal

Ao eixo transversal corresponde a estabilidade longitudinal e os movimentos de referência são os de picar e cabrar [3].

Para assegurar a indispensável estabilidade longitudinal, foram os modelos dotados de planos de cauda que têm como função principal equilibrar o modelo com uma forma aerodinâmica, que gera um momento oposto ao da asa. A asa encontra-se em equilíbrio, pois a resultante é nula, já que ambas as forças, aplicadas no mesmo ponto e de sentidos opostos, têm a mesma grandeza. Para se obter o equilíbrio desejado, dotam-se os aviões de um plano horizontal de cauda, ou estabilizador, que, solidário à asa por meio da fuselagem, criará também, com as variações do ângulo de ataque, um binário que contraria o gerado na asa.

2.10 Estabilidade Lateral

Os movimentos que correspondem à estabilidade lateral são os de berço, ou de inclinação da asa, e realizam-se em torno do eixo longitudinal [3].

A estabilidade lateral assegura-se dotando os modelos de determinadas superfícies verticais (derivadas), normalmente colocadas acima do eixo longitudinal, e dispondo as asas em diedro, isto é, elevando as suas extremidades em relação à parte central.

As formas de diedro mais usadas em aeromodelos são as seguintes:

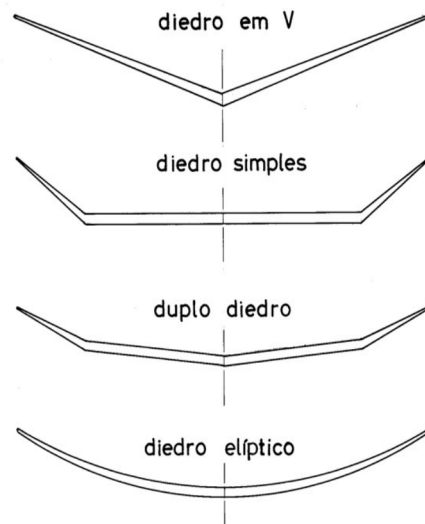


Figura 18 Diedro das asas
Fonte: RODRIGUES, 2004

O mais eficiente é, sem dúvida, o diedro elíptico, mas a grande dificuldade na sua construção leva os praticantes a usarem outras formas de diedro, também eficientes.

O duplo diedro tem uma ação estabilizadora muito enérgica e, por isso, é utilizado com frequência nos moto modelos de vôo livre, em virtude da instabilidade na saída destes modelos.

2.11 Estabilidade de Rota

A estabilidade de rota, que se realiza em torno do eixo vertical, opõe-se aos movimentos de deriva, à esquerda e à direita [3].

A estabilidade de rota é assegurada por uma racional distribuição da área lateral do modelo e, em muito menor escala, pela forma em flecha dada à asa.

No entanto, o que assegura essencialmente a estabilidade de rota é uma distribuição adequada da área lateral do modelo, da qual a maior parte deve encontrar-se atrás do eixo vertical.

Em resumo, para se obter uma boa estabilidade lateral, deve o centro de área lateral do modelo encontrar-se no eixo longitudinal, um pouco atrás do centro de gravidade, tanto mais recuado quanto maior se pretenda o momento estabilizador.

2.12 Centro de Área Lateral (C.A.L.)

O *C.A.L.* é o ponto onde se supõe aplicada a resultante das forças que atuam perpendicularmente ao plano vertical do modelo, considerado na direção do movimento [3].

Por outras palavras, é o ponto de aplicação da resultante de todas as forças aerodinâmicas, que atacam lateralmente o modelo.

2.13 Eixo de Rolamento

O eixo de rolamento se define como a reta que passa pelos centros das áreas laterais, anterior e posterior, tomadas com referência ao centro de gravidade [3] (Figura 19).

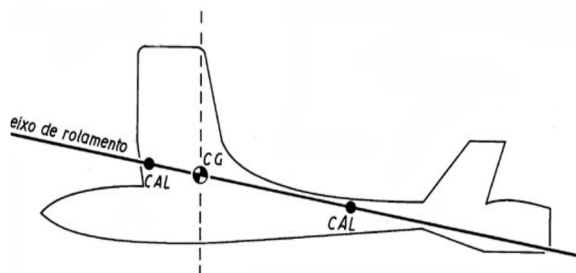


Figura 19 Eixo de rolamento

Fonte: RODRIGUES, 2004

Essa linha determina o eixo segundo o qual o modelo roda quando é investido por uma corrente de ar lateral. Se o eixo de rolamento fizer com a direção do movimento um ângulo positivo, o modelo tenderá a inclinar o nariz para cima, verificando-se o contrário, se o eixo tiver uma incidência negativa.

Daqui se conclui assumir este eixo grande importância no voo em espiral, principalmente nos modelos a motor (moto modelos de voo livre, «borrachas», telecomandados, etc.), nos quais é fundamental assegurar uma distribuição da área lateral que determine um eixo de rolamento de incidência positiva.

Isso permitirá uma mais segura estabilidade nas saídas em espiral, sendo mais um fator a contribuir para que o modelo, nas voltas, não incline o nariz para baixo e entre em parafuso.

3. TIPOS DE MOTORES

De acordo com [3] atualmente a categoria mais praticada de aeromodelismo é o radio controlado (RC), que se divide basicamente em três modalidades, que se diferem pelo seu tipo de motor:

3.1 Motores a Explosão (combustão interna)

Podem chegar a escalas maiores que 50% das dimensões reais de uma aeronave [3]. Seus motores apresentam alta potência e isso produz um vôo mais adequado e prazeroso. São de longe, os aeromodelos mais utilizados em todo o mundo, com milhões de unidades vendidas ao longo dos anos.

As principais vantagens deste tipo de aeromodelo são:

- Um envelope de vôo mais real, aumentando a possibilidade de treino dos pilotos
- O som dos motores e o fato de não ligarem sem que seja efetuado um procedimento de partida costuma evitar a grande maioria dos acidentes.
- A durabilidade, pois aeromodelos a combustão bem cuidados costumam durar por mais de 20 anos
- O baixo custo, pois aeromodelos a combustão (por serem extremamente populares) podem ser adquiridos usados a um baixo preço, e sua manutenção é quase ausente de custos
- Os reparos são simples de se executar, pois a madeira é facilmente colável.
- Devido a sua alta potência, os aeromodelos a combustão podem ser construídos com vários materiais diferentes, como fibra de vidro, madeira, metal, espuma PU, fibra de carbono e etc.
- Como normalmente são aviões maiores, fica mais fácil de enxergar, minimizando os acidentes e perdas de material.
- Normalmente os materiais e componentes envolvidos na construção e na linkagem destes aeromodelos, são de maior qualidade, evitando-se assim acidentes desnecessários e perigosos.

- As principais desvantagens são:
- Sujeira, o óleo deixa o aeromodelo bem sujo após um voo
- Ruído, o que limita seu uso às pistas de aeromodelismo devidamente homologadas

3.2 Motores Elétricos

De acordo com [3] utilizam alta tecnologia como, por exemplo, baterias de Polímero de Lítio (LiPo), motores "brushless" (sem escovas), e têm como um dos diferenciais a possibilidade de construir modelos com tamanho e peso reduzidos, como na classe micro, que engloba aeromodelos minúsculos, que chegam a pesar apenas 4 gramas e ter 15 centímetros de envergadura.

Apesar do destaque principalmente para modelos menores, a atual geração de motores brushless e baterias LiPo e LiFePo permitem a utilização de motorização elétrica em modelos nas mais diferentes escalas, chegando a mais de 10m de envergadura.

Além da possibilidade de modelos em escala micro, outras vantagens são:

- Baixo nível de ruído;
- Facilidade na montagem de modelos com extrema acuidade de escala visual, pois motores elétricos não precisam de aberturas para escapamento;
- Facilidade na montagem de aeromodelos multimotores (bimotores, trimotores, quadrimotores, etc.), devido ao menor peso dos motores, ausência de vibração e por ser manterem curvas de aceleração equilibradas entre os diversos motores sem necessidade de cuidados adicionais com regulagens;
- Envelope de voo mais abrangente, permitindo pousos lentos, vôos em locais fechados, ou parques;
- O baixo custo desta categoria, inferior ao custo e a manutenção de um aeromodelo similar a combustão. Mesmo inicialmente sendo recomendável adquirir algumas baterias adicionais e carregador, não existe necessidade de compra de combustível;
- Facilidade, baixo custo e pouco tempo de reparação quando ocorre dano devido a queda.
- As principais desvantagens são:

- O tempo reduzido de vôo, pois a bateria não voa mais do que poucos minutos;
- A baixa durabilidade do avião, pois normalmente os aeromodelos elétricos são feitos de isopor e este material se deforma com o tempo, além disso, é facilmente quebrável;
- Devido ao seu baixo ruído e a capacidade de serem acionados por faltas de sinal do transmissor, acidentes podem ocorrer.

3.3 A Hélice

A hélice é o elemento do aeromodelo que transforma o movimento de rotação do motor no de translação do aparelho. Por outras palavras, o hélice transforma a energia mecânica fornecida pelo motor de explosão ou pela meada de elástico na potência necessária à tração do modelo [3].

É ao conjunto hélice–motor que se chama grupo moto propulsor do aeromodelo ou avião.

A hélice é constituída por uma, duas ou mais lâminas, dispostas com certa inclinação, que tomam o nome de pás, e pelo cubo, que é a parte central, vizinha do eixo.

Consoante o número de pás, os hélices denominam-se de mono pás, bi pás, tri pás, etc.

As suas características aerodinâmicas são muito semelhantes às de uma asa e, assim, distinguem-se também, no hélice: o bordo de ataque, o bordo de fuga, os bordos marginais, os perfis, etc.

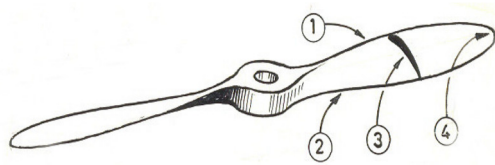


Figura 20 A hélice

1 – bordo de ataque;

2 – bordo de fuga;

3 – perfil;

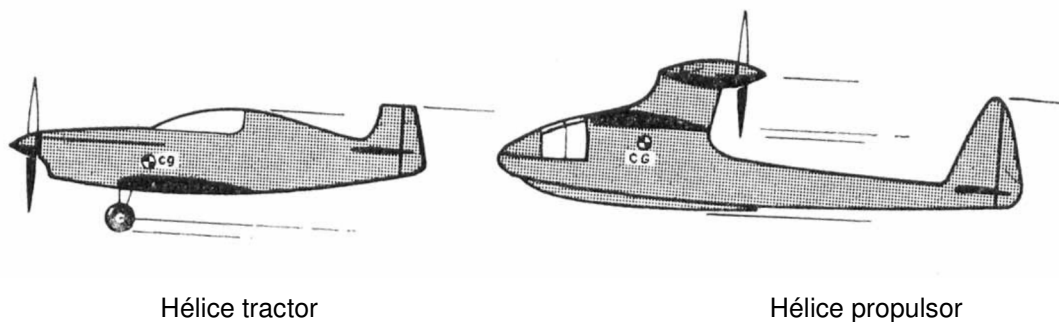
4 – bordo marginal.

Fonte: RODRIGUES, 2004

As hélices podem ser tratores ou propulsores e esquerdos ou direitos.

A hélice trator está colocada e exerce a sua ação, à frente do centro de gravidade e a hélice propulsora atrás daquele centro (Figura 21).

Pode dizer-se, em linguagem simples, que a hélice trator puxa o modelo, ao passo que o propulsor o empurra.



Hélice trator

Hélice propulsor

Figura 21 Tipos de hélices

Fonte: RODRIGUES, 2004

Segundo o sentido de rotação, as hélices dividem-se em esquerdos e direitos. Os primeiros rodam da esquerda para a direita, colocando-se o observador na frente do modelo, e os segundos no sentido inverso (Figura 22).

As hélices mais utilizadas são os direitos, em virtude de ser esse o sentido universal dado aos motores a pistão, que rodam ao invés dos ponteiros do relógio. Todavia, nos modelos com motor de elástico, já que a torção da meada se pode dar tanto num sentido como no outro, adotam-se às vezes hélices esquerdos.

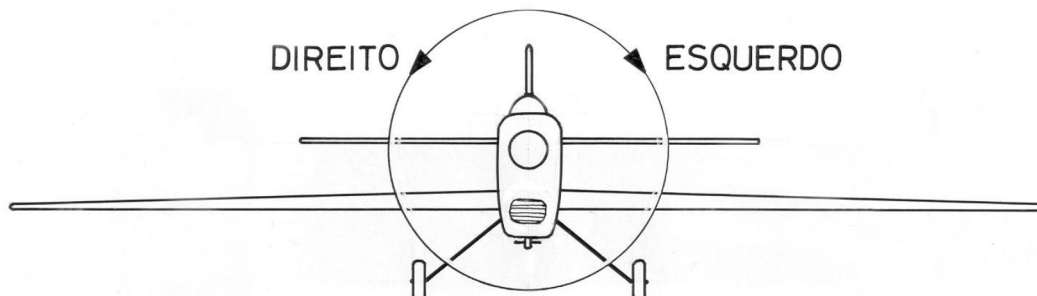


Figura 22 Sentidos de rotação do hélice

Fonte: RODRIGUES, 2004

4. AFINAÇÃO E CENTRAGEM

4.1 Afinação

A afinação, como também se costuma dizer, consiste em controlar todos os elementos do modelo, verificando se a construção representa fielmente as condições que o plano determina [3].

Assim, há que verificar com rigor a distribuição de massas e a disposição das superfícies aerodinâmicas, bem como o funcionamento de todos os dispositivos do modelo. Na realidade, verifica-se com frequência que um modelo, afinado logo depois de terminada a construção, dá um bom rendimento nos primeiros vôos, mas depois de algumas horas de campo perde todas as qualidades de equilíbrio, em virtude de ligeiros empenos sofridos pela ação do sol, do vento ou da umidade. Depois de se haver montado todo o modelo, deve examinar-se a posição relativa de todos os seus elementos.

Coloca-se o modelo sobre um estaleiro ou bancada, rigorosamente direitos (Figura 23), para verificação da simetria das diversas partes e controle das cotas mais importantes.

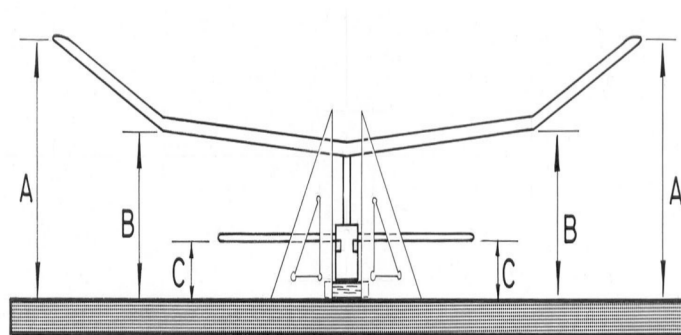


Figura 23 Afinação e centragem do modelo

Fonte: RODRIGUES, 2004

O modelo deve assentar-se sobre blocos de madeira, de modo que os lados da fuselagem fiquem normais ao estaleiro e a linha longitudinal da fuselagem paralela também ao plano do estaleiro.

Assim, visto o modelo de frente, verifica-se se os diedros estão simétricos e se a sua inclinação é a determinada no projeto; se a posição do estabilizador, em relação à asa é a conveniente; se a deriva mantém a indispensável perpendicularidade com o estabilizador; se os lados da fuselagem são normais ao estabilizador.

4.2 Centragem

De acordo com [3] centrar um modelo resume-se, fundamentalmente, em realizar pequenos ajustes nos calços da asa e do estabilizador, para uma dada posição do *C.G.*, ou vice-versa, de modo a obter o coeficiente de planeio mais conveniente à natureza do vôo que se pretende.

Um modelo que tenha sido bem afinado é, em geral, fácil de centrar. Não existem, no entanto, formulas decisivas para a obtenção de centragens ótimas. Num mesmo modelo podem aplicar-se centragens diferentes, com bons resultados.

Os primeiros ensaios de centragem devem ser feitos com tempo calmo, em terreno liso, livre de obstáculos; de preferência no interior de um hangar, ginásio ou salão de grandes proporções. Deve começar-se por lançar o modelo à mão (contra o vento, se for ao ar livre), segurando a fuselagem nas proximidades do *C.G.*

O modelo deve-se largar ligeiramente picado (nunca em posição cabrada), acompanhando-o com o braço de forma a imprimir-lhe uma velocidade semelhante à do vôo (Figura 24).



Figura 24 Lançamento do aeromodelo

Fonte: RODRIGUES, 2004

O modelo não deve ser atirado bruscamente. A melhor prática consiste em correr com ele, mantendo-o na atitude de vôo, acelerar-lhe ligeiramente a velocidade e largá-lo bem em frente, com o nariz um pouco inclinado para baixo.

Uma vez entregue aos seus próprios meios, ele iniciará um vôo, que porá em evidência os possíveis defeitos de centragem.

O vôo será correto, considerando-se o modelo centrado em planeio, quando a sua trajetória for o mais longa e direita possível, sem que o modelo ondule, oscile lateralmente ou se afunde.

Depois de o modelo se mostrar razoavelmente centrado, nos lançamentos à mão, deve tentar-se vôos mais altos. Junto ao solo, o modelo não manifesta com clareza as deficiências ligeiras que, porventura, ainda possua.

5. VEÍCULO AÉREO NÃO TRIPULADO

Um VANT ou VARP é mais conhecido como drone (zangão, em inglês), é todo e qualquer tipo de aeronave que não necessita de pilotos embarcados para ser guiada [4]. Esses aviões são controlados a distância por meios eletrônicos e computacionais, sob a supervisão e governo humanos, ou sem a sua intervenção, por meio de CLP.

Dotados de grande capacidade de geração de imagens e informações de inteligência, esses aviões – controlados por um sistema autônomo de navegação ou por controle remoto – fornecem aos policiais da fronteira imagens em tempo real, potencializando o combate ao crime organizado e ao tráfico de drogas. O VANT mostra que a tecnologia embarcada em pequenos e médios aviões pode ser usada para fins de defesa e segurança, utilizando as imagens para agilizar nas abordagens. Os VANTs também surgem como grandes aliados da defesa do meio ambiente e uma parceria inédita entre a Polícia Militar Ambiental do Estado de São Paulo e o Inova (São Carlos/SP), com apoio do INCT-SEC (ICMC/USP), está viabilizando o emprego dos aviões não-tripulados para o monitoramento ambiental em áreas rurais. É um apoio de peso ao trabalho das patrulhas terrestres que percorrem regiões imensas à procura de agressões ao meio ambiente. Estima-se que o país tenha hoje entre 65 a 70 milhões de hectares de áreas agrícolas e para fiscalizar essa área, o IBAMA conta com um contingente de apenas 1.319 fiscais espalhados por todo o território nacional. Verifica-se também um alto índice de impunidade, uma vez que flagrar a agressão ao meio ambiente é uma tarefa difícil e exige muitas vezes que os fiscais arrisquem suas vidas no exercício de sua função. A alta qualidade das imagens captadas permite aos aviões não-tripulados detectar com precisão e eficiência áreas desmatadas, diminuição

de APPs, atividades irregulares de mineração em leitos de rios, além de identificar crimes como, queimadas, pesca ilegal e corte de vegetação nativa. Se adaptado com sensores termais e multiespectrais, os VANTs podem revelar dados impossíveis de serem vistos a olho nu. Apenas para exemplificar, o uso de tecnologia infravermelha pode revelar, entre outros, a utilização de substâncias químicas empregadas em lavouras, a contaminação de cursos d'água por agentes químicos, o estresse hídrico de plantações e doenças em áreas agrícolas; é a tecnologia usada a favor da proteção do meio ambiente e da agricultura. É claro que falar de tecnologia envolve custos que nem sempre são baixos. A partir de trinta mil reais, é possível adquirir um VANT fabricado com tecnologia 100% nacional. Vale então a análise custo/benefício: em um país com as dimensões continentais como o Brasil a fiscalização por terra se torna muitas vezes inviável e com o emprego dos aviões não-tripulados, a localização das agressões ao meio ambiente e a identificação das operações de tráfico na fronteira podem ser mais frequentes, até em tempo real. Isso permite direcionar os recursos do estado de forma precisa, resultando na otimização das intervenções e na produção de provas concretas contra crimes que muitas vezes ficam impunes pela dificuldade de seu registro e sua mensuração.

6. ARDUINO

Conforme [5] é uma plataforma utilizada para criar protótipos eletrônicos com hardware livre, projetada com um micro controlador de placa única, com suporte de entrada/saída digitais ou analógicos, possui uma linguagem de programação padrão, a qual tem origem em Wiring que permite escrever programas para controlar aparelhos ligados a ele e é programado em C/C++. As vantagens do Arduino é a de criar ferramentas que são acessíveis, com baixo custo, flexíveis e fáceis de usar por profissionais ou amadores e pode ser usado para o desenvolvimento de objetos interativos independentes; uma placa Arduino é composta por um controlador, algumas linhas de E/S digital e analógica, além de uma interface serial ou USB, para interligar-se ao computador, que é usado para programar o Arduino e interagir em tempo real. A placa do Arduino em si não possui qualquer recurso de rede, porém é comum combinar um ou mais Arduinos, usando extensões chamadas de shields. A interface do programa é simples, podendo ser escrita em várias linguagens. A mais popular é a Processing, mas existem outras que podem comunicar-se com a conexão serial são: Super Collider,

ActionScript, Pure Data e Java. O projeto Arduino iniciou-se na cidade de Ivrea, na Itália, em 2005, com o objetivo de interagir em projetos escolares de forma a ter um preço menor que outros sistemas de prototipagem eletrônica disponíveis no mercado. O sucesso começou com o recebimento de uma menção honrosa na categoria Comunidades Digitais em 2006 pela Prix Ars Eletrônico, e a marca de mais de 50.000 placas vendidas até outubro de 2008.

Hoje, seu hardware é feito por meio de um micro controlador Atmel AVR, sendo que este não é uma exigência podendo ser estendido de acordo com as necessidades específicas de cada protótipo; considerando esta característica, muitos projetos semelhantes se baseiam em cópias alteradas com placas de desenvolvimento com objetivos específicos, que recebem seus próprios nomes. Além de o sistema poder ser montado pelo próprio usuário, os fabricantes possuem um serviço de venda do produto pré-montado, e também por distribuidores oficiais com locais de vendas em todo o mundo.

6.1 Hardware

De acordo com [5] sua placa possui elementos que servem de complemento para simplificar a programação e juntar-se com outros circuitos. A forma padrão que os conectores são expostos, concede o CPU a fazer a ligação entre outros módulos expansivos, conhecidos como Shields. Os Arduinos originais utilizam a série de chips mega AVR, especialmente os AT mega de 8 bits, AT mega 168, ATmega 328 e a AT mega1280; contudo muitos outros processadores foram utilizados por cópias deles. Quando seu software é utilizado, ele monta todas as placas sobre uma programação de ligação serial, mas a maneira que é instalada no hardware varia em cada versão. Suas placas seriais contêm um simples circuito inversor para converter entre os sinais dos níveis RS-232 e TTL. Modernamente existem algumas maneiras diferentes para realizar a transmissão dos dados, como por placas programáveis via USB, conectado através de um chip adaptador USB para Serial como o FTDI FT232. Diferentes placas como o Arduino Mini e o não oficial Boarduino, utilizam um módulo, cabo adaptador USB, Bluetooth ou outras maneiras de transmissão. A maioria dos pinos de entrada e saída dos micros controladores são para uso de outros circuitos. A versão Diecimila, contém 14 pinos digitais, seis das quais podem produzir sinais MLP, além de seis entradas

analógicas que ficam livres em cima da placa, através de conectores fêmeas de 0,1 polegadas (ou 0,25 centímetros).

O modelo Nano, Boarduino e placas compatíveis com estas, fornecem conectores machos na parte de baixo da placa, para serem ligados em protoboards.

6.2 Software

Segundo [5] é representado de forma esquemático para estabelecer a programação a profissional e a pessoas não acostumadas com o desenvolvimento de software; possui um editor de código com meios de realce de sintaxe, parênteses adequados e identificação automática, sendo capaz de juntar um conjunto de informações e carregar programas para a placa com um único clique. O software possui a capacidade de programar em C/C++, isso possibilita criar facilmente muitas operações de entrada e saída, tendo que definir apenas duas funções no pedido para fazer um programa funcional:

- `setup ()` – Incluída no início, que pode ser usada para inicializar as configurações;
- `loop ()` – Chamada para repetir um bloco de funções ou esperar até que seja desligada.

6.3 Aplicações

Como afirm [5] a principal finalidade do Arduino num sistema é facilitar a prototipagem, implementação ou emulação do controle de sistemas interativos, a nível doméstico, comercial ou móvel, da mesma forma que o CLP controla sistemas de funcionamento industriais. Com ele é possível enviar ou receber informações de basicamente qualquer sistema eletrônico, como identificar a aproximação de uma pessoa e variar a intensidade da luz do ambiente conforme a sua chegada. Ou abrir as janelas de um escritório de acordo com a intensidade da luz do sol e temperatura ambiente.

São utilizados em aplicações na área de impressão 3D, robótica, engenharia de transportes, engenharia agrônômica e musical.

6.4 Acessórios

Consoante com [5] o Arduino e suas cópias fazem uso de Shields (escudos, em inglês) que são placas de circuito estancadas, normalmente fixadas em cima do aparelho por meio de uma conexão mantida por pinos-conectores. São expansões que tornam acessível várias funções específicas, desde a manipulação de motores até sistemas de rede sem fio.

Exemplos:

Liquidware Touch Shield

Shields Extensores

XBee Shield

Liquidware Input Shield

Ethernet Shield

Tabela 6.4.1: Versões do dispositivo Arduino

Modelo	Descrição e tipo de conexão ao hospedeiro	Controlador
Serial Arduino	Serial DB9 para programação	ATmega8
Arduino Extreme	USB para programação	ATmega8
Arduino Mini	Versão em miniatura do Arduino utilizando montagem superficial	ATmega168
Arduino Nano	Versão menor que o <i>Arduino Mini</i> , energizado por USB e conectado por montagem superficial	ATmega168/328
LilyPadArduino	Projeto minimalista para aplicações portáteis, utilizando montagem superficial	ATmega168
Arduino NG	USB para programação	ATmega8

Arduino NG plus	USB para programação	ATmega168
Arduino BT	interface bluetooth para comunicação	ATmega168
ArduinoDiecimila	Interface USB	Atmega168 em um pacote <i>DIL28(foto)</i>
ArduinoDuemilanove	<i>Duemilanove</i> significa "2009" em italiano. É energizado via USB/DC, com alternância automática	Atmega168 (Atmega328 para a versão mais nova)
Arduino Mega	Montagem superficial	ATmega1280 para E/S adicionais e memória
Arduino Uno	Utiliza Atmega8U2 para driver conversor Serial-USB, ao invés do FTDI. "Uno" significa "Um" em italiano. Esta versão é considerada a versão 1.0 do projeto, sendo que as placas que sucederem a esta serão referenciadas.	ATmega328

7. PARÂMETROS DE PROJETO

Os parâmetros de projeto da escolha da aeronave foram:

- **Custo de fabricação:** Foi utilizado o isopor P3, assim como o DEPRON que são materiais leves, resistentes e de baixo custo.
- **Desempenho:** O projeto da aeronave, tamanho da asa, perfil, diedro, formato de fuselagem, posição e potência do motor, ângulo da asa, tamanho e peso da bateria. Foi adotado de acordo com pesquisa realizada por BONAVINA [6]
- **Capacidade de transporte de carga:** De acordo com a pesquisa de BONAVINA [], esta aeronave é capaz de transportar 400gr de carga útil. O arduino acoplado a uma sensor shield pesa aproximadamente 240gr.
- **Custo de operação:** Com a finalidade de obter um baixo custo de operação optou-se por utilizar um motor elétrico ao invés de um à combustão. O motor

elétrico apresenta baixo custo de manutenção, baixo custo de funcionamento, melhor rendimento energético, limpeza e simplicidade de comando.

8. MATERIAIS E MÉTODOS

De acordo com [6] inicialmente foi utilizado uma plataforma experimental de um veículo aéreo não tripulado que foi construída com a configuração de asa alta, maciça em isopor de alta densidade P3 com perfil simétrico e diedro nas pontas. Foi utilizado um motor elétrico sem escovas CF2822 com capacidade de empuxo de 8N com configuração de motor a trator, (Figura 25); a construção da fuselagem foi realizada com o material depron de 5mm.

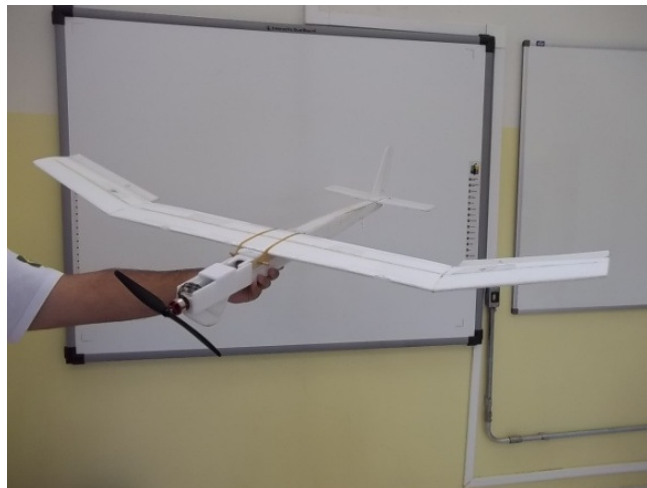


Figura 25 Configuração inicial da aeronave pré fabricada

Fonte: BONAVIDA, 2013

Foi adicionado um compartimento de carga para o alojamento da eletrônica embarcada (Arduino, sensores, bateria, rádio receptor), (Figura 26).



Figura 26 Aeronave modificada com o sistema de estabilização de vôo

Para realizar as modificações do aeromodelo foram utilizados os seguintes materiais e métodos.

8.1 Isopor

O isopor é rígido com varias formas e aplicações, e que se apresenta como uma forma moldada constituída por um aglomerado de grânulos [8]. É um material leve e com alta capacidade de isolamento térmico e baixo custo. No aeromodelismo é usado para se fabricar asas, profundos, estabilizadores, lemes e outros componentes aerodinâmicos com a vantagem da precisão e baixo peso. O Isopor é classificado, segundo a ABNT - NBR 11752, pela sua densidade, na asa do protótipo construído foi utilizado o isopor do tipo P3

P1 - 09 a 10,9Kg/m³

P2 - 11 a 12,9Kg/m³

P3 - 13 a 15,9Kg/m³

P4 - 16 a 19,9Kg/m³

P5 - 20 a 24,9Kg/m³

P6 - 25 a 29,9Kg/m³

P7 - 30 a 35,0Kg/m³

8.2 Depron

É uma placa de isopor, que foi utilizada como, matéria prima para fabricação do aeromodelo, é também muito utilizada como material base para colagem de

comunicações visuais e apresentações e também para recortes, decorações e montagens em comunicações e eventos. Nos aeromodelos, sua utilização é vantajosa devido ao baixo peso aliada à rigidez e estabilidade dimensional.

8.3 Cola Térmica

A cola térmica é um adesivo muito utilizado em produtos de artesanato e que é vendido em forma de bastão para ser usado com uma pistola para aplicação. A pistola de cola quente derrete os bastões para a aplicação da cola que pode ser usada em uma grande variedade de superfícies; o ponto negativo da cola térmica é ser densa, e o ponto positivo é que ela é mais fácil de ser manipulada, pois seca rápido.

8.4 Micro Servos

Os Micro servos 9g SG90TowerPro são excelentes para serem utilizados em aeromodelismo ou em projetos mecatrônicos.

Especificações:

- Voltagem de Operação: 3,0 – 7,2v
- Velocidade: 0,12seg/60Graus (4,8v) sem carga
- Torque: 1,2 Kg.cm (4,8v) e 1,6 Kg.cm (6,0v)
- Temperatura de Operação: -30C ~ +60C
- Dimensões: 32x30x12 mm
- Tipo de Engrenagem: Nylon
- Tamanho Cabo: 245 mm
- Peso: 9g

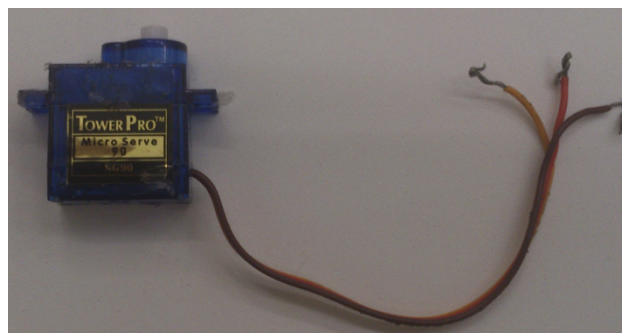


Figura 27 Micro Servo

8.5 Sensores GY-80

Placa de sensores Gy-80 para estabiliza o aeromodelo quando ele estiver em vôo. Essa placa possui giroscópios, acelerômetros, magnetômetro, barômetro e termômetro. Esta placa combina cinco sensores em um único pacote minúsculo.



Figura 28 Placa de sensores Gy-80

8.6 Micro controlador Arduino

Neste projeto será utilizado um micro controlador ArduinoATMega 2560, possui 54 pinos de entrada/saídas digitais (sendo 14 pinos podendo ser utilizadas como saídas PWM), 16 pinos de entradas analógicas, 4 UARTs a 16 MHz cristal oscilador, uma conexão USB, um conector de alimentação, para alimentação pode se utilizar a entrada USB ou ligá-lo a um adaptador AC ou uma bateria to-DC.

O controlador arduino utilizados nesse projeto contém as seguintes especificações:

- Micro controlador: ATmega1280
- Tensão de funcionamento: 5V
- Tensão de entrada (recomendado): 7-12V
- Tensão de entrada (limites): 6-20V
- Pinos E/S Digitais: 54 (15 oferecem saída PWM)
- Pinos de entrada analógica: 16
- Corrente DC por pino de E/S: 40 mA
- De corrente DC para 3.3V Pin: 50 mA
- Memória Flash: 128 KB, dos quais 4 KB usados por bootloader
- EEPROM: 4 KB

- Velocidade do relógio: 16 MHz

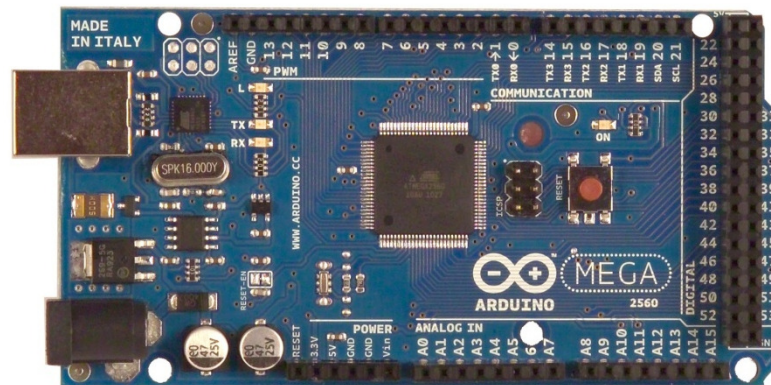


Figura 29 Micro controlador ArduinoATMega 2560

8.7 Bateria de Lítio Polímero

1. Capacidade típica: 1300mAh
2. Tensão: 3.70V
3. Tamanho: 7*30*60 milímetro
4. Espessura: 7mm
5. Resistência Nominal $\leq 50m\Omega$
6. Peso: 25.3g



Figura 30 Bateria do Aeromodelo

8.8 Motor

Motor usado é o Brushless Emax CF 2822 por ser um motor leve, pequeno e muito forte e muito usado na área de aeromodelismo. Suas características:

- N° de células: 2-3x Li-Poly
- RPM/v: 1200
- Diâmetro do estator: 22x10mm
- Diâmetro do eixo: 3mm
- Peso: 39g
- Peso recomendado do modelo: 200 - 600g
- Max Amps: 16A/60s

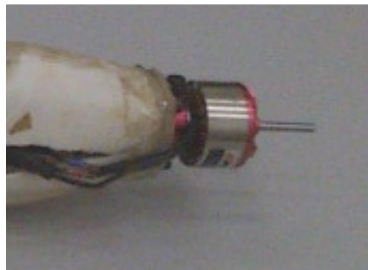


Figura 31 Motor do Aeromodelo

8.9 Receptor

Foi utilizado o receptor HK-TR6A V2 um comando eletrônico faz a interface com o aeromodelo, ou seja, é ele que “recebe e envia” todos os sinais de comando do RC para o aeromodelo.

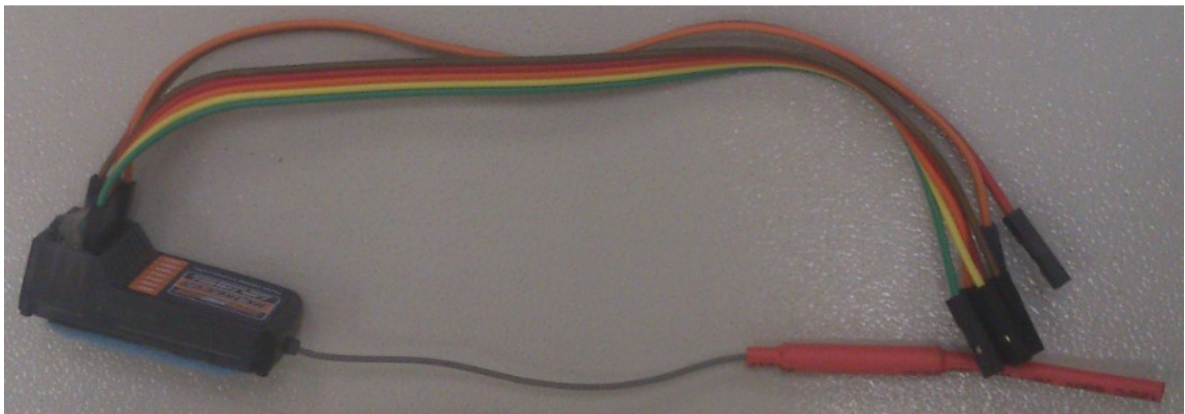
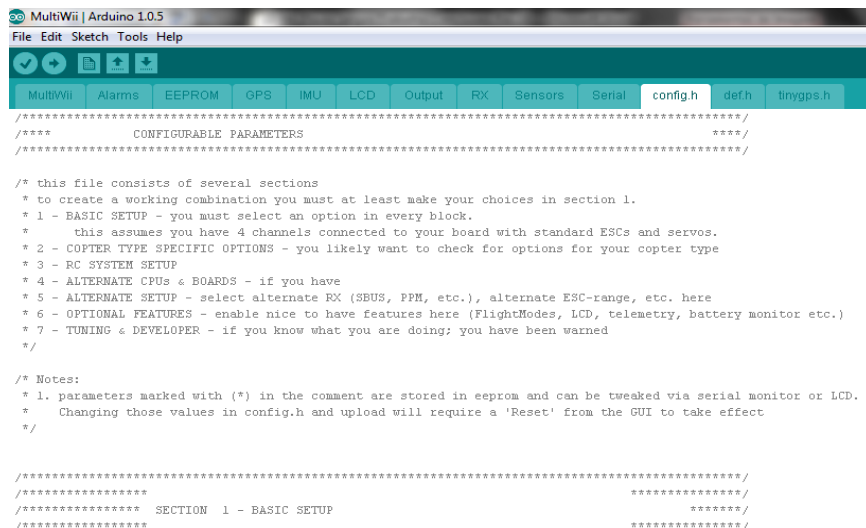


Figura 32 Receptor do aeromodelo

8.10 Programas utilizados na configuração do aeromodelo

Os programas utilizados para a realização desta pesquisa foi o MULTIWII 2.2 [14] (Figura 33) um código para arduino muito utilizado nesse meio em aeromodelos, drones, helicópteros entre outros, é um código completo, pois nele você apenas o configura de acordo com seu projeto idealizado e o WinGUI [15] (Figura 34), que é um software de aeromodelos utilizado para controle do comportamento do aeromodelo durante o vôo, visualização das leituras feitas pelos sensores, quando utilizado o sistema de GPS com esse software possibilita ao usuário um controle sobre o posicionamento aéreo do aeromodelo. São programações muito utilizadas para controle e configuração em VANT'S também, verificar ANEXO I.



```
MultiWii | Arduino 1.0.5
File Edit Sketch Tools Help
MultiWii Alarms EEPROM GPS IMU LCD Output RX Sensors Serial config.h def.h tinygps.h

/***** CONFIGURABLE PARAMETERS *****/

/* this file consists of several sections
 * to create a working combination you must at least make your choices in section 1.
 * 1 - BASIC SETUP - you must select an option in every block.
 *     this assumes you have 4 channels connected to your board with standard ESCs and servos.
 * 2 - COPTER TYPE SPECIFIC OPTIONS - you likely want to check for options for your copter type
 * 3 - RC SYSTEM SETUP
 * 4 - ALTERNATE CPUS & BOARDS - if you have
 * 5 - ALTERNATE SETUP - select alternate RX (SBUS, PPM, etc.), alternate ESC-range, etc. here
 * 6 - OPTIONAL FEATURES - enable nice to have features here (FlightModes, LCD, telemetry, battery monitor etc.)
 * 7 - TUNING & DEVELOPER - if you know what you are doing; you have been warned
 */

/* Notes:
 * 1. parameters marked with (*) in the comment are stored in eeprom and can be tweaked via serial monitor or LCD.
 * Changing those values in config.h and upload will require a 'Reset' from the GUI to take effect
 */

/***** SECTION 1 - BASIC SETUP *****/
```

Figura 33 MULTIWII 2.2

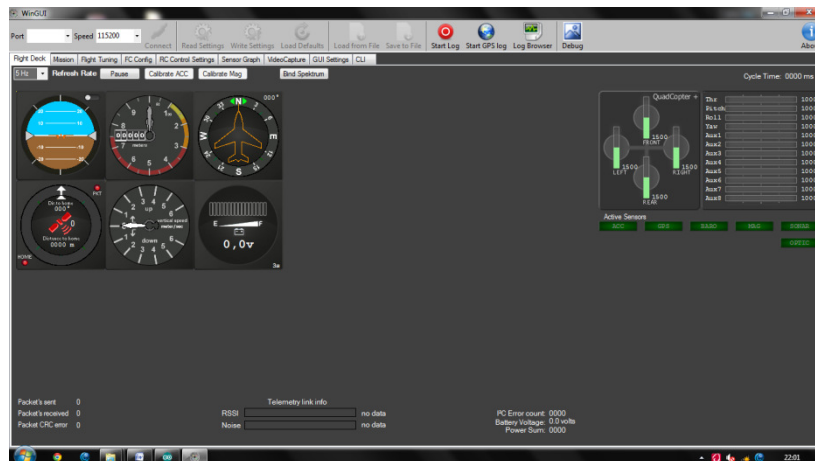


Figura 34 WinGUI

9. MODIFICAÇÃO FINAL DO PROJETO

Foram realizadas as modificações no aeromodelo (Figura 26) e instalados os equipamentos de estabilização de vôo. Foram realizados dois testes de vôo, onde o aeromodelo voou por cerca de dois minutos, mas apresentou grande instabilidade e dificuldade no controle do aeromodelo ocasionando a queda do mesmo. Devido à queda o aeromodelo ficou muito danificado tornando inviável sua utilização (Figura 35).

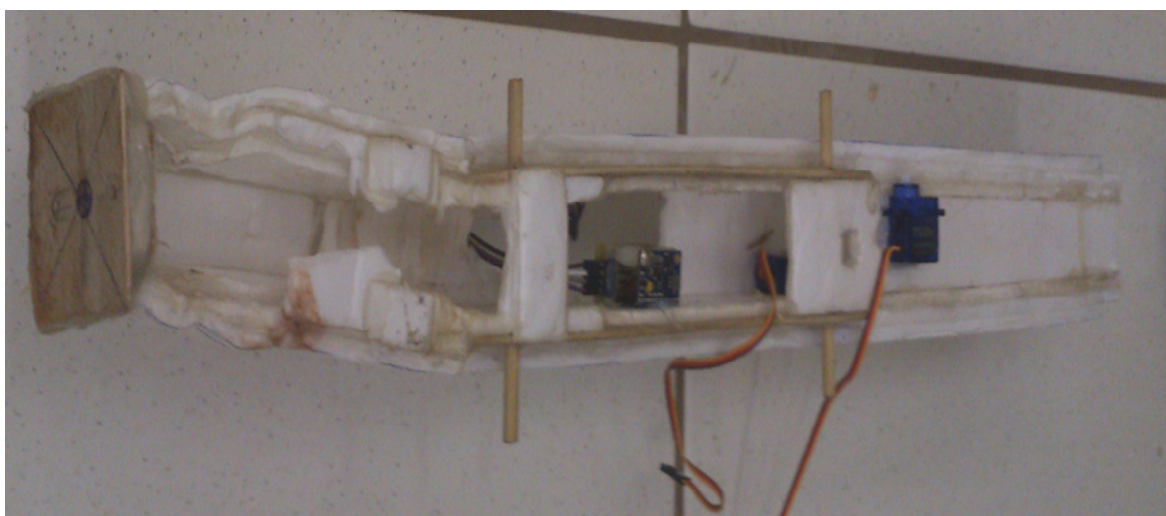


Figura 35 Aeromodelo Danificado

Foi utilizada uma nova fuselagem a qual possui a configuração de asa alta, maciça em isopor de alta densidade P3 com perfil assimétrico (plana na superfície inferior e convexa na superfície superior) com diedro e winglets nas pontas revestido com fita adesiva. (Figura 36).

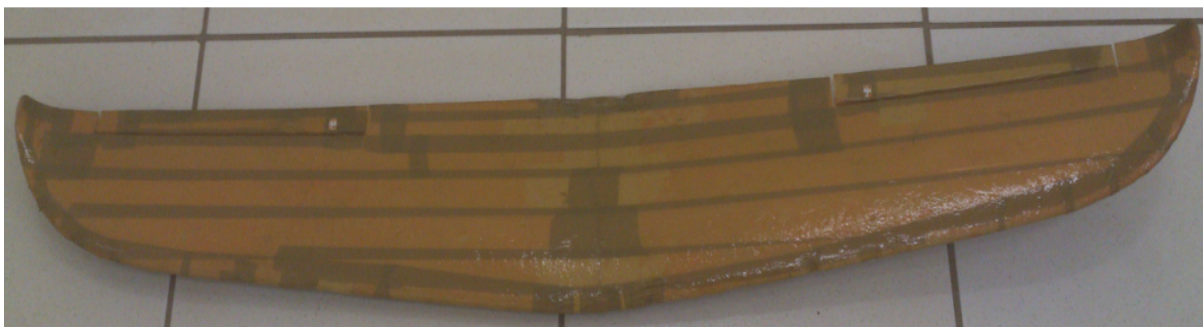


Figura 36 Perfil da asa do modelo atual

A fuselagem do protótipo (Figura 37) é pré fabricada com o material PVC, a qual é mais leve e resistente do que o primeiro modelo utilizado, o leme e o profundor foram construídos com depron de 5 mm, cola térmica e revestida com fita adesiva.

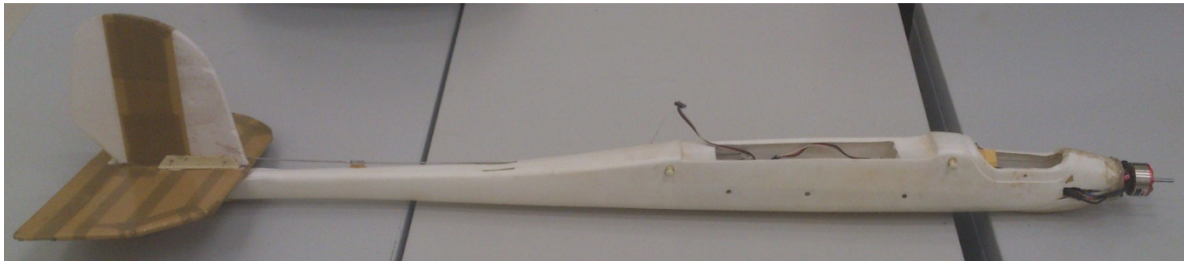


Figura 37 Fuselagem do modelo atual

A (figura38) mostra o aeromodelo atual utilizado na pesquisa e com todos os equipamentos citados anteriormente instalados para se obter a estabilização de vôo do aeromodelo.



Figura 38 Aeromodelo montado com os equipamentos

10. RESULTADOS E DISCUSSÃO

O primeiro teste foi realizado com a utilização do software WinGUI, o qual mostra quantitativamente os valores de deslocamento obtidos através dos sensores (acelerômetro) e mostra também as reações das superfícies de comando da aeronave.

O segundo teste foi uma tentativa de vôo com todo o sistema de estabilização embarcado, porém desligado. O avião decolou, mas devido à dinâmica de vôo instável a aeronave colidiu com o solo vindo assim a danificar o primeiro protótipo.

O conjunto software (programa) e hardware (arduino) instalado no protótipo aéreo apresentaram o funcionamento correto ao serem testados em solo.

Teste do aileron: ao inclinar a aeronave no sentido horário, olhando-a por trás, o aileron do lado esquerdo da asa dianteira deve deslocar-se para cima e o da direita, deve deslocar-se para baixo. Dessa maneira, o protótipo gera uma tendência a retornar a posição inicial e vice e versa como indica a figura abaixo.

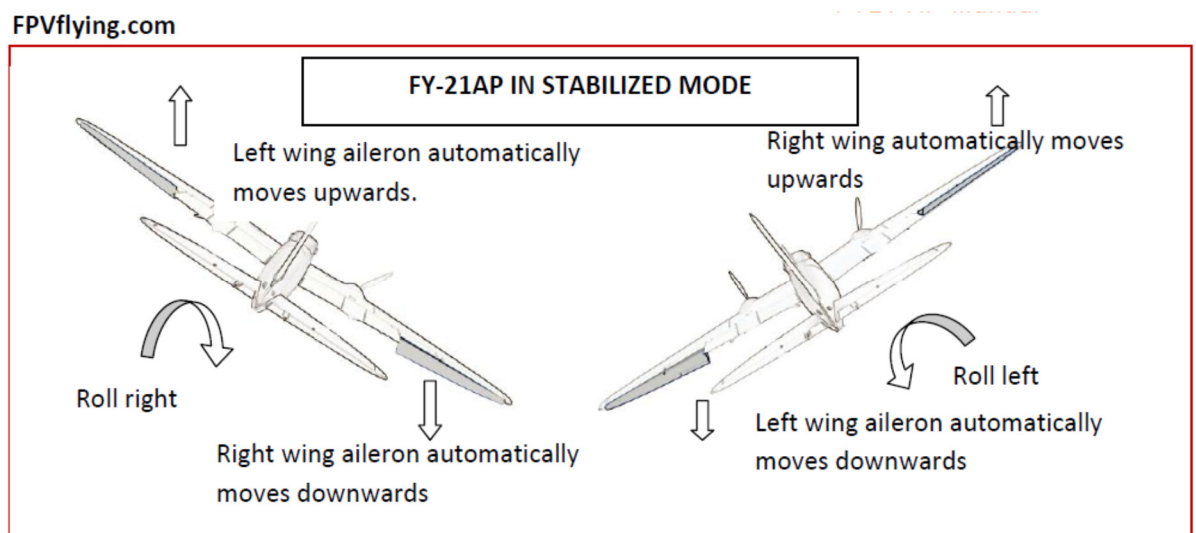


Figura 39 – Teste do aileron (fonte: FPVflying.com)

Teste do profundor: incline a aeronave com o nariz para cima. A aeronave deverá mover o profundor para baixo. E quando mover o nariz da aeronave para baixo, o profundor deverá se mover para cima, como mostra a (Figura 40).

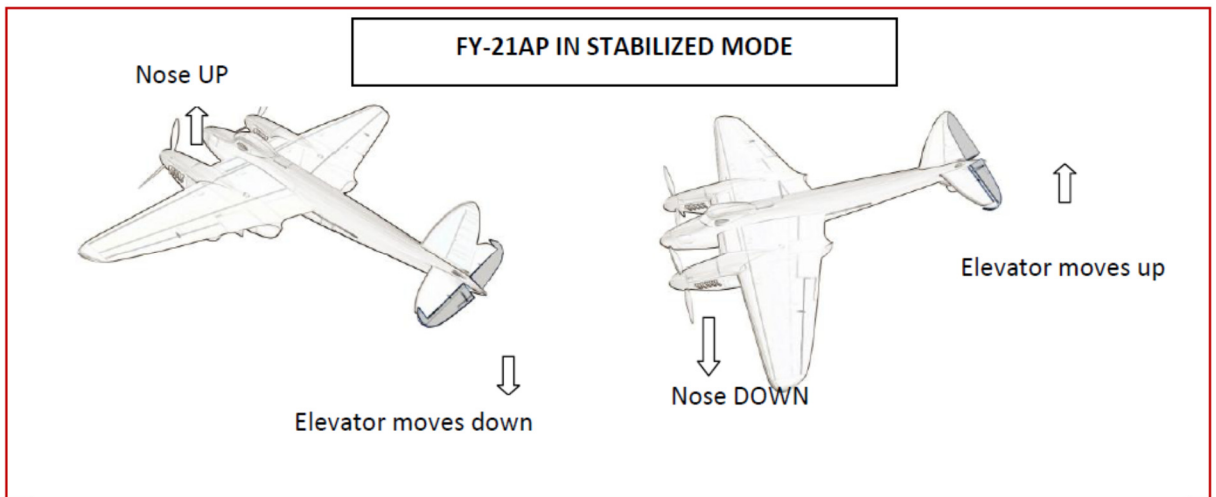


Figura 40- Teste do profundor (fonte: FPVflying.com)

Teste do leme: gire a aeronave no sentido horário sob o plano horizontal, e o leme deve se comportar de acordo com a (Figura 41):

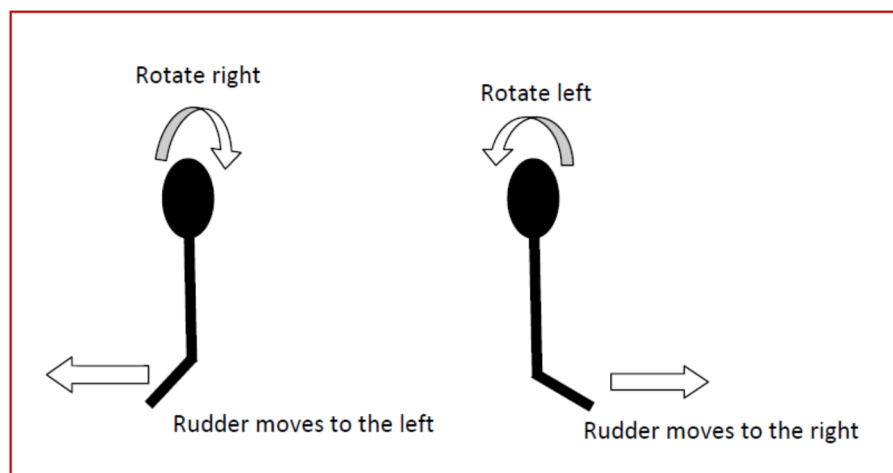


Figura 41- Teste do leme (fonte: FPVflying.com)

11. CONSIDERAÇÕES FINAIS

O acelerômetro de três eixos (sensores GY-80) realiza as aquisições dos sinais de entrada (sensores), o micro controlador o arduino MEGA2560 (hardware) processa os dados de entrada conforme o algoritmo de controle (software) e envia os sinais de saída aos micro servos os quais movimentam as superfícies de controle da aeronave.

Esse sistema controla o comportamento da aeronave nos três eixos de movimentação e detecta as mudanças de posição do modelo em relação ao plano

horizontal. Caso o comportamento da aeronave se altere, sinais de controle serão enviados as superfícies de comando, ailerons, profundor e leme de maneira a compensar as mudanças. Realizando esse processo constantemente estabiliza a aeronave, mantendo-a em um estado de equilíbrio contínuo, o que resulta num vôo suavemente nivelado. O sistema quando ativado é possível retomar o controle do vôo a qualquer momento, bastando movimentar as hastes do rádio transmissor e soltá-la até a posição neutra do controle, assim a aeronave retorna imediatamente ao modo de vôo nivelado. O vôo nivelado pode ser ativado ou desativado com um simples toque no botão do rádio transmissor.

Os testes realizados em solo apresentaram os resultados esperados. Agora serão necessários realizarem trabalhos futuros em relação aos ensaios de vôo. Assim como analisar o desempenho da aeronave no túnel de vento para futuras correções do algoritmo de controle e realizar ensaios de vôo e teste dos parâmetros de resposta do sistema.

12. REFERÊNCIAS BIBLIOGRÁFICAS

[1]WEBSITE WIKIPEDIA **Aeromodelismo**. Disponível em: <<http://pt.wikipedia.org/wiki/Aeromodelismo>>. Acesso em: 29 de abril 2014.

[2]WEBSITE WIKIPEDIA **Aeromodelismo**. Disponível em: <[http://pt.wikipedia.org/wiki/Confederação_Brasileira_de_Aeromodelismo](http://pt.wikipedia.org/wiki/Confedera%C3%A7%C3%A3o_Brasileira_de_Aeromodelismo)>. Acesso em: 29 de abril 2014.

[3]RODRIGUES, José Carlos; COLAREJO, José Manuel Padinha. Versão Digital Comemorativa dos 40 Anos de edição: **Aeromodelismo Teórico e Prático. C.A.L. – 2004**. Disponível em: <<http://www.e-voo.com/downloads/>>. Acesso em: 21 mai.2014.

[4]WEBSITE portal do agronegócio **Vants**. Disponível em: <<http://www.portaldoagronegocio.com.br/artigo/outros-2107/vants-uma-solucao-inteligente>>. Acesso em: 20 mai. 2014.

[5]WEBSITE WIKIPEDIA **Arduino**. Disponível em: <http://pt.wikipedia.org/wiki/Arduino> . Acesso em: 20 de abril 2014.

[5]WEBSITE **techtudo**. Disponível em:<<http://www.techtudo.com.br/noticias/noticia/2013/10/o-que-e-um-arduino-e-o-que-pode-ser-feito-com-ele.html>>. Acesso em: 02 de mai 2014.

[6] BONAVINA, Paulo da Silva; MORAES, Fernando de Haro. **Pesquisa e desenvolvimento de uma plataforma experimental de um veículo aéreo não tripulado**. 8 f. Relatório final Iniciação Científica. Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Araraquara. 2013.

[7]WEBSITE **filipeflop**. Disponível em:<<http://www.filipeflop.com/pd-71590-micro-servo-9g-sg90-towerpro>>. Acesso em: 02 de mai 2014.

[8] WEBSITE WIKIPEDIA **Poliestireno**. Disponível em: <<http://pt.wikipedia.org/wiki/poliestireno>>. Acesso em: 25 de abril 2014.

[9]WEBSITE **Arduino.** Disponível em: <<http://arduino.cc/en/Main/ArduinoBoardMega>>. Acesso em: 27 de mai 2014.

[10] WEBSITE asas elétricas **produtos.** Disponível em:<http://www.asaseletricas.com.br/loja/product_info.php?products_id=1574>.Acesso em: 20 mai. 2014.

[11]WEBSITEselfbuilt.Disponível em: <<http://selfbuilt.net/shop/gy-80-inertial-management-unit> traduzido com o google tradutor>. Acesso em: 15 abr 2014.

[12]WEBSITE portuguesebattery**produtos.** Disponível em:<http://portuguese.ppbattery.com/chinahigh_capacity_lithium_polymer_battery_703060_1300mah_3_7v-173412.html>. Acesso em: 12 mai. 2014.

[13]WEBSITE ehowcola quente. Disponível em:<http://www.ehow.com.br/informacoes-sobre-cola-quente-sobre_5816/>. Acesso em: 10 mai. 2014.

[14] WEBSITE code.google.com/p/multiwii/downloads/list Acesso em 13 de jun 2014.

[15] WEBSITE code.google.com/p/mw-wingui/downloads/list Acesso em 13 de jun 2014.

[16] WEBSITE www.fpvflying.com/ Acesso em 13 de jun 2014.

13. ANEXO-I

```
/*
*****
*****/
/****
*****/
CONFIGURABLE PARAMETERS
*****/
*****

/* this file consists of several sections
* to create a working combination you must at least make your choices in
section 1.
* 1 - BASIC SETUP - you must select an option in every block.
* this assumes you have 4 channels connected to your board with
standard ESCs and servos.
* 2 - COPTER TYPE SPECIFIC OPTIONS - you likely want to check for
options for your copter type
* 3 - RC SYSTEM SETUP
* 4 - ALTERNATE CPUS & BOARDS - if you have
* 5 - ALTERNATE SETUP - select alternate RX (SBUS, PPM, etc.), alternate
ESC-range, etc. here
* 6 - OPTIONAL FEATURES - enable nice to have features here
(FlightModes, LCD, telemetry, battery monitor etc.)
* 7 - TUNING & DEVELOPER - if you know what you are doing; you have
been warned
*/

/* Notes:
* 1. parameters marked with (*) in the comment are stored in eeprom and can
be tweaked via serial monitor or LCD.
* Changing those values in config.h and upload will require a 'Reset' from
the GUI to take effect
*/

/*
*****
*****
***** SECTION 1 - BASIC SETUP
*****
*****
*****

/*
***** The type of multicopter *****
*/
#define GIMBAL
#define BI
#define TRI
#define QUADP
#define QUADX
#define Y4
#define Y6
#define HEX6
#define HEX6X
#define HEX6H // New Model
#define OCTOX8
#define OCTOFLATP
#define OCTOFLATX
#define FLYING_WING
#define VTAIL4
#define AIRPLANE
#define SINGLECOPTER
#define DUALCOPTER
#define HELI_120_CCPM
#define HELI_90_DEG

/*
***** Motor minthrottle *****
*/
/* Set the minimum throttle command sent to the ESC (Electronic Speed
Controller)
This is the minimum value that allow motors to run at a idle speed */
#define MINTHROTTLE 1300 // for Turnigy Plush ESCs 10A
#define MINTHROTTLE 1120 // for Super Simple ESCs 10A
#define MINTHROTTLE 1064 // special ESC (simonk)
#define MINTHROTTLE 1050 // for brushed ESCs like ladybird
#define MINTHROTTLE 1150 // (*)

/*
***** Motor maxthrottle *****
*/
/* this is the maximum value for the ESCs at full power, this value can be
increased up to 2000 */
#define MAXTHROTTLE 1850

/*
***** Mincommand *****
*/
/* this is the value for the ESCs when they are not armed
in some cases, this value must be lowered down to 900 for some specific
ESCs, otherwise they failed to initiate */
#define MINCOMMAND 1000

/*
***** I2C speed
*****
*/
#define I2C_SPEED 100000L //100kHz normal mode, this value must be
used for a genuine WMP
#define I2C_SPEED 400000L //400kHz fast mode, it works only with
some WMP clones

/*
***** Internal i2c Pullups *****
*/
/* enable internal I2C pull ups (in most cases it is better to use external
pullups) */
#define INTERNAL_I2C_PULLUPS

/*
***** boards and sensor definitions *****
*****
*****
***** Combined IMU Boards
*****
/* if you use a specific sensor board:
please submit any correction to this list.
Note from Alex: I only own some boards, for other boards, I'm not sure,
the info was gathered via rc forums, be cautious */
#define FFIMUV1 // first 9DOF+baro board from Jussi, with
HMC5843 <- confirmed by Alex
#define FFIMUV2 // second version of 9DOF+baro board from Jussi,
with HMC5883 <- confirmed by Alex
#define FREEIMUV1 // v0.1 & v0.2 & v0.3 version of 9DOF board
from Fabio
#define FREEIMUV03 // FreeIMU v0.3 and v0.3.1
#define FREEIMUV035 // FreeIMU v0.3.5 no baro
#define FREEIMUV035_MS // FreeIMU v0.3.5_MS
<- confirmed by Alex
#define FREEIMUV035_BMP // FreeIMU v0.3.5 BMP
#define FREEIMUV04 // FreeIMU v0.4 with MPU6050, HMC5883L,
MS561101BA <- confirmed by Alex
#define FREEIMUV043 // same as FREEIMUV04 with final MPU6050
(with the right ACC scale)
#define NANOWII // the smallest multiwii FC based on MPU6050 +
pro micro based proc- confirmed by Alex
#define PIPO // 9DOF board from erazz
#define QUADRINO // full FC board 9DOF+baro board from witespy
with BMP085 baro- confirmed by Alex
#define QUADRINO_ZOOM // full FC board 9DOF+baro board from
witespy second edition
#define QUADRINO_ZOOM_MS // full FC board 9DOF+baro board from
witespy second edition <- confirmed by Alex
#define ALLINONE // full FC board or standalone 9DOF+baro board
from CSG_EU
#define AEROQUADSHIELDv2
#define ATAVRSBIN1 // Atmel 9DOF (Contribution by EOSBandi).
requires 3.3V power.
#define SIRIUS // Sirius Navigator IMU
<- confirmed by Alex
#define SIRIUSGPS // Sirius Navigator IMU using external MAG on
GPS board <- confirmed by Alex
#define SIRIUS600 // Sirius Navigator IMU using the WMP for the
gyro
#define SIRIUS_AIR // Sirius Navigator IMU 6050 32U4 from
MultiWiiCopter.com
#define SIRIUS_AIR_GPS // Sirius Navigator IMU 6050 32U4 from
MultiWiiCopter.com with GPS/MAG remote located
#define MINIWII // Jussi's MiniWii Flight Controller
<- confirmed by Alex
#define MICROWII // MicroWii 10DOF with ATmega32u4, MPU6050,
HMC5883L, MS561101BA from http://flyduino.net/
#define CITRUSv2_1 // CITRUS from qrcr.ca
#define CHERRY6DOFv1_0
#define DROTEK_10DOF // Drotek 10DOF with ITG3200, BMA180,
HMC5883, BMP085, w or w/o LLC
#define DROTEK_10DOF_MS // Drotek 10DOF with ITG3200, BMA180,
HMC5883, MS5611, LLC
#define DROTEK_6DOFv2 // Drotek 6DOF v2
#define DROTEK_6DOF_MPU // Drotek 6DOF with MPU6050
#define DROTEK_10DOF_MPU//
#define MONGOOSE1_0 // mongoose 1.0
http://store.ckdevices.com/
#define CRIUS_LITE // CriusMultiWii Lite
#define CRIUS_SE // CriusMultiWii SE
#define OPENLRSv2MULTI // OpenLRS v2 Multi Rc Receiver board
including ITG3205 and ADXL345
#define BOARD_PROTO_1 // with MPU6050 + HMC5883L + MS baro
#define BOARD_PROTO_2 // with MPU6050 + slave MAG3110 + MS
baro
#define GY_80 // Chinese 10 DOF with L3G4200D ADXL345
HMC5883L BMP085, LLC
#define GY_85 // Chinese 9 DOF with ITG3205 ADXL345
HMC5883L LLC
#define GY_86 // Chinese 10 DOF with MPU6050 HMC5883L
MS5611, LLC
#define GY_521 // Chinese 6 DOF with MPU6050, LLC
#define INNOVWORKS_10DOF // with ITG3200, BMA180, HMC5883,
BMP085 available here http://www.diymulticopter.com
#define INNOVWORKS_6DOF // with ITG3200, BMA180 available here
http://www.diymulticopter.com
#define MultiWiiMega // MEGA + MPU6050+HMC5883L+MS5611
available here http://www.diymulticopter.com
#define PROTO_DIY // 10DOF mega board
#define IOI_MINI_MULTIWII// www.bambucopter.com
#define Bobs_6DOF_V1 // BobsQuads 6DOF V1 with ITG3200 & BMA180
#define Bobs_9DOF_V1 // BobsQuads 9DOF V1 with ITG3200,
BMA180 & HMC5883L
#define Bobs_10DOF_BMP_V1 // BobsQuads 10DOF V1 with ITG3200,
BMA180, HMC5883L & BMP180 - BMP180 is software compatible with
BMP085
#define FLYDUINO_MPU
#define CRIUS_AIO_PRO_V1
#define DESQUARED6DOFV2GO // DEsquared V2 with ITG3200 only
#define DESQUARED6DOFV4 // DEsquared V4 with MPU6050
#define LADYBIRD
#define MEGAWAP_V2_STD // available here:
http://www.multircshop.com <- confirmed by Alex
```

```

//define MEGAWAP_V2_ADV
//define HK_MultiWii_SE_V2 // Hobbyking board with MPU6050 +
HMC5883L + BMP085
//define HK_MultiWii_328P // Also labeled "Hobbybro" on the back.
ITG3205 + BMA180 + BMP085 + NMC5883L + DSM2 Connector (Spektrum
Satellite)
//define RCNet_FC // RCNet FC with MPU6050 and MS561101BA
http://www.rcnet.com
//define RCNet_FC_GPS // RCNet FC with MPU6050 + MS561101BA
+ HMC5883L + UBXGPS http://www.rcnet.com
//define FLYDU_ULTRA // MEGA+10DOF+MT3339 FC

/***** independent sensors *****/
/* leave it commented if you already checked a specific board above */
/* I2C gyroscope */
//define WMP
//define ITG3200
//define L3G4200D
//define MPU6050 //combo + ACC

/* I2C accelerometer */
//define NUNCHUCK // if you want to use the nunckuk connected to a
WMP
//define MMA7455
//define ADXL345
//define BMA020
//define BMA180
//define NUNCHACK // if you want to use the nunckuk as a standalone
I2C ACC without WMP
//define LIS3LV02
//define LSM303DLx_ACC
//define MMA8451Q

/* I2C barometer */
//define BMP085
//define MS561101BA

/* I2C magnetometer */
//define HMC5843
//define HMC5883
//define AK8975
//define MAG3110

/* Sonar */ // for visualization purpose currently - no control code behind
//define SRF02 // use the Devantech SRF i2c sensors
//define SRF08
//define SRF10
//define SRF23

/* ADC accelerometer */ // for 5DOF from sparkfun, uses analog PIN
A1/A2/A3
//define ADCACC

/* enforce your individual sensor orientation - even overrides board specific
defaults */
//define FORCE_ACC_ORIENTATION(X, Y, Z) {accADC[ROLL] = Y;
accADC[PITCH] = -X; accADC[YAW] = Z;}
//define FORCE_GYRO_ORIENTATION(X, Y, Z) {gyroADC[ROLL] = -Y;
gyroADC[PITCH] = X; gyroADC[YAW] = Z;}
//define FORCE_MAG_ORIENTATION(X, Y, Z) {magADC[ROLL] = X;
magADC[PITCH] = Y; magADC[YAW] = Z;}

/* Board orientation shift */
/* If you have frame designed only for + mode and you cannot rotate FC
phisycally for flying in X mode (or vice versa)
* you can use one of this options for virtual sensors rotation by 45
deegres, then set type of multicopter according to flight mode.
* Check motors order and directions of motors rotation for matching with
new front point! Uncomment only one option! */
//define SENSORS_TILT_45DEG_RIGHT // rotate the FRONT 45
degres clockwise
//define SENSORS_TILT_45DEG_LEFT // rotate the FRONT 45
degres counterclockwise

/*****
*****/
/***** SECTION 2 - COPTER TYPE SPECIFIC OPTIONS *****/
*****/
/*****
*****/

/***** TRI *****/
#define YAW_DIRECTION 1
//define YAW_DIRECTION -1 // if you want to reverse the yaw correction
direction
/* you can change the tricopter servo travel here */
#define TRI_YAW_CONSTRAINT_MIN 1020
#define TRI_YAW_CONSTRAINT_MAX 2000
#define TRI_YAW_MIDDLE 1500 // (*) tail servo center pos. - use this for
initial trim; later trim midpoint via LCD

/***** BI *****/
/* you can change the bi-copter servo travel direction here */
//define BI_PITCH_DIRECTION 1
#define BI_PITCH_DIRECTION -1

/***** ARM/DISARM *****/
/* optionally disable stick combinations to arm/disarm the motors.
* In most cases one of the two options to arm/disarm via TX stick is
sufficient */
//define ALLOW_ARM_DISARM_VIA_TX_YAW
//define ALLOW_ARM_DISARM_VIA_TX_ROLL

/***** Airplane *****/
//define USE_THROTTLESERVO // For use of standard 50Hz servo on
throttle.
#define SERVO_RATES {100, 100, 100, 100, 100, 100, 100} //
Rates in 0-100%
#define SERVO_DIRECTION {-1, 1, 1, -1, -1, -1, -1} // Invert
servos by setting -1

//define FLAPPERONS AUX4 // Mix Flaps with Ailerons.
#define FLAPPERON_EP {1500, 1700} // Endpoints for flaps on a 2 way
switch else set {1020,2000} and program in radio.
//define FLAPPERON_EP {1200, 1500} // Or Flapperons up for CrowMix
#define FLAPPERON_INVERT {1, -1} // Change direction om flapperons
{Wing1, Wing2}

//define FLAPS AUX4 // Traditional Flaps on A2 invert with
SERVO_DIRECTION servo[2].
#define FLAP_EP {1500, 1900} // Endpoints for flaps on a 2 way
switch else set {1020,2000} and program in radio.

//define FLAPSPEED 3 // Make flaps move slowm Higher value
is Higher Speed.

/***** Common for Heli& Airplane *****/
//define D12_POWER // Use D12 on PROMINI to power sensors. Will
disable servo[4] on D12
#define SERVO_OFFSET {0, 0, 0, 0, 0, 0, 0} // (*) Adjust
Servo MID Offset & Swash angles
// Selectable channels:=
ROLL,PITCH,THROTTLE,YAW,AUX1,AUX2,AUX3,AUX4

/* Governor: attempts to maintain rpm through pitch and voltage changes
* predictive approach: observe input signals and voltage and guess
appropriate corrections.
* (the throttle curve must leave room for the governor, so 0-50-75-80-80 is
ok, 0-50-95-100-100 is _not_ ok.
* Can be toggled via aux switch.
*/
//define GOVERNOR_P 7 // (*) proportional factor. Higher value ->
higher throttle increase. Must be >=1; 0 = turn off
//define GOVERNOR_D 4 // (*) decay timing. Higher value -> takes
longer to return throttle to normal. Must be >=1;
//define GOVERNOR_R 10 // (*) voltage impact correction scale in 0.1
units. Higher value -> more compensation for voltage drops. normal is value
10 <=> 1.0; 0 is off

/*****
*****/
/***** SECTION 3 - RC SYSTEM SETUP *****/
*****/
/*****
*****/

/* note: no need to uncomment something in this section if you use a
standard receiver */

/***** special receiver types *****/
/*****

/***** PPM Sum Reciver *****/
/* The following lines apply only for specific receiver with only one PPM
sum signal, on digital PIN 2
Select the right line depending on your radio brand. Feel free to modify
the order in your PPM order is different */
//define SERIAL_SUM_PPM
//define SERIAL_SUM_PPM
ROLL,PITCH,THROTTLE,ROLL,AUX1,AUX2,AUX3,AUX4,8,9,10,11 //For
Graupner/Spektrum
//define SERIAL_SUM_PPM
ROLL,PITCH,THROTTLE,YAW,AUX1,AUX2,AUX3,AUX4,8,9,10,11 //For
Robe/Hitec/Futaba
//define SERIAL_SUM_PPM
ROLL,PITCH,YAW,THROTTLE,AUX1,AUX2,AUX3,AUX4,8,9,10,11 //For
Multiplex
//define SERIAL_SUM_PPM
PITCH,ROLL,THROTTLE,YAW,AUX1,AUX2,AUX3,AUX4,8,9,10,11 //For
some Hitec/Sanwa/Others

// Uncommenting following line allow to connect PPM_SUM receiver to
standard THROTTLE PIN on MEGA boards (eg. A8 in CRIUS AIO)
//define PPM_ON_THROTTLE

/***** Spektrum Satellite Reciver *****/
/* The following lines apply only for Spektrum Satellite Receiver
Spektrum Satellites are 3V devices. DO NOT connect to 5V!

```

For MEGA boards, attach sat grey wire to RX1, pin 19. Sat black wire to ground. Sat orange wire to Mega board's 3.3V (or any other 3V to 3.3V source).

```

For PROMINI, attach sat grey to RX0. Attach sat black to ground. */
// #define SPEKTRUM 1024
// #define SPEKTRUM 2048
// #define SPEK_SERIAL_PORT 1 // Forced to 0 on Pro Mini and single
serial boards; Set to your choice of 0, 1, or 2 on any Mega based board
(defaults to 1 on Mega).
// *****

```

```

// Defines that allow a "Bind" of a Spektrum or Compatible Remote
Receiver (aka Satellite) via Configuration GUI.
// Bind mode will be same as declared above, if your TX is capable.
// Ground, Power, and Signal must come from three adjacent pins.
// By default, these are Ground=4, Power=5, Signal=6. These pins are in
a row on most MultiWii shield boards. Pins can be overridden below.
// Normally use 3.3V regulator is needed on the power pin!! If your
satellite hangs during bind (blinks, but won't complete bind with a solid light),
go direct 5V on all pins.
// *****

```

```

// For Pro Mini, the connector for the Satellite that resides on the FTDI
can be unplugged and moved to these three adjacent pins.
// #define SPEK_BIND // Un-Comment for SpektrumSatellite Bind
Support. Code is ~420 bytes smaller without it.
// #define SPEK_BIND_GROUND 4
// #define SPEK_BIND_POWER 5
// #define SPEK_BIND_DATA 6

```

```

// ***** SBUS RECIVER *****

```

```

/* The following line apply only for Futaba S-Bus Receiver on MEGA
boards at RX1 only (Serial 1).
You have to invert the S-Bus-Serial Signal e.g. with a Hex-Inverter like
IC SN74 LS 04 */
// #define SBUS

```

```

/* ***** RC signal from the serial port via Multiwii Serial Protocol
***** */
// #define RCSERIAL

```

```

// ***** SECTION 4 - ALTERNATE CPUs & BOARDS *****
// *****
// *****
// *****

```

```

// ***** Teensy 2.0 Support *****
// *****
/* uncomment this if you use a teensy 2.0 with teensyduino
it needs to run at 16MHz */
// #define TEENSY20

```

```

// ***** Settings for ProMicro, Leonardo and other Atmega32u4 Boards
***** */
// *****

```

```

// ***** pin Layout *****
// *****
/* activate this for a better pinlayout if all pins can be used => not possible
on ProMicro */
// #define A32U4ALLPINS

```

```

// ***** PWM Setup *****
// *****
/* activate all 6 hardware PWM outputs Motor 5 = D11 and 6 = D13.
note: not possible on the sparkfunpromicro (pin 11 & 13 are not broken out
there)
if activated:
Motor 1-6 = 10-bit hardware PWM
Motor 7-8 = 8-bit Software PWM
Servos = 8-bit Software PWM
if deactivated:
Motor 1-4 = 10-bit hardware PWM
Motor 5-8 = 10-bit Software PWM
Servos = 10-bit Software PWM */
// #define HWPWM6

```

```

// ***** Aux 2 Pin *****
// *****
/* AUX2 pin on pin RX0 */
// #define RCAUX2PINRX0

```

```

/* aux2 pin on pin D17 (RXLED) */
// #define RCAUX2PIND17

```

```

// ***** Buzzer Pin *****
// *****
/* this moves the Buzzer pin from TX0 to D8 for use with ppm sum or
spectrum sat. RX (not needed if A32U4ALLPINS is active) */
// #define D8BUZZER

```

```

// ***** Promicro version related *****
// *****
/* Inverted status LED for Promicrover 10 */
// #define PROMICRO10

```

```

// ***** override default pin assignments *****
// *****

```

```

/* only enable any of this if you must change the default pin assignment, e.g.
your board does not have a specific pin */
/* you may need to change PINx and PORTx plus #shift according to the
desired pin! */
// #define OVERRIDE_V_BATPIN A0 // instead of A3 // Analog
PIN 3

```

```

// #define OVERRIDE_LEDPIN_PINMODE pinMode (A1, OUTPUT); //
use A1 instead of d13
// #define OVERRIDE_LEDPIN_TOGGLE PINC |= 1<<1; // PINB |=
1<<5; // switch LEDPIN state (digital PIN 13)
// #define OVERRIDE_LEDPIN_OFF PORTC &= ~(1<<1); //
PORTB &= ~(1<<5);
// #define OVERRIDE_LEDPIN_ON PORTC |= 1<<1; // was
PORTB |= (1<<5);

```

```

// #define OVERRIDE_BUZZERPIN_PINMODE pinMode (A2,
OUTPUT); // use A2 instead of d8
// #define OVERRIDE_BUZZERPIN_ON PORTC |= 1<<2 // PORTB
|= 1;
// #define OVERRIDE_BUZZERPIN_OFF PORTC &= ~(1<<2);
// PORTB &= ~1;

```

```

// ***** SECTION 5 - ALTERNATE SETUP *****
// *****
// *****

```

```

// ***** Serial com speed *****
/* This is the speed of the serial interfaces */
# define SERIAL0_COM_SPEED 115200
# define SERIAL1_COM_SPEED 115200
# define SERIAL2_COM_SPEED 115200
# define SERIAL3_COM_SPEED 115200

```

```

/* interleaving delay in micro seconds between 2 readings WMP/NK in a
WMP+NK config
if the ACC calibration time is very long (20 or 30s), try to increase this delay up
to 4000
it is relevant only for a conf with NK */
# define INTERLEAVING_DELAY 3000

```

```

/* when there is an error on I2C bus, we neutralize the values during a short
time. expressed in microseconds
it is relevant only for a conf with at least a WMP */
# define NEUTRALIZE_DELAY 100000

```

```

// ***** SECTION 6 - OPTIONAL FEATURES *****
// *****
// *****

```

```

// ***** continuous gyro calibration *****
/* Gyrocalibration will be repeated if copter is moving during calibration. */
# define GYROCALIBRATIONFAILSAFE

```

```

// ***** AP FlightMode *****
/* Temporarily Disables GPS_HOLD_MODE to be able to make it possible
to adjust the Hold-position when moving the sticks.*/
# define AP_MODE 40 // Create a deadspan for GPS.

```

```

// ***** Assisted AcroTrainer *****
// *****
/* Train Acro with auto recovery. Value set the point where ANGLE_MODE
takes over.
Remember to activate ANGLE_MODE first!...
A Value on 200 will give a very distinct transfer */
// #define ACROTRAINER_MODE 200 //
http://www.multiwii.com/forum/viewtopic.php?f=16&t=1944#p17437

```

```

// ***** Failsafe settings *****
/* Failsafe check pulses on four main control channels CH1-CH4. If the
pulse is missing or below 985us (on any of these four channels)
the failsafe procedure is initiated. After FAILSAFE_DELAY time from failsafe
detection, the level mode is on (if ACC or nunchuk is available).
PITCH, ROLL and YAW is centered and THROTTLE is set to
FAILSAFE_THROTTLE value. You must set this value to descending about
1m/s or so
for best results. This value is depended from your configuration, AUW and
some other params. Next, after FAILSAFE_OFF_DELAY the copter is
disarmed,

```

```

and motors is stopped. If RC pulse coming back before reached
FAILSAFE_OFF_DELAY time, after the small quard time the RC control is
returned to normal. */
//define FAILSAFE // uncomment to activate the
failsafe function
#define FAILSAFE_DELAY 10 // Guard time for failsafe
activation after signal lost. 1 step = 0.1sec - 1sec in example
#define FAILSAFE_OFF_DELAY 200 // Time for Landing before
motors stop in 0.1sec. 1 step = 0.1sec - 20sec in example
#define FAILSAFE_THROTTLE (MINTHROTTLE + 10) // (*) Throttle level
used for landing - may be relative to MINTHROTTLE - as in this case

/****** INFLIGHT ACC Calibration
******/
/* This will activate the ACC-Inflight calibration if unchecked */
//define INFLIGHT_ACC_CALIBRATION

/****** Disable WMP power pin
******/
/* disable use of the POWER PIN
(Already done if the option RCAUXPIN12 is selected) */
//define DISABLE_POWER_PIN

/****** OSD Switch
******/
/* This adds a box that can be interpreted by OSD in activation status (to
switch on/off the overlay for instance)
//define OSD_SWITCH

/****** TX-related
******/

/* introduce a deadband around the stick center
Must be greater than zero, comment if you dont want a deadband on roll,
pitch and yaw */
//define DEADBAND 6

/* defines the neutral zone of throttle stick during altitude hold, default
setting is
+/-40 uncommnd and change the value below if you want to change it. */
//define ALT_HOLD_THROTTLE_NEUTRAL_ZONE 40

/****** Display settings
******/
//define LCD_SERIAL_PORT 0 // must be 0 on Pro Mini and single
serial boards; Set to your choice on any Mega based board

//define SUPPRESS_OLED_I2C_128x64LOGO // suppress display of
OLED logo to save memory

/* double font height for better readability. Reduces visible #lines by half.
* The lower part of each page is accessible under the name of shifted
keyboard letter :
* 1 - 1 , 2 - @ , 3 - # , 4 - $ , 5 - % , 6 - ^ , 7 - & , 8 - * , 9 - (
* You must add both to your lcd.telemetry.* sequences
*/
//define DISPLAY_FONT_DSIZ //currently only aplicable for
OLED_I2C_128x64

/* style of display - AUTODETECTED via LCD_setting - only activate to
override defaults */
//define DISPLAY_2LINES
//define DISPLAY_MULTILINE
//define MULTILINE_PRE 2 // multiline configMenu # pref lines
//define MULTILINE_POST 6 // multiline configMenu # post lines
/****** Navigation
******/
/* keys to navigate the LCD menu */
#define LCD_MENU_PREV 'p'
#define LCD_MENU_NEXT 'n'
#define LCD_VALUE_UP 'u'
#define LCD_VALUE_DOWN 'd'

#define LCD_MENU_SAVE_EXIT 's'
#define LCD_MENU_ABORT 'x'

/****** LCD configuration menu
******/

/* uncomment this line if you plan to use a LCD or OLED for tweaking
parameters
*
http://www.multiwii.com/wiki/index.php?title=Extra_features#Configuration_Me
nu */
//define LCD_CONF

/* to include setting the aux switches for AUX1 -> AUX4 via LCD */
//define LCD_CONF_AUX

/* optional exclude some functionality - uncomment to suppress some
unwanted telemetry pages */
//define SUPPRESS_LCD_CONF_AUX34

/****** LCD telemetry
******/

```

```

/* to monitor system values (battery level, loop time etc. with LCD
* http://www.multiwii.com/wiki/index.php?title=LCD_Telemetry */

/****** Activation
******/
//define LCD_TELEMETRY

/* to enable automatic hopping between a choice of telemetry pages
uncomment this. */
//define LCD_TELEMETRY_AUTO "123452679" // pages 1 to 9 in
ascending order
//define LCD_TELEMETRY_AUTO "212232425262729" // strong
emphasis on page 2

/* manual stepping sequence; first page of the sequence gets loaded at
startup to allow non-interactive display */
//define LCD_TELEMETRY_STEP "0123456789" // should contain a 0 to
allow switching off.

/* optional exclude some functionality - uncomment to suppress some
unwanted telemetry pages */
//define SUPPRESS_TELEMETRY_PAGE_1
//define SUPPRESS_TELEMETRY_PAGE_2
//define SUPPRESS_TELEMETRY_PAGE_3
//define SUPPRESS_TELEMETRY_PAGE_4
//define SUPPRESS_TELEMETRY_PAGE_5
//define SUPPRESS_TELEMETRY_PAGE_6
//define SUPPRESS_TELEMETRY_PAGE_7
//define SUPPRESS_TELEMETRY_PAGE_8
//define SUPPRESS_TELEMETRY_PAGE_9

/****** RSSI
******/
//define RX_RSSI
//define RX_RSSI_PIN A3

/****** Buzzer
******/
#define BUZZER
//define RCOPTIONSBEEP // uncomment this if you want the buzzer
to beep at any rcOptions change on channel Aux1 to Aux4
//define ARMEDTIMEWARNING 330 // (*) Trigger an alarm after a certain
time of being armed [s] to save you lipo (if your TX does not have a
countdown)
//define PILOTLAMP //Uncomment if you are using a X-Arcraft Pilot
Lamp

/****** battery voltage monitoring
******/
/* for V BAT monitoring
after the resistor divisor we should get [0V;5V]->[0;1023] on analog V_BATPIN
with R1=33k and R2=51k
vbat = [0;1023]*16/VBATSACLE
must be associated with #define BUZZER ! */
//define VBAT // uncomment this line to activate the vbat code
#define VBATSACLE 131 // (*) change this value if readed Battery
voltage is different than real voltage
#define VBATNOMINAL 126 // 12.6V full battery nominal voltage - only
used for lcd.telemetry
#define VBATLEVEL_WARN1 107 // (*) 10.7V
#define VBATLEVEL_WARN2 99 // (*) 9.9V
#define VBATLEVEL_CRIT 93 // (*) 9.3V - critical condition: if vbat ever
goes below this value, permanent alarm is triggered
#define NO_VBAT 16 // (*) Avoid beeping without any battery

/****** powermeter (battery capacity monitoring)
******/

/* enable monitoring of the power consumption from battery (think of mAh)
allows to set alarm value in GUI or via LCD
Full description and howto here
http://www.multiwii.com/wiki/index.php?title=Powermeter
Two options:
1 - hard: - (uses hardware sensor, after configuration gives very good
results)
2 - soft: - (good results +-5% for plush and mystery ESCs @ 2S and 3S,
not good with SuperSimple ESC) */
//define POWERMETER_SOFT
//define POWERMETER_HARD
/* PLEVELSCALE is the step size you can use to set alarm */
#define PLEVELSCALE 50 // if you change this value for other granularity,
you must search for comments in code to change accordingly
/* larger PLEVELDIV will get you smaller value for power (mAh equivalent)
*/
#define PLEVELDIV 5000 // (*) default for soft - if you lower PLEVELDIV,
beware of overrun in uint32 pMeter
#define PLEVELDIVSOFT PLEVELDIV // for soft always equal to
PLEVELDIV; for hard set to 5000
#define PSENSORNUL 510 // (*) set to analogRead() value for zero
current; for I=0A my sensor gives 1/2 Vss; that is approx 2.49V;
#define PINT2mA 13 // (*) for telemetry display: one integer step on arduino
analog translates to mA (example 4.9 / 37 * 100

/****** altitude hold
******/

```

```

/* uncomment to disable the altitude hold feature.
 * This is useful if all of the following apply
 * + you have a baro
 * + want altitude readout
 * + do not use altitude hold feature
 * + want to save memory space
 */
#define SUPPRESS_BARO_ALTHOLD

/* Natural alt change for rapid pilots. It's temporary switch OFF the althold
when throttle stick is out of deadband defined with
ALT_HOLD_THROTTLE_NEUTRAL_ZONE
 * but if it's commented: Smooth alt change routine is activated, for slow auto
and aerophoto modes (in general solution from alexmos). It's slowly
increase/decrease
 * altitude proportional to stick movement (+/-100 throttle gives about +/-50
cm in 1 second with cycle time about 3-4ms)
 */
#define ALTHOLD_FAST_THROTTLE_CHANGE

/***** altitude variometer *****/
/*****

/* enable to get audio feedback upon rising/falling copter/plane.
 * Requires a working baro.
 * For now, Output gets sent to an enabled vt100 terminal program over the
serial line.
 * choice of two methods (enable either one or both)
 * method 1 : use short term movement from baro ( bigger code size)
 * method 2 : use long term observation of altitude from baro (smaller code
size)
 */
#define VARIOMETER_12 // possible values: 12 = methods 1 & 2 ; 1
= method 1 ; 2 = method 2
#define SUPPRESS_VARIOMETER_UP // if no signaling for up
movement is desired
#define SUPPRESS_VARIOMETER_DOWN // if no signaling for down
movement is desired
#define VARIOMETER_SINGLE_TONE // use only one tone (BEL);
necessary for non-patched vt100 terminals

/***** board naming *****/
/*****

/*
 * this name is displayed together with the MultiWii version number
 * upon powerup on the LCD.
 * If you are without a DISPLAYD then You may enable LCD_TTY and
 * use arduino IDE's serial monitor to view the info.
 *
 * You must preserve the format of this string!
 * It must be 16 characters total,
 * The last 4 characters will be overwritten with the version number.
 */
#define BOARD_NAME "MultiWii V-.-"
// 123456789.123456

/***** Support multiple configuration profiles in EEPROM *****/
/*****
#define MULTIPLE_CONFIGURATION_PROFILES

/*****
/*****
/***** SECTION 7 - TUNING & DEVELOPER *****/
/*****
/*****

/***** Experimental: force a stable, fixated (high) cycle time *****/
/*****
/* when activated, the displayed cycle time in GUI will not be correct.
 * Tunable via LCD config menu.
 * value of 0 turns the feature off.
 */
#define CYCLETIME_FIXATED 9000 // (*)

/***** special ESC with extended range [0-2000] microseconds *****/
/*****
#define EXT_MOTOR_RANGE

/***** motor, servo and other presets *****/
/*****
/* motors will not spin when the throttle command is in low position
this is an alternative method to stop immediately the motors */
#define MOTOR_STOP

/* some radios have not a neutral point centered on 1500. can be changed
here */
#define MIDRC 1500

/***** Servo Refreshrates *****/
/* Default 50Hz Servo refresh rate*/
#define SERVO_RFR_50HZ

/* up to 160Hz servo refreshrate ..works with the most analog servos*/
#define SERVO_RFR_160HZ

/* up to 300Hz refreshrate it is as fast as possible (100-300Hz depending on
the count of used servos and the servos state).
for use with digital servos
dont use it with analog servos! they may get damage. (some will work but be
careful) */
#define SERVO_RFR_300HZ

/***** HW PWM Servos *****/
/* HW PWM Servo outputs for ArduinoMega..moves:
Pitch = pin 44
Roll = pin 45
CamTrig = pin 46
SERVO4 = pin 11 (assigned to PPM or SPECTRUM CH9 on copter
configuration)
SERVO5 = pin 12 (assigned to PPM or SPECTRUM CH10 on copter
configuration)
this option disable other software PWM's for servos - only five hardware
controlled servos available
 */
#define MEGA_HW_PWM_SERVOS

/***** Serial command handling - MSP and other *****/
/*****

/* to reduce memory footprint, it is possible to suppress handling of serial
commands.
 * This does _not_ affect handling of RXserial, Spektrum or GPS. Those will
not be affected and still work the same.
 * Enable either one or both of the following options */

/* Remove handling of all commands of the New MultiWii Serial Protocol.
 * This will disable use of the GUI, winGUI, android apps and any other
program that makes use of the MSP.
 * You must find another way (like LCD_CONF) to tune the parameters or
live with the defaults.
 * If you run a LCD/OLED via i2c or serial/Bluetooth, this is safe to use */
#define SUPPRESS_ALL_SERIAL_MSP // saves approx 2700 bytes

/* Remove handling of other serial commands.
 * This includes navigating via serial the lcd.configuration menu,
lcd.telemetry and permanent.log .
 * Navigating via stick inputs on tx is not affected and will work the same. */
#define SUPPRESS_OTHER_SERIAL_COMMANDS // saves approx 0 to
100 bytes, depending on features enabled

/***** diagnostics *****/
/*****

/* to log values like max loop time and others to come
logging values are visible via LCD config
set to 1, enable 'R' option to reset values, max current, max altitude
set to 2, adds min/max cycleTimes
set to 3, adds additional powerconsumption on a per motor basis (this uses
the big array and is a memory hog, if POWERMETER <> PM_SOFT) */
#define LOG_VALUES 1

/* Permanent logging to eeprom - survives (most) upgrades and parameter
resets.
 * used to track number of flights etc. over lifetime of controller board.
 * Writes to end of eeprom - should not conflict with stored parameters yet.
 * Logged values: accumulated lifetime, #powercycle/reset/initialize events,
#arm events, #disarm events, last armedTime,
 * #failsafe@disarm, #2c_errs@disarm
 * To activate set to size of eeprom for your mcu: promini 328p: 1023 ; 2560:
4095.
 * Enable one or more options to show the log
 */
#define LOG_PERMANENT 1023
#define LOG_PERMANENT_SHOW_AT_STARTUP // enable to display
log at startup
#define LOG_PERMANENT_SHOW_AT_L // enable to display log when
receiving 'L'
#define LOG_PERMANENT_SHOW_AFTER_CONFIG // enable to display
log after exiting LCD config menu
#define LOG_PERMANENT_SERVICE_LIFETIME 36000 // in seconds;
service alert at startup after 10 hours of armed time

/* to add debugging code
not needed and not recommended for normal operation
will add extra code that may slow down the main loop or make copter non-
flyable */
#define DEBUG

/* Use this to trigger LCD configuration without a TX - only for debugging -
do NOT fly with this activated */
#define LCD_CONF_DEBUG

/* Use this to trigger telemetry without a TX - only for debugging - do NOT
fly with this activated */
#define LCD_TELEMETRY_DEBUG //This form rolls between all
screens, LCD_TELEMETRY_AUTO must also be defined.
#define LCD_TELEMETRY_DEBUG 6 //This form stays on the screen
specified.

/* Enable string transmissions from copter to GUI */

```



```
//#define DEBUGMSG
```

```
/*  
**** Regression testing ****  
*/
```

/* for development only:
to allow for easier and reproducible config sets for test compiling, different sets
of config parameters are kept
together. This is meant to help detecting compile time errors for various
features in a coordinated way.

It is not meant to produce your flying firmware

To use:

- do not set any options in config.h,
 - enable with #define COPTERTEST 1, then compile
 - if possible, check for the size
- repeat with other values of 2, 3, 4 etc.

*/

```
//#define COPTERTEST 1
```

```
/*  
****  
**** END OF CONFIGURABLE PARAMETERS  
****  
*/
```