



Projecte Final de Carrera
ENGINYERIA INDUSTRIAL

Control d'un Quadcopter

Memòria

Autor: Gabriel de la Cal Mendoza
Director: Manel Velasco Garcia
Convocatòria: Data a presentar

Escola Tècnica Superior d'Enginyeria Industrial de Barcelona



Resum

Índice

Resum	1
1. Prefaci	5
1.1. Motivació	5
1.2. Requeriments previs	5
2. Introducció	6
2.1. Estudi de l'art	6
2.2. Objectius del projecte	6
3. Definició del model	7
3.1. Definició de les variables	7
3.2. Obtenció del model	8
3.3. Representació del model amb Matlab	10
4. Disseny del controlador	11
5. Implementació del control	12
6. Construcció del Quadcopter	13
6.1. Descripció dels components	13
6.2. Muntatge	17
7. Anàlisi econòmic	18

Índice de figures

1.	Marcos de referencia en el quadcopter	7
2.	Raspberry Pi a utilitzar	13
3.	Conectors de la Raspberry Pi	13
4.	IMU MPU-6050	14
5.	Emissora i Receptor	14
6.	Connexió per I2C entre MPU-6050 i RPi	21
7.	Detecció del sensor MPU-6050	22
8.	Bancada per a l'estudi del motor	24
9.	Montatge per a l'estudi del motor	24
10.	Relació entre la força (en grams) i el PWM	25
11.	Relació entre la força (en grams) i el PWM	26

1. Prefaci

1.1. Motivació

La principal motivació d'aquest projecte és la d'aplicar els coneixements bàsics adquirits en aquests 6 anys de carrera, i més en particular en l'àrea del control en haver fet l'intensificació d'automàtica.

Llavors, a l'hora de plantear un tema per al projecte ràpidament va sorgir la idea de realitzar-lo sobre el control d'un sistema, i més en particular sobre un que estigués actualment emergent tant en mercats com en el camp de l'investigació.

D'entre les diferents alternatives, un quadcopter és la més atractiva tant per la seva simplicitat constructiva com pel no excessiu cost. Existeixen actualment en el mercat infinitat de proveïdors dels components necessaris per a construir un quadcopter, amb un gran ventall d'opcions per a escollir cada element com motors, bateries, electrònica, etc.

Les possibles aplicacions són nombroses, tant que encara no s'han ni tan sols explotat totes les possibles. Com a exemple: vigilància de superfícies obertes, transport de petits paquets, eina d'oci, entre d'altres.

El preu d'un quadcopter no es gaire elevat ja que un comercial com el *ARDrone2.0* oscil·la els 300 € i en construir un amb menys prestacions no hauria de superar aquest cost.

1.2. Requeriments previs

Com a requeriments previs és necessari tenir certs coneixements mínims en automàtica per tal de controlar el quadcopter, així com l'inquietut per aprendre el que sigui necessari per a complir, en la mesura del possible, els objectius inicials del present projecte com superar les dificultats que hagin sorgit, tenint en compte que el fi no justifica els medis.

2. Introducció

Un Quadcopter es un vehicle volador no tripulat (*UnmannedAerialVehicle*) que es caracteritza per tenir quatre rotors a mode d'actuadors en comptes de dos com en el cas dels helicòpters. Aquest tipus d'autogir intenta obtenir una flotabilitat estable i vol precís balancejant les forces produïdes pels quatre motors.

Un dels avantatges que s'obtenen amb aquest canvi és la major capacitat de càrrega ja que es tenen 4 motors per a soportar el pes. L'estabilitat del vehicle millora en permetre aterratges i enlairaments verticals amb una major maniobrabilitat. També pot treballar en àrees de difícil accés o més agressives, com amb pluja i vent.

2.1. Estudi de l'art

2.2. Objectius del projecte

Esquemàticament es pot representar com una estructura en *X* amb el seu centre coincidint amb el centre de masses i quatre actuadors a les puntes de cada braç, tots ells apuntant en la mateixa direcció y sentit

descripción por encima

historia

estudio del arte

futuras aplicaciones

3. Definició del model

3.1. Definició de les variables

Per a caracteritzar la planta amb la que es treballarà, és necessari obtenir un model del quadcopter. Les constants pròpies del model es deixaran en forma de paràmetres a calibrar una vegada es tingui l'objecte físic. D'aquesta manera el model serà general per a tot quadcopter que comparteixi la mateixa família de paràmetres. És necessari considerar dos marcs de referència: l'inercial format pels eixos x, y, z i el del cos (Body) format pels eixos x_B, y_B, z_B . El primer té la perspectiva de l'observador en terra, estàtic, mentre que el segon és solidari a l'estructura. Segons l'orientació del eixos del cos amb aquesta referència es poden donar els següents dos casos:

- *Crosstype*: Els eixos de coordenades coincideixen amb els braços de l'estructura ja que es tenen els actuadors a les puntes de cada braç.
- *X-type*: Els eixos i l'estructura formen 45° . Es tenen llavors dos motors al davant i dos al darrere.

Per ser més usual la primera opció, es decideix utilitzar la configuració *Crosstype* tal i com es té en la figura 1.

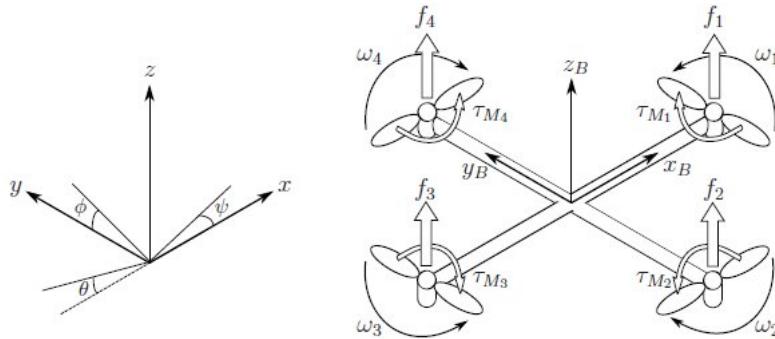


Figura 1: Marcs de referència en el quadcopter

Es suposa que l'objecte és un rotor esfèric, i per tant el seu tensor d'inèrcia és diagonal:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (1)$$

Es defineix la posició linear absoluta amb les coordenades x, y, z pel vector ξ i igualment per a la posició angular a partir de η segons:

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad q = \begin{bmatrix} \xi \\ \eta \end{bmatrix} \quad (2)$$

on ϕ és l'angle de capcineig (Pitch), θ és el de balanceig (Roll) i ψ el de guiñada (Yaw). Per a l'orientació angular entre els dos marcs es té un sistema de referència amb angles Tait Bryan, on la matriu de transformació és:

$$R = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta C_\phi & C_\theta C_\phi \end{bmatrix} \quad (3)$$

amb $C_\phi = \cos(\phi)$ i $S_\phi = \sin(\phi)$.

Les velocitats lineals en el marc de referència del cos (Body Frame) es representen amb el vector v_B y les velocitats angulars amb γ segons:

$$v_B = \begin{bmatrix} v_{x,b} \\ v_{y,b} \\ v_{z,b} \end{bmatrix} \quad \gamma = \begin{bmatrix} p \\ n \\ r \end{bmatrix} \quad (4)$$

En canvi, les velocitats en el marc de referència inercial (Inertial Frame) es representen per $\dot{\eta}$ per a les velocitats lineals i per $\dot{\xi}$ per a les angulars:

$$\dot{\xi} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad \dot{\eta} = \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} \quad (5)$$

Ja que la derivada dels angles ϕ , θ i ψ no és el vector de velocitats angulars és necessari tenir un canvi de base per relacionar el marc de referència inercial al del cos amb la matriu W_η :

$$W_\eta = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix} \quad \text{amb} \quad \gamma = [W_\eta] \dot{\eta} \quad (6)$$

Les forces de sustentació i moments generats pels quatre actuadors són f_1, f_2, f_3, f_4 i w_1, w_2, w_3, w_4 respectivament. Seguint l'orientació de la figura 1, per tal de poder anular els moments produïts en l'eix z_B (en el marc del cos) el sentit de gir dels actuadors 4 i 2 és en el de les agulles del rellotge (clockwise) i els dels 1 i 3 en sentit contrari (counterclockwise).

Interessa conèixer quina força i moment aportarà cada motor per a una velocitat angular coneiguda. Es tenen les següents relacions per a cada actuador:

$$\begin{aligned} f_i &= kw_i^2 \\ \tau_{M_i} &= bw_i^2 \end{aligned} \quad (7)$$

Per tant l'empenta total T_B proporcionada en la direcció z_B i els moments generats τ_B pels motors és:

$$T_B = k \left(\sum_{i=1}^4 w_i^2 \right) e_{z_B} = \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 w_i^2 \end{bmatrix} \quad \tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} lk(w_4^2 - w_2^2) \\ lk(w_3^2 - w_1^2) \\ \sum_{i=1}^4 \tau_{M_i} \end{bmatrix} \quad (8)$$

3.2. Obtenció del model

Les equacions que governen el sistema s'obtenen amb el mètode de Euler-Lagrange, pel que es parteix obtenint el Lagrangian del sistema:

$$\mathcal{L} = E_{cinetica} - E_{potencial} = (E_{translacio} + E_{rotacio}) - E_{potencial} \quad (9)$$

Substituïnt cada component per la seva expressió:

$$\mathcal{L}(q, \dot{q}) = \frac{m}{2} \dot{\xi}^T \dot{\xi} + \frac{1}{2} \gamma^T I \gamma - mgz \quad (10)$$

Es troba el vector de forces i moments com:

$$F = \begin{bmatrix} f \\ \tau_B \end{bmatrix} = \frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} \quad (11)$$

$$\text{amb} \quad q = [x \ y \ z \ \phi \ \theta \ \psi]^T \quad \text{i} \quad \dot{q} = [\dot{x} \ \dot{y} \ \dot{z} \ \dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T.$$

En calcular F és necessari fer el canvi de variables de γ a $\dot{\eta}$ amb el canvi de base $\dot{\eta} = [W_\eta]^{-1} \gamma$ per tal de poder derivar el Lagrangià respecte \dot{q} , que són les variables pròpies del marc de referència inercial

$$\frac{1}{2} \gamma^T I \gamma = \frac{1}{2} (W_\eta \dot{\eta})^T I (W_\eta \dot{\eta}) = \frac{1}{2} \dot{\eta}^T (W_\eta^T I W_\eta) \dot{\eta} = \frac{1}{2} \dot{\eta}^T J \dot{\eta} \quad (12)$$

on la matriu J queda com

$$J = W_\eta^T I W_\eta = \begin{bmatrix} I_{xx} & 0 & -I_{xx} S_\theta \\ 0 & I_{yy} C_\phi^2 + I_{zz} S_\phi^2 & (I_{yy} - I_{zz}) C_\phi S_\phi C_\theta \\ -I_{xx} S_\theta & (I_{yy} - I_{zz}) C_\phi S_\phi C_\theta & I_{xx} S_\theta^2 + I_{yy} S_\phi^2 C_\theta^2 + I_{zz} C_\phi^2 C_\theta^2 \end{bmatrix} \quad (13)$$

i per tant el Lagrangià queda com:

$$\mathcal{L}(q, \dot{q}) = \frac{m}{2} \dot{\xi}^T \dot{\xi} + \frac{1}{2} \dot{\eta}^T J \dot{\eta} - mgz \quad (14)$$

Les components lineals y angulars no depenen unes de les altres, i per tant es poden estudiar per separat obtenint dos equacions: una per les forces lineals y un altre per als moments. Això vol dir que la força que exerceixen els actuadors no depenen de les velocitats angulars que es tinguéssin, i tampoc es tindran acceleracions angulars diferents segons l'altura a la que es trobi el quadcopter: l'objecte girarà de la mateixa manera sigui quina sigui la seva posició en l'espai.

Llavors, fent la derivada parcial respecte \dot{q} s'obté

$$F = \frac{d}{dt} \left(\frac{m}{2} (1 \cdot \dot{\xi} + \dot{\xi} \cdot 1) + \frac{1}{2} \frac{\partial}{\partial \dot{\eta}} (\dot{\eta}^T J \dot{\eta}) \right) - \frac{\partial \mathcal{L}}{\partial q} \quad (15)$$

Com que J és una matriu simètrica, es pot dir que $\frac{\partial}{\partial \dot{\eta}} (\dot{\eta}^T J \dot{\eta}) = 2 \frac{\partial}{\partial \dot{\eta}} (\dot{\eta}^T J) \dot{\eta}$.

Demostració: Per probar això es veurà per al cas $\frac{\partial}{\partial x} (x^T A x) = 2 \frac{\partial}{\partial x} (x^T A) x$ amb

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (16)$$

Llavors

$$\frac{\partial}{\partial x} (x^T A x) = \frac{\partial}{\partial x} \left([x_1 x_2 x_3] \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \right) = \quad (17)$$

$$\frac{\partial}{\partial x} (x_1^2 a_{11} + x_1 x_2 a_{21} + x_1 x_3 a_{31} + x_1 x_2 a_{12} + x_2^2 a_{22} + x_2 x_3 a_{32} + x_1 x_3 a_{13} + x_2 x_3 a_{23} + x_3^2 a_{33}) = \quad (18)$$

Com que A és simètrica $a_{12} = a_{21}$, $a_{13} = a_{31}$ i $a_{23} = a_{32}$, i en fer la derivada direccional resulta

$$\frac{\partial}{\partial x} (x^T A x) = \begin{bmatrix} 2x_1 a_{11} + x_2 a_{21} + x_3 a_{31} + x_2 a_{12} + x_3 a_{13} \\ x_1 a_{21} + x_1 a_{12} + 2x_2 a_{22} + x_3 a_{32} + x_3 a_{23} \\ x_1 a_{31} + x_2 a_{32} + x_1 a_{31} + x_2 a_{23} + 2x_3 a_{23} \end{bmatrix} = 2 \cdot \begin{bmatrix} x_1 a_{11} + x_2 a_{12} + x_3 a_{13} \\ x_1 a_{12} + x_2 a_{22} + x_3 a_{23} \\ x_1 a_{13} + x_2 a_{23} + x_3 a_{23} \end{bmatrix} \quad (19)$$

I en avaluar l'altre costat de la igualtat es té el mateix resultat

$$2 \frac{\partial}{\partial x} (x^T A) x = 2 \frac{\partial}{\partial x} \left(\begin{bmatrix} x_1 a_{11} + x_2 a_{12} + x_3 a_{13} \\ x_1 a_{21} + x_2 a_{22} + x_3 a_{23} \\ x_1 a_{31} + x_2 a_{32} + x_3 a_{33} \end{bmatrix}^T \right) x = \quad (20)$$

$$= 2 \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 2 \cdot \begin{bmatrix} x_1 a_{11} + x_2 a_{12} + x_3 a_{13} \\ x_1 a_{12} + x_2 a_{22} + x_3 a_{23} \\ x_1 a_{13} + x_2 a_{23} + x_3 a_{23} \end{bmatrix} \quad (21)$$

■

Com que $2\frac{\partial}{\partial\dot{\eta}}(\dot{\eta}^T J)\dot{\eta} = 2J\dot{\eta}$ es té, aplicant la regla de la cadena en el producte $J\dot{\eta}$:

$$F = \frac{d}{dt}(m\dot{\xi} + J\dot{\eta}) - \frac{\partial\mathcal{L}}{\partial q} = m\ddot{\xi} + J\ddot{\eta} + \dot{J}\dot{\eta} - \left(\frac{1}{2}2\frac{\partial}{\partial\dot{\eta}}(\dot{\eta}^T J)\dot{\eta} - mg \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right) \quad (22)$$

Per arribar a aquest resultat s'ha aplicat la derivada direccional a mgz :

$$D_q(mgz) = D_\xi(mgz) = mg \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (23)$$

Separant les components lineals i angulars en dues equacions:

$$f = m\ddot{\xi} + mg \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = RT_B \quad (24)$$

$$\tau = J\ddot{\eta} + \underbrace{\left(\dot{J} - \frac{\partial}{\partial\dot{\eta}}(\dot{\eta}^T J) \right)\dot{\eta}}_{C(\eta, \dot{\eta})} = J\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} \quad (25)$$

On $C(\eta, \dot{\eta})$ és la matriu de Coriolis. Per obtenir el sistema d'equacions del model s'han d'aïllar les acceleracions, i s'obté:

$$\begin{cases} \ddot{\xi} &= \frac{1}{m}RT_B - g \cdot [0 \ 0 \ 1]^T \\ \ddot{\eta} &= J^{-1}(\tau - C(\eta, \dot{\eta})\dot{\eta}) \end{cases} \quad (26)$$

Per a realitzar el control serà útil representar el sistema 26 en forma d'espai d'estats:

$$\frac{\partial}{\partial t} \begin{bmatrix} \xi \\ \dot{\xi} \\ \eta \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} \dot{\xi} \\ \frac{1}{m}RT_B - g \cdot [0 \ 0 \ 1]^T \\ \dot{\eta} \\ J^{-1}(\tau - C(\eta, \dot{\eta})\dot{\eta}) \end{bmatrix} \quad (27)$$

3.3. Representació del model amb Matlab

4. Disseny del controlador

5. Implementació del control

6. Construcció del Quadcopter

El conjunt de peces que formen aquest aparell estan connectades entre sí segons la funció que realitzen. Esencialment la Raspberry Pi controla els motors segons les seïals que reb de l'IMU i el Receptor. Tot el conjunt és alimentat per una bateria LiPo i s'adapta el voltatge de 11.1V a 5V per mitjà d'un Regulador per tal d'alimentar a la Raspberry. Tots els components estan subjectats a una estructura (Frame) que també pateix les forces y moments.

6.1. Descripció dels components

Es descriu tot seguit cada component i el criteri de selecció que s'ha aplicat.

Raspberry Pi

Abreujat com a RPi, és un petit ordinador integrat en una sola placa (Single-Board Computer o SBC en anglès) del tamany d'una targeta de crèdit, és a dir, amb unes dimensions de 85.6cm x 53.98cm, desenvolupat per la Fundació Raspberry Pi amb l'intenció de promocionar les ciències computacionals a les escoles [2].



Figura 2: Raspberry Pi a utilitzar

S'ha optat per aquesta opció pel seu econòmic preu, la velocitat de processament i baix consum. A més, s'ha volgut ampliar els coneixements d'aquest petit monstre. En particular s'utilitza la segona revisió del model B:

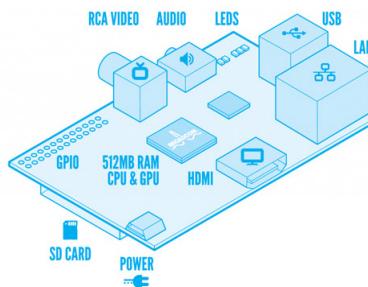


Figura 3: Conectors de la Raspberry Pi

Té un System-on-Chip (SoC) Broadcom BCM2835 amb un ARM1176JZF-S a 700 Mhz, una GPU VideoCore IV i 512 MB de memòria RAM. Disposa de dos ports USB, una sortida mini-jack 3.5mm, sortida d'audio/vídeo HDMI, una sortida RCA i un port RJ45 10/100 d'Ethernet.

L'alimentació es realitza per mitjà d'un mini USB a 5V/700mA, amb un consum de 3.5W. El sistema operatiu és un Raspbian, gravat en una targeta SD de 4GB. Disposa d'un conjunt de pins que permeten comunicació amb perifèrics de baix nivell UART, I2C, SPI i 8 pins de propòsit general (General Purpose Input Output o GPIO).

GY-521 MPU-6050

Es tracta d'una Unitat de Mesura Inercial (IMU en anglès) que integra en un mateix encapsulat de $4x4x0,9mm$ un acceleròmetre i un giròscop, ambdós de 3 eixos. Disposa dun conversor ADC de 16 bits per a cada eix i es comunica mitjançant un protocol de comunicació *I2C*. S'ha optat per utilitzar aquest dispositiu pel seu baix cost i la fàcil comunicació que comporta amb la RPi.



Figura 4: IMU MPU-6050

Com a característiques dels sensors: el giròscop té un rang de $\pm 250, \pm 500, \pm 1000, \pm 2000$ graus/-segon, i l'acceleròmetre de $\pm 2g, \pm 4g, \pm 8g, 16g$. La tensió d'alimentació és del rang de $2,375V - 3,46V$ i cap la possibilitat d'utilitzar un mòdul *DMP* (Digital Motion Processor), però s'ha decidit implementar un filtre de Kalman per llegint les dades en cru (raw) de la cua *FIFO* del sensor.

Emissor-Receptor

Amb aquest parell de components es transmet la consigna generada des del transmissor cap al receptor mitjançant ones de radio. El model que s'utilitza és el *Turnigy5X5ChMini*, per què és un model fàcil d'utilitzar i econòmic. Les especificacions tècniques més rellevants són:



Figura 5: Emissora i Receptor

El transmissor té unes dimensions de $156x152x50mm$, un pes de 265g i s'alimenta a 6V (4 bateries AA). El receptor té unes mides de $33,5x20,5x13mm$ i s'alimenta a 4.8-6V. Disposa de 5 canals de radio, amb transmissió segura a 2.4GHz amb el mètode FHSS. Pot configurarse per traballar amb dos modes (mode1-mode2).

La senyal que es reb per cada canal en el receptor és de *PWM* de $50Hz$ amb un Duty que veria de $1ms$ a $2ms$.

Bateria LiPo

Per a alimentar a tot el conjunt s'utilitza una LiPo *Turnigy2200mAh3S1P25C*. Per tant, és capaç d'entregar 2.2A durant una hora, i com que la capacitat és de 25C, la descàrrega pot ser de $2,2 * 25 = 55A$ amb un pic de descàrrega de 35C, és a dir, amb un pic de $2,2 * 35 = 77A$ durant 10 segons. Aquesta bateria està formada per tres cel·les que proporcionen un voltatge total d'uns 11,1V:



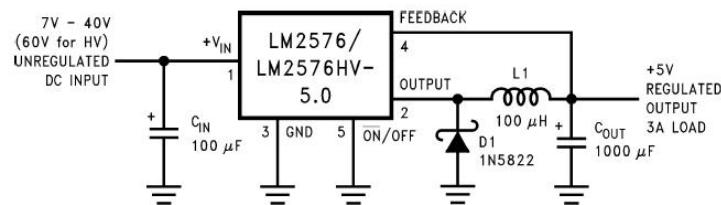
El conector de càrrega és el *JST - XH* i el de descàrrega el *XT60*. Per a tenir en compte amb per al quadcopter és interessant saber que pesa 188g i respecte a l'estructura (frame) que les seves dimensions són de 105x33x24mm.

Regulador Step-Down

Per adaptar la tensió de 11.1V de la bateria LiPo a 5V per alimentar la Raspberry Pi, és necessari un regulador. En aquest cas es té l'Interruptor-Mode Màxim BEC LM2576S, que proporciona fins a 3A de corrent



L'esquema del component [6]:



Motors Brushless

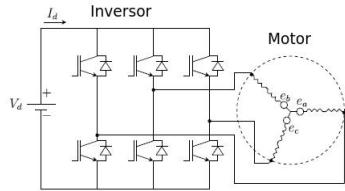
Els actuadors que s'utilitzen són motors brushless (BL-DC). En particular el Turnigy 2213 20turn 1050kv 19A Outrunner. Les seves especificacions més rellevants són:

- Kv: 1050rpm/v
- Corrent de treball: 6A - 16A
- Corrent de pic: 19A
- Pes: 56g
- Dimensions: 27.6 x 32mm
- Mida de l'eix: 3.175mm

La relació entre la senyal de PWM i la força, així com el consum es troben a l'Annex

Variador ESC

Per a controlar un motor brushless l'etapa de potència a utilitzar un ESC (Electronic Speed Controller) que és un variador que consta d'un inversor trifàsic:



Turnigy AE-20A Brushless ESC Specification: Output: Continuous 20A, burst 25A up to 10 seconds. Input Voltage: 2-4 cells lithium battery or 5-12 cells NiMH battery. BEC: Linear 2A @ 5V Control Signal Transmission: Optically coupled system. Max Speed: 2 Pole: 210,000rpm 6 Pole: 70,000rpm 12 Pole: 35,000rpm Size: 50mm (L) * 26mm (W) * 12mm (H). Weight: 19g.

Features: High performance microprocessor brings out the best compatibility with all kinds of motors and the highest driving efficiency. Wide-open heatsink design to get the best heat dissipation effect. Improved Normal, Soft, Very-Soft start modes, compatible with aircraft and helicopter. Smooth, linear, quick and precise throttle response. Multiple protection features: Low-voltage cut-off protection / Over-heat protection / Throttle signal loss protection Programable via transmitter Programming features: Brake setting (we recommend using brake for only folding props applications) Battery type(Li-xx or Ni-xx) Low voltage cutoff setting Factory default setup restore Timing settings (to enhance ESC efficiency and smoothness) Soft acceleration start ups (for delicate gearbox applications) Low voltage cutoff type (power reduction or immediate shutdown)

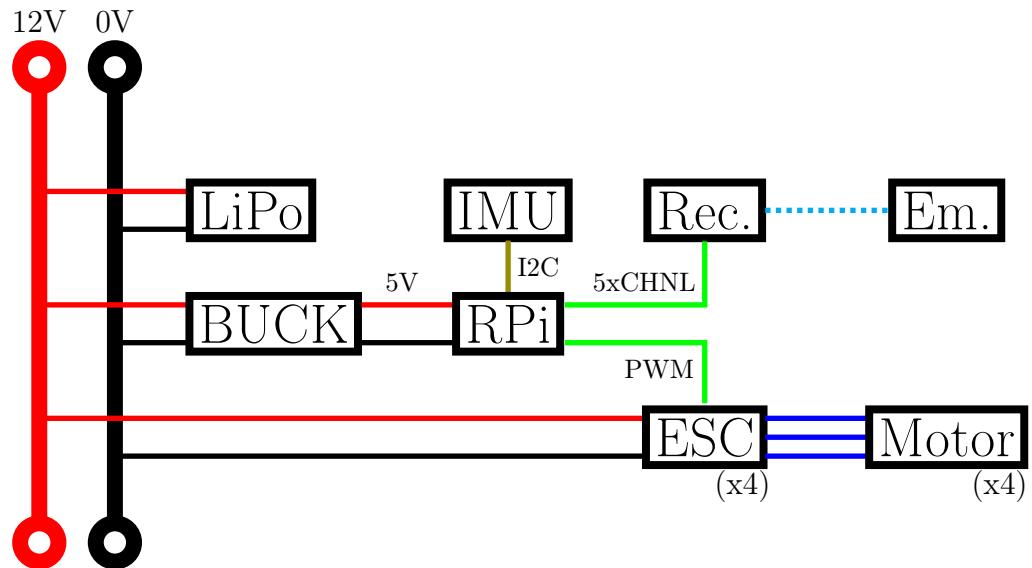
Factory default settings: Brake: off Battery type: Li-xx (Li-ion or Li-Po) Low voltage cutoff threshold: Soft cut-off (2.6V) Timing setup: Low Soft Acceleration Start Up: Normal Low voltage cutoff type: Medium

Frame

Peso 190g

Hèlix

6.2. Muntatge



7. Anàlisi econòmic

ANNEXES

Annex 1: Obtenció vector de forces

Per a treballar amb la dinàmica del sistema s'obté el vector de forces a partir del Lagrangià:

```

clear all
clc

%%% Para usar la funcion Laplace en el archivo Laplace.m se deben declarar
%%% las variables de esta manera:

syms x dx ddx y dy ddy z dz ddz Phi dPhi ddPhi Theta dTheta ddTheta Psi dPsi ddPsi
syms f1 f2 f3 f4 m1 m2 m3 m4
% syms w1 w2 w3 w4
syms Ixx Iyy Izz
syms Ax Ay Az           % No se tienen en cuenta coeficientes de friccion
syms k m g l b

l=0.165                  % longitud de los brazos
Ixx=0.004                 % Momento de inercia del eje x
Iyy=0.004                 % Momento de Inercia del eje y
Izz=0.008                 % Momento de Inercia del eje z
m=0.85                   % Peso del Quadcopter
g=9.81                    % gravedad

%%% El vector de variables que se usaran

v=[x dx ddx y dy ddy z dz ddz Phi dPhi ddPhi Theta dTheta ddTheta Psi dPsi ddPsi]

Xi=[x; y; z]
dXi=[dx; dy; dz]

Eta=[Phi; Theta; Psi]
dEta=[dPhi; dTheta; dPsi]

Q=[Xi; Eta]
dQ=[dXi; dEta]

II=[[Ixx 0 0];
    [0 Iyy 0];
    [0 0 Izz]]

A=[[Ax 0 0];
   [0 Ay 0];
   [0 0 Az]]

R=[[cos(Psi)*cos(Theta) cos(Psi)*sin(Theta)*sin(Phi)-sin(Psi)*cos(Phi) cos(Psi)*sin(Theta)*cos(Phi);
   [sin(Psi)*cos(Theta) sin(Psi)*sin(Theta)*sin(Phi)+cos(Psi)*cos(Phi) sin(Psi)*sin(Theta)*cos(Phi)];
   [-sin(Theta) cos(Theta)*sin(Phi) cos(Theta)*cos(Phi)]]]

Weta=[[1 0 -sin(Theta)];
      [0 cos(Phi) cos(Theta)*sin(Phi)];
      [0 -sin(Phi) cos(Theta)*cos(Phi)]]]

J=transpose(Weta)*II*Weta

```

```

L=m/2*transpose(dXi)*dXi+(1/2)*transpose(dEta)*transpose(Weta)*II*Weta*dEta-m*g*z

%%% Aqui se obtiene con el Lagrangiano el vector de fuerzas F segun
%%% F=(d/dt)(dp(L)/dp(dq))-d(L)/dq

F=Lagrange(L,v)

%%% Derivando a mano se obtiene que TauB=J*ddEta+CdEta
%%% Y como TauB=[F(4); F(5); F(6)], se tiene que el termino de Coriolis es:

CdEta=[F(4); F(5); F(6)]-J*[ddPhi; ddTheta; ddPsi]

%%% Hasta se tiene el producto Coriolis(Eta,dEta)*dEta. Pero como matlab
%%% no elimina las aceleraciones se hace a mano, y queda:

CdEta = [(Iyy*dTheta^2*sin(2*Phi))/2 - (Izz*dTheta^2*sin(2*Phi))/2 - Ixx*dPsi*dTheta*cos(Theta) -
           dPhi*(Ixx*dPsi*cos(Theta) - 2*Iyy*dTheta*cos(Phi)*sin(Phi) + 2*Izz*dTheta*cos(Phi)*sin(Phi) -
           -dPhi*(Ixx*dTheta*cos(Theta) - Iyy*dTheta*cos(Phi)^2*cos(Theta) + Izz*dTheta*cos(Phi)^2*cos(Theta))

T=f1+f2+f3+f4                                % Fuerza total (Total Thrust)
TB=[0; 0; T]                                    % Thrust en la referencia del Quadctopter
TauB=[1*(f4-f2);
      1*(f3-f1);
      % (m1+m2+m3+m4)]
      b*(f1-f2+f3-f4)]

```

Annex 2:Preparació de la Raspberry Pi

Per a posar a punt la RPi s'ha d'instal·lar Raspbian i configurar-lo per tal que es pugui comunicar amb la placa MPU-6050.

Instalació Raspbian

Havent introduït la targeta SD en un lector adequad, es detecta en una terminal mitjançant l'ordre `df -h`. Suposant que hagués estat gravada anteriorment:

```
/dev/sdb1      56M   19M   38M  33% /media/boot  
/dev/sdb2    1.8G  1.4G  266M  85% /media/9c7e2035-df9b-490b-977b-d60f2170889d
```

S'han de desmontar les dos particions, tant la `/dev/sdb1` com la `/dev/sdb2`:

```
umount /dev/sdb1  
umount /dev/sdb2
```

Havent descarregat la imatge a instalar per exemple a l'Escriptori, es procedeix amb:

```
sudo dd bs=4M if=~/Escritorio/2013-07-26-wheezy-raspbian.img of=/dev/sdb
```

Passats uns minuts ja es té la SD grabada amb el sistema operatiu Raspbian.

Configuració de la Raspberry

Per tal de poder comunicar la RPi amb la IMU MPU-6050 per I2C és necessari configurar el sistema. Primer és necessari instal·lar els drivers més rellevants. De l'arxiu

```
sudo nano /etc/modules
```

s'han d'afegir les següents dues línies al final de l'arxiu:

i2c-bcm2708
i2c-dev

En l'arxiu blacklist:

```
sudo vi /etc/modprobe.d/raspi-blacklist.conf
```

les següents dues línies han de començar amb un signe # (de comentari):

```
#blacklist spi-bcm2708  
#blacklist i2c_bcm2708
```

Per provar de connectar el sensor, és necessari seguir primer el connexióat:

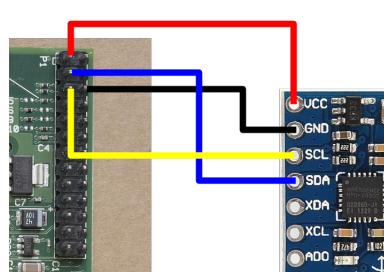


Figura 6: Connexió per I2C entre MPU-6050 i RPi

És a dir:

- Pin1-3.3V es connecta VCC.
- Pin3-SDA es connecta a SDA
- Pin5-SCL es connecta a SCL
- Pin6-Ground es connecta a GND

S'ha d'instal·lar el paquet *i2c-tools*:

```
sudo apt-get install i2c-tools
```

Per a veure el sensor s'escriu:

```
sudo i2cdetect -y 1
```

El resultat és

```
pi@raspberrypi ~ $ sudo i2cdetect -y 1
      0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  -- - - - - - - - - - - - - - - - - - - - - - -
10:  - - - - - - - - - - - - - - - - - - - - - -
20:  - - - - - - - - - - - - - - - - - - - - - -
30:  - - - - - - - - - - - - - - - - - - - - - -
40:  - - - - - - - - - - - - - - - - - - - - - -
50:  - - - - - - - - - - - - - - - - - - - - - -
60:  - - - - - - - - - - - - - - - - - - - - 68 - -
70:  - - - - - - - - - - - - - - - - - - - - - -
```

Figura 7: Detecció del sensor MPU-6050

que és el dispositiu que correspon al MPU-6050.

Llibreria wiringPi

S'utilitza aquesta llibreria[7] com a interfície del GPIO (General Porpouse Input Output). Permet, per tant, llegir tant les consignes com els sensors i generar les accions de control. La instal·lació i descripció de la llibreria està explicat a la pàgina web incluïda a la bibliografia.

Annex 3: Filtre de kalman

Annex 4: Caracterització dels motors

Per tal de poder controlar la planta és necessari saber com responen els actuadors per tal de realitzar un control precís. D'aquests és necessari saber quina força i moment proporcionen, així com el consum d'energia donat un PWM. Per a obtenir una corva de la força respecte del PWM es pesa amb una balança l'acció del motor. La rudimentària bancada que es té per a tal objectiu és d'un colze articulat a un eix de tal manera que la distància de l'eix del motor al fulcre i d'aquest a la balança sigui la mateixa (en aquest cas de 12 cm):

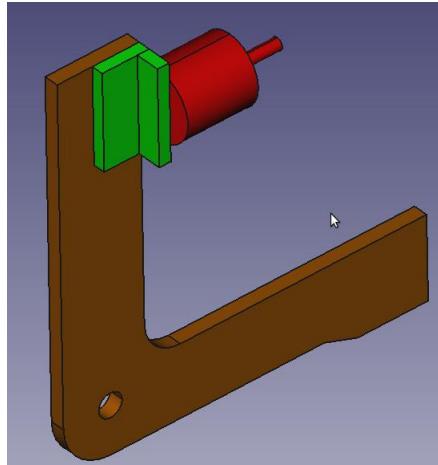


Figura 8: Bancada per a l'estudi del motor

Les parts estan fetes de fusta (part marró) i plàstic ABS (part verda). Tota l'estructura es subjecta a unes fustes:



Figura 9: Montatge per a l'estudi del motor

Es transmet al ESC a quina velocitat s'ha de fer funcionar al motor des de la Raspberry Pi per mitjà d'una senyal PWM. El programa que s'executa per a fer les proves és:

```
#include <stdio.h>
#include <termios.h>
#include <wiringPi.h>
#include <unistd.h>

#define PIN 2 // pin 13

int main(void){
```

```

int i,p;
wiringPiSetup();

pinMode(PIN,OUTPUT);
digitalWrite(PIN,LOW);

for(i=60;i<=100;i+=5){
    for(p=0;p<250;p++){
        digitalWrite(PIN,HIGH);
        delayMicroseconds(950+i*10);
        digitalWrite(PIN,LOW);
        delayMicroseconds(19050-i*10);

        printf("PWM(i): %d \t h:%d \n",i,p);
    }
}
return 0;
}

```

S'obté una mesura de força i d'intensitat per cada increment del 0.25 % del PWM enviat al Variador. Sobre:

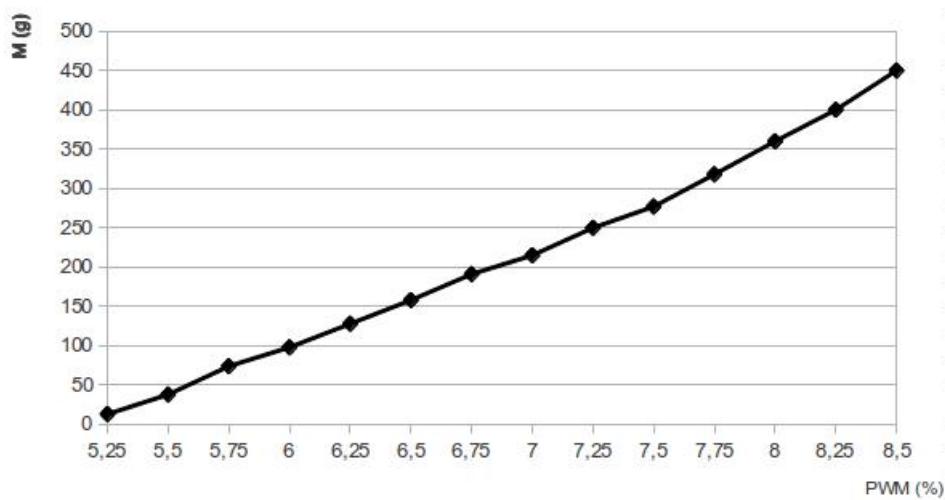


Figura 10: Relació entre la força (en grams) i el PWM

En l'eix de les abscises el PWM(%) és entre el 5% i el 10% de la senyal enviada, que correspon al mínim i màxim que el motor. Com que per a un PWM elevat la força era massa gran i no es preveu que es farà funcionar el sistema en semblants condicions de treball, no s'estudia.

Es veu una relació proporcional del PWM amb la força generada. Es considera que una regressió lineal serà suficient. S'obté que la força $f(PWM)$ del motor serà:

$$f = 32,5758 \cdot PWM(\%) - 32,1758$$

amb un coeficient de determinació en la regressió és de $R^2 = 0,9929$. Aquest resultat facilita el problema en ser una relació prou senzilla.

Per a l'intensitat es té una relació tan amigable com en el cas anterior, però com que no serà cas d'estudi, només es veurà una aproximació del que pot arribar a consumir cada motor per a assegurar una autonomia de vol no irrisòria. Es té que

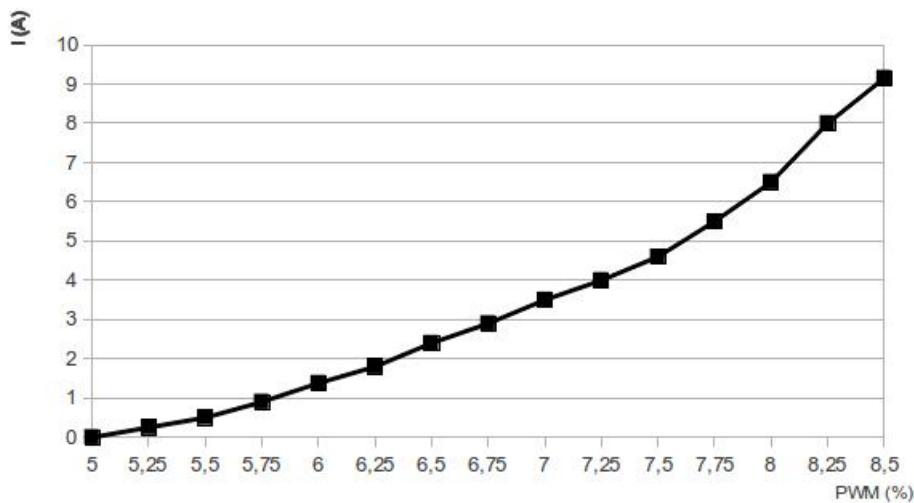


Figura 11: Relació entre la força (en grams) i el PWM

Com que en el punt d'equilibri cada motor suporta una cuarta part del pes, el PWM hauria de correspondre a una força de $m_{Quadcopter} \cdot g/4 \approx 0,225\text{kg}$. El consum corresponent és de 3,5A aproximadament. Entre els quatre motors es consumiràn $3,5 \cdot 4 = 14\text{A}$. Com que la bateria es de 2200mAh es suposa una autonomia de 7 minuts aproximadament.

Referencias

- [1] Teppo Luukkonen (August 22,2011). *Modelling and control of quadcopter*
- [2] Wikipedia de la Raspberry Pi: http://en.wikipedia.org/wiki/Raspberry_Pi
- [3] Accel·leròmetre i Giròscop MPU-6050 per a Arduino: <http://playground.arduino.cc/Main/MPU-6050#.UzhsvCK9jb4>
- [4] MPU-6050. Especificació del producte: <http://www.invensense.com/mems/gyro/documents/PS-MPU-6000A-00v3.4.pdf>
- [5] MPU-6050. Mapa de Registres i descripcions: <http://www.invensense.com/mems/gyro/documents/RM-MPU-6000A-00v4.2.pdf>
- [6] LM2576S Datasheet: <http://www.ti.com/lit/ds/symlink/lm2576.pdf>
- [7] Llibreria wiringPi: <http://wiringpi.com/>