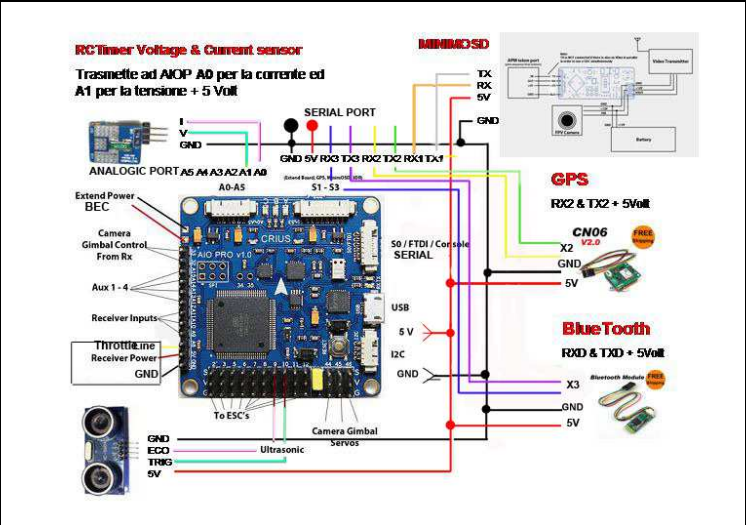
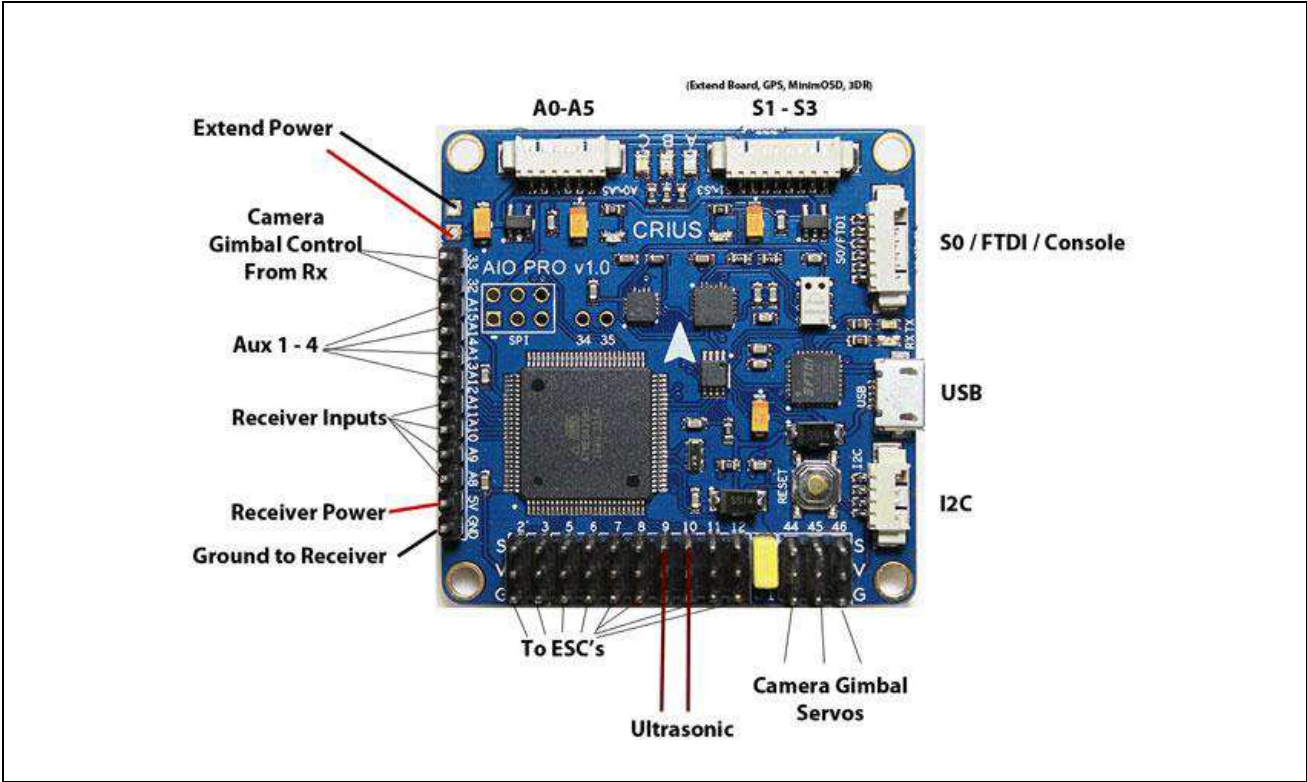


# BOARD: Crius AIOPro



- Extend Power 5V:** A) from ESC to servo port (yellow Jumper ON) B) from BEC to input Power (yellow Jumper OFF)
- Receiver Connections**
- GND – To Receiver Ground
- 5v – To power Receiver 5V
- A8 / THROTTLE – To Receiver (also used for PPMSUM / CPPM Input)
- A9 / ROLL – To Receiver
- A10 / PITCH – To Receiver
- A11 / YAW – To Receiver
- A12 / AUX1 – Normally channel 6 Receiver
- A13 / AUX2 – Normally channel 7 Receiver
- A14 / AUX3
- A15 / AUX4
- CAM PITCH - Input from Receiver
- CAM ROLL - Input from Receiver / (Pin 32 also used for optional Buzzer)

**Motor Connections**

- 2 – Motor Connection
- 3 – Motor Connection
- 4 – Motor Connection
- 5 – Motor Connection
- 6 – Motor Connection
- 7 – Motor Connection
- 8 – Motor Connection
- 9 – Echo for Ultrasonic
- 10 – Trigger(segnaie) for Ultrasonic
- 11 – Motor Connection
- 12 – Motor Connection

**Motor connection** - D2/D3/D5/D6/D7/D8/D11/D12 (See Appendix for Motor Layouts)

- 46 / TRIGGER (segnale) (Camera Gimbal Servo)
- 45 / ROLL (Camera Gimbal Servo)
- 44 / PITCH (Camera Gimbal Servo)

**I2C port** (Scl, Sda, 5v, Gnd) - For I2C Sensors and I2C GPS (Optional)

**S0/FTDI** (known as console in MegaPirateNG Code

- (Gnd, gnd, 5v, rx0, Tx0, Dtr) - can be connected a Bluetooth module
- (This is shared with USB port and cannot be used at the same time)

**S1-S3**

- (Tx1, Rx1, Tx2, Rx2 Tx3, Rx3, Vcc, Gnd)
- RX1/TX1 used for Remzibi's OSD board (Not tested), (FrySky Telemetry interface coming in future) TX2/RX2 used for GPS, you can connect any standard NMEA GPS receiver
- TX3/RX3 used for telemetry modules ,like Xbee, 3DRadio or MinimOSD

**A0-A5**

- A0 and A1 used for Attopilot

Il segnale **CTS** serve per inviare un reset alla mcu tramite un condensatore in serie. É necessario per avviare il boot loader che trasferisce nella flash il codice.hex .

**I2C port=TWI devices** Il bus I2C di trasmissione dati con protocolli seriali, basandosi su due fili, non permette la comunicazione contemporanea tra Master e Slave. Lo scambio dati deve essere gestito dal Master tramite gli indirizzi (univoci) degli slave.

**DOCUMENTAZIONE COMPLETA**

<http://www.baronerosso.it/forum/modelli-multirotori/272972-megapirateng-guide-e-documentazione.html>

<http://code.google.com/p/megapirateng/downloads/list>

<https://code.google.com/p/arducopter/wiki/ArduCopter2>

## PicoBlade™ Molex

**Molex's 1.25mm (.049") pitch PicoBlade™** system is designed for high-density harness applications. PicoBlade provides the same 1.0 amperes of current as similar 2.00mm (.079") pitch systems, but in a more compact design.

PicoBlade™ is offered in wire-to-board and wire-to-wire options.



## SETTAGGI DELLA RADIO

Canale 1-2-3-4 **Comandi**

Canale 5 **Stable-Loiter-Auto**

Canale 6 Setting **su potenziometro (normalmente attivato solo in tuning)**

Canale 7 **RTL oppure Waypoint, ecc.**

Canale 8 ....

### Alternative migliori

Al canale 5 si associa lo switch "3POS" con weight 62 ed offset -10, al quale poi si aggiunge tramite funzione ADD lo switch a 2 pos (es. AIL) con weight a 20 ed offset 0.

In questo modo si ottengono i sei diversi livelli di PWM (<1230; 1231-1360; 1361-1490; 1491-1620; 1621-1749; >1750) da associare ad altrettanti modi di volo che si preferiscono.

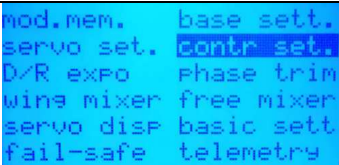
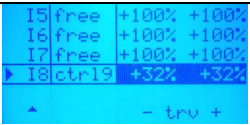
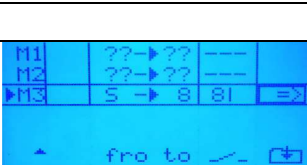
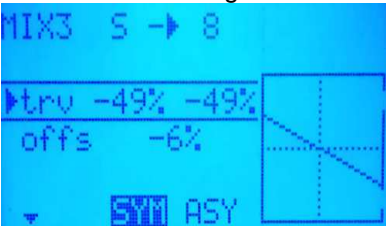
Tramite Megaplanner poi l'assegnazione è molto semplice perché indica in tempo reale il valore PWM ricevuto dal controller e la scelta del modo associato.

A questo punto l'unica accortezza è scegliere bene la sequenza dei modi di volo, ovvero rendere adiacenti dei modi che siano simili fra loro, per evitare problemi nel passaggio fra uno e l'altro. Vedere [http://code.google.com/p/arducopter/wiki/AC2\\_ModeSwitch](http://code.google.com/p/arducopter/wiki/AC2_ModeSwitch)

Tutorials per settaggio interruttori della radio:

<http://code.google.com/p/ardupilot-mega/wiki/Sixmodes>

### GRAUPNER setting interruttori 3+2 pos

	<p>Activate the 3-pos switch (by direct or by mixing mode) on radio channel 5 Activate the 2-pos switch on radio channel 6</p>
<p>Then adjust the following values (+32% and +32%)</p>	
<p>Now back to the main menu, choose "Free mixer" and set the following values First 6 =&gt; 5, and then go to the arrow on the right to set the travel and offset.</p>	
<p>Set travel and offset according to the following screen</p> 	<p>To have a better setting connect MP, select Configuration, chose Modes and verify the right setting moving the switches the right Mode selection. Moving trv SYM values it's possible to change settings.</p>

## INSTALLAZIONE SW

To install MegaPirateNG on your Crius AIO we first of all need to download two programs and their associated files and libraries.

1. MegaPirateNG - Latest MegaPirate can be found here <http://code.google.com/p/megapirateng/downloads/list>

2. Arduino - Arduino can be found here <http://arduino.cc/en/Main/Software>

## Arduino

La versione di Arduino è almeno la 1.0.3.

Copiare le librerie di MegaPirate in Arduino

Per le librerie meglio aggiungerle, se le si sostituisce il problema sorge nel caso si voglia compilare uno sketch che necessita di specifiche librerie che prima c'erano e poi non più

Now we can get started...

*Arduino is used to configure and compile the MegaPirateNG Code and upload it to your Crius AIO.*

- Create a directory on your PC called Crius (on your desktop will be fine)
- Inside the Crius directory create two more directories, Arduino and MegaPirate.
- Unzip the MegaPirateNG files into the MegaPirate Directory you created
- Unzip the Arduino files into the Arduino Directory you created.
- Now open the MegaPirate Folder and copy the **Libraries** folder to the Arduino Folder (you will be prompted and asked if you wanted to **merge the two directories** and overwrite any files etc).
- Now open up the Arduino folder and run Arduino.

Under tools check that its is configured for the correct board, for the Crius is should be **Arduino Mega 2560 or Mega ADK**

Now select the serial port the Crius AIO, this is normally auto detected. This is done in >Tools>Serial Port in the menu.

Ok we now have Arduino setup so the next step is to configure MegaPirate

Use File > Open in Arduino and navigate to the Megapirate folder your created, open the sub folder called **ArduCopter**. Inside this folder there is a file called **ArduCopter.pde**. open this file.

You will notice that there are tab's along the top the only one that we have to make changes to is the **APM\_Config.h** and as you can see this is the second tab along just click on this to edit it.

### Parametri da modificare

```
// Select your sensor board
#define PIRATES_SENSOR_BOARD PIRATES_CRIUS_AIO_PRO_V1
/*
    PIRATES_ALLINONE
    PIRATES_FFIMU
    PIRATES_FREEIMU
    PIRATES_BLACKVORTEX
    PIRATES_FREEIMU_4 // New FreeIMU
0.4.1 with MPU6000, MS5611 and 5883L
    PIRATES_DROTEK_10DOF_MPU // MPU6000, MS5611 and 5883L
    PIRATES_CRIUS_AIO_PRO_V1 // Crius AllInOne Pro v1
*/

// RC configuration
// Uncomment if you uses PPM Sum signal from receiver
// #define SERIAL_PPM ENABLED

#define TX_CHANNEL_SET TX_mwi
/*
    TX_set1 //Graupner/Spektrum
    TX_standard //standard PPM layout
Robbe/Hitec/Sanwa ROLL,PITCH,THROTTLE,YAW,MODE,AUX2,CAMPITCH,CAMROLL
    TX_set2 //some
Hitec/Sanwa/others PITCH,ROLL,THROTTLE,YAW, AUX1,AUX2,CAMPITCH,CAMROLL
    TX_mwi //MultiWii layout
*/

// Select your baro sensor
#define CONFIG_BARO AP_BARO_MS5611_I2C
/*
    AP_BARO_BMP085_PIRATES
    AP_BARO_MS5611_I2C
```

```

*/

// Warning: COPTER_LEDS is not compatible with LED_SEQUENCER, so enable only one
option
#define COPTER_LEDS_ENABLED // New feature coming from ArduCopter
// #define LED_SEQUENCER_ENABLED // Old Oleg's LED Sequencer, see leds.pde for
more info

#define MAX_SONAR_RANGE 400

#define OSD_PROTOCOL OSD_PROTOCOL_NONE
/*
    OSD_PROTOCOL_NONE
    OSD_PROTOCOL_SYBERIAN
    OSD_PROTOCOL_REMZIBI // Read more at:
http://www.rcgroups.com/forums/showthread.php?t=921467
*/

// For BlackVortex, just set PIRATES_SENSOR_BOARD as PIRATES_BLACKVORTEX, GPS
will be selected automatically
#define GPS_PROTOCOL GPS_PROTOCOL_AUTO
/*
    GPS_PROTOCOL_NONE        without GPS
    GPS_PROTOCOL_NMEA
    GPS_PROTOCOL_SIRF
    GPS_PROTOCOL_UBLOX       <<< Select this for UBLOX LEA-6 (CRIUS GPS boards
and others)
    GPS_PROTOCOL_MTK16
    GPS_PROTOCOL_BLACKVORTEX
    GPS_PROTOCOL_AUTO        auto select GPS
*/

#define SERIAL0_BAUD          115200 // Console port
#define SERIAL2_BAUD          38400  // GPS port
#define SERIAL3_BAUD          57600  // Telemetry (MAVLINK) port

// New in 2.0.43, but unused in MegairateNG
// MPNG: Piezo uses AN5 pin in ArduCopter, we uses AN5 for CLI switch
#define PIEZO_DISABLED
#define PIEZO_LOW_VOLTAGE    DISABLED
#define PIEZO_ARMING         DISABLED

#define FRAME_CONFIG QUAD_FRAME
/*
    QUAD_FRAME
    TRI_FRAME
    HEXA_FRAME
    Y6_FRAME
    OCTA_FRAME
    OCTA_QUAD_FRAME
    HELI_FRAME
*/

#define FRAME_ORIENTATION X_FRAME
/*
    PLUS_FRAME
    X_FRAME
    V_FRAME

```

Per il **GPS V1** , diversamente dal V2, occorre accenderlo e spegnerlo due volte. Su MPNG il led che si accende quando fa il fix è il **C**.

## A (Red)

Blink = Disarmed, motors will not spin with throttle up

Solid = **Armed**, motors are live;

## B (Yellow)

Flash during calibration, otherwise not used

## C (Blue) fix gps

OFF = No GPS connected or Serial received

Blink = Waiting for Lock,

Solid = **Lock**,

RX,TX **will flash during serial communications**, such as loading code and telemetry

All LEDs blinking in sequence means you are in the one-time ESC calibration routine.

Reset the board to exit calibration by removing power.

## Installation log: setup finale e primo volo

### ISTRUZIONI COMPLETE

<http://code.google.com/p/arducopter/wiki/ArduCopter2Quick>

## Le mie porte:

Con cavo USB =COM 9

Con BT = COM 12

1 - Ricaricare MPNG senza GPS configurato e collegato

2 - Fare **calibrazione della radio** e accelerometri su Mission Planner.

3 - Collegare gli ESC come da schemino e fare la calibrazione degli ESC come da istruzioni (senza eliche):

[AC2 ESC - arducopter - Arduino-based autopilot for multicopter craft, from quadcopters to traditional helis - Google Project Hosting](#)

## ESC calibration (AKA: setting the end points)

One of the things that's trickiest with quadcopters is calibrating all four ESCs so that they respond the same to APM and your RC system. But it's essential. Your copter won't fly correctly if you don't do this!

Note 1: the Auto ESC calibration mode simply causes the APM to allow a full passthrough to the ESCs. If you power up the APM while in this mode you'll send the same PWM signal to all the ESCs. That's all it does. Many ESCs use full throttle at startup to enter programming mode, full throttle position is then saved as the upper end point and when you pull the throttle down to zero, that position is saved as the lower end point.

Note 2: The excellent DJI Opto ESCs (30 amp and 18 amp) do not require and will not respond to the tuning Please also note the following:

that some brands of ESCs do not allow calibration of the input throttle range and will not arm unless you instead adjust your radio's throttle end-points so that the minimum throttle is around 1000 pwm.

For most other brands (including the 3DR ESCs) however you can calibrate the input throttle range using the Automatic method.

For programmable transmitters, set the 4 stick controlled channels endpoints to +100% / -100% and center their trims.

### Automatic ESC calibration (all at once)

This method works once you have all the ESCs connected to the power distribution board and have connected all your RC cables as instructed earlier in the manual and have otherwise set up your multicopter.

Safety First! - Remove the props!

**Disconnect USB and battery.** The entire system must *not have power*

Turn your radio on and put the radio throttle **stick at maximum**.

**Connect the Lipo** to power the APM, radio receiver and ESCs. You will probably hear **one short musical tone** from the ESCs. (but NOT two beeps after the short musical tone - if you get two beeps after the short musical tone and the motors are live it's probable that your throttle needs to be reversed)

When the APM boots, **the ABC LED's will cycle slowly** and continuously

Leaving the throttle stick at full, **disconnect the Lipo and reconnect it again to reboot the APM** and power the system. The APM will now pass the radio signal directly to the ESCs, triggering calibration. **The ABC LED's will cycle quickly - You will probably hear the short musical note and then 2 beeps** (it depends on what brand of ESC you have; the 3DR ones will do as described). **After the two beeps there is a pause, during this pause drop the throttle to full down.** You should hear confirmation/arming beeps.

Do not unplug, **move the throttle up a little to confirm all ESCs are armed** and the motors are working in sync.

**Unplug the battery again and then reconnect.** Now arm the APM with right rudder/low throttle for 4 secs, release rudder and then all your motors should spin if given a little throttle.

Your ESC's are now calibrated. No further action is required.

Note: If after performing an Automatic ESC calibration the ESCs still beep quickly and continuously on power-up, make sure you have done the radio calibration in the Mission Planner! If that has been done you will need to do a Manual ESC calibration sequence as described below.

## Testing

Once you have calibrated your ESCs, you can test them by plugging in your LiPo, leave the props off for the moment.

Ensure your transmitters mode select switch is set to the "Stabilize Mode" position.

Once the APM boots up, arm your ESCs by pushing the left (yaw - throttle) stick all the way down and to the right for at least four seconds. (The A red LED will stop flashing and be on solid).

Note: If the ESCs are not beeping or are beeping slowly but the APM still wont arm (red LED solid on), try setting the transmitter trim 7 or 8 clicks down from center and then push the left stick down and to the right again. An offset below zero is sometimes needed to permit arming.

Once Armed you can then give a small amount of throttle. All motors should spin about same speed and they should start at the same time. If the motors do not all start at the same time and spin at the same speed, the ESC's are still not properly programmed.

You can also disarm your ESCs and motors by pushing the left (yaw - throttle) stick all the way down and to the left for at least four seconds. (The A red LED will resume flashing).

The APM will disarm the ESCs automatically after 10 secs, however it is very strongly recommended to manually disarm the APM at all times when it is not being flown. The potential for grievous bodily injury is greatly reduced.

**(Red A LED Flashing = Disarmed = Safe)**

**(Red A LED Solid = Armed = NOT Safe)**

***Don't forget to ensure that your transmitters mode select switch is set to the "Stabilize Mode" position when you try to Arm the motors.***

### Notes

If after calibration your motors do NOT spin same speed nor start at the same time, repeat the calibration process. If you tried the auto calibration above and it didn't work or the ESCs do not drive the motors identically, try the manual calibration method described above. That should work almost every time. (Rarely after a full manual calibration you will also need to do an additional final automatic calibration).

Finally, there are a huge number of brands and types of ESCs available and some of them do not adhere to the normal programming conventions (sometimes even though they claim to) and they may simply not work with the APM the way it is now. This is an unfortunately necessary but true disclaimer. DJI Opto ESCs are pretuned and work fine but do not need and do not support calibration.

### Recommended ESC settings as follows:

Brake: OFF

Battery Type: LiPo

CutOff Mode: Soft-Cut (Default)

CutOff Threshold: Medium (Default)

Start Mode: Normal (Default)

Timing: MEDIUM

### AVVERTENZA

*the default setting for Rate Control P is 0.1750. This is optimized for the standard Jdrones/3DR quadcopter with **850 motors and a 10" prop**. If, however, you're using the bigger **880 motors and a 12" prop**, which are much **more powerful**, that **Rate Roll P will be too aggressive and your quad will seem quite unstable**. So in that case, **lower your Rate Roll P to 0.15 or even 0.1**, and you should see much more stable flight.*

1 - Motori ed eliche hanno lo stesso senso di rotazione di MultiWii quindi non è necessario invertire il senso di rotazione dei motori.

2 - Impostare sulla radio solo il modo STABILIZE lasciare tutto il resto in default.

3 - Provare il quad in mano e verificare i comandi e le risposte dei sensori sui motori.

4 - Fare il primo hovering.

Il quad deve star su da solo senza troppi interventi sugli stick. Se non si toccano gli stick vola meglio, se ne va lentamente in giro ma non è stato ancora attivato il GPS.  
In MultiWii ci sono moltissimi filtri da applicare per smorzare le oscillazioni mentre in MPNG basta un buon PID tuning

*Passi successivi:*

- PID tuning
- test dei vari flight mode (Alt Hold, Acro, ecc.)
- collegamento GPS e prova del LOITER

### GPS

Il **GPS Rctimer: U-Blox CN-06 v.2**, (ma anche il v.1), funziona senza problemi.  
*Eeprom per salvataggio perpetuo della configurazione, porta UART (TTL) con protezione EMI, batteria ricaricabile per hotstart, 9600 Baud; Se usato per I2C NavBoard deve essere cambiato a 115200 baud (5Hz update rate).*

Si può alimentare il GPS dal piedino Vcc del connettore I2C 5V  
Semplicemente si collega con la presa S1 - S3 a Tx2 & Rx2 e si imposta nell' APM\_ config.h la velocità:

```
#define SERIAL2_BAUD 115200 // GPS port
```

e il tipo di colloquio:

```
#define GPS_PROTOCOL GPS_PROTOCOL_NMEA
```

Non serve toccare nulla nei firmware a bordo dell'MTK e dell'Atmega 328 che fa il parsing, questo consente di poterlo eventualmente spostare da un modello all'altro senza dover riconfigurare niente.

Si collega il modulo GPS (se necessario bypassando il 328p con un semplice erase flash), si imposta la velocità serial2 a 115200 ed il modulo viene visto correttamente.

Effettuate la Calibrazione sensori e la calibrazione radio la FC si arma e disarma correttamente con il gas spento e timone tutto a cabrare ed a destra per almeno 5 sec.

I motori restano fermi ma dando gas (poco!) si attivano.

*Nel caso del V1 il Gps, funzionando a 3.3V trasmette, dal suo piedino TX un segnale ad onda quadra con un'ampiezza di 3.3V che rientra perfettamente nel livello logico 1 del microprocessore e pertanto funziona perfettamente.*

*L'Atmega invece, funzionando a 5V trasmette dal suo piedino Tx ovviamente un segnale a 5V che il Gps non gradisce ricevere e pertanto a lungo andare si brucia.*

*Per non vederlo morire anticipatamente e **solo per il CN-06 v.1**, si deve mettere la resistenza da 10K sul piedino Rx del Gps per abbassare il livello dei 5V che le invia il microprocessore.*

Nel file APM\_Config.h bisogna modificare le linee come segue:

Codice:

```
#define GPS_PROTOCOL GPS_PROTOCOL_UBLOX
/*
  GPS_PROTOCOL_NONE      without GPS
  GPS_PROTOCOL_NMEA
  GPS_PROTOCOL_SIRF
  GPS_PROTOCOL_UBLOX     <<< Select this for UBLOX LEA-6 (CRIUS GPS boards and
others)
  GPS_PROTOCOL_MTK16
  GPS_PROTOCOL_BLACKVORTEX
  GPS_PROTOCOL_AUTO      auto select GPS
*/
...
#define SERIAL2_BAUD      115200      // GPS port
```

Sul Mission Planner, collegandosi con la USB, basta andare su **CLI e fare test**. Il comando RAWGPS fa vedere le frame che si scambiano gps e board e si può vedere che protocollo usano e più o meno a che velocità



Se si prova a fare il test con la CLI, senza il fix i dati rawgps sono totalmente incomprensibili: una riga continua di quadratini e punti di domanda senza nulla di intellegibile. Sembra che siano dati binari e non stringhe di testo.

se la velocità è giusta si devono vedere le stringhe NMEA

Se il GPS non vuole saperne basta **riportarlo con u-center ai valori iniziali NMEA 9600 bps poi ci pensa MPNG a sistemarlo** come vuole lui.

## Bluetooth

*L'unica cosa che non si può fare con il BT e caricare il firmware.*

*Il BT viene collegato alla porta FTDI avendo ovviamente cura di invertire Tx & Rx ovvero:*

*BT Tx > FTDI Rx*

*BT Rx > FTDI Tx*

*Mentre GND e +5V si possono prendere dalla I2C (se il BT è a +5V)*

*Quindi il **GPS** va collegato alla alla presetta S1 - S3 al Tx2 & Rx2 (**Seriale2**) mentre il BT va collegato alla FTDI sulla presetta S0/FTDI/Console a Rx0 e Tx0 (quindi **Seriale 0**) come per MultiWii ma bisogna cambiare la velocità del MavLink sulla seriale 3*

Attivare BT sul PC e fare il **pairing** con la CRIUS BT (pw=0000) e impostare la periferica sul PC a 115200 baud.

Quando sul PC si inserisce BT si creano DUE porte seriali di ingresso e di uscita dati ad es. COM(XX) e COM(YY) dove XX e YY indicano la numerazione delle due porte.

Su MPNG si farà partire il Mission Planner e si seleziona una delle due porte.

Con la prima (XX) dice che la porta non c'è, con la seconda (YY) parte ma la luce rossa sul modulo CRIUS BT rimane lampeggiante e il Mission Planner va in timeout.

Bisogna nel solito **APM\_config.h** mettere **la giusta velocità per il MavLink** ovvero quella della nostra scheda BT.

```
#define SERIAL0_BAUD 115200 // Console port
#define SERIAL2_BAUD 115200 // GPS port
#define SERIAL3_BAUD 115200 // Telemetry (MAVLINK) port (era 57600)
```

Se si incontrano problemi (certamente con Win7) ad effettuare il collegamento provare ad impostare il controllo di flusso "Xon-Xoff" nelle impostazioni della porta COM (YY).

Per vedere su MissionPlanner con il GPS la pianta di Maps non occorre che il PC sia necessariamente collegato ad internet: basta che dalla sezione "Flight Planner" tenendo premuto ALT con il mouse selezioni l'area della mappa interessata, poi, con il tasto destro del mouse, si seleziona **MapTools** e poi **Prefetch**.

Verrà richiesto fino a che livello di zoom salvare la mappa.

Così facendo si salva la mappa sul disco e sarà disponibile anche offline.

## App per Android

Questa è a pagamento ed è da approfondire. [Ground Control Station for Android | AutoQuad](#)

**Copter-gcs** è invece gratuita

Caricata sullo Smartphone da [copter-gcs - GCS for ardu\\* using Android - Google Project Hosting](#) si vede l'orizzonte artificiale (HUD), i segnali dei canali dalla radio, i PID, ecc.

Se capita che invece del punto dove si trova il quadri come nel Mission Planner compaia una videata nera è un problema di scritte tradotte. Se si lascia tutto in inglese si capisce meglio e non si hanno sovrapposizioni.

## Modalità di volo

<a href="#">Acro</a>	<b>ACRO mode</b> -Rate based control
<a href="#">Stabilize</a>	<b>STABILIZE mode</b> - Good for FPV and general flying
<a href="#">Simple</a>	<b>SIMPLE mode</b> - Good for new pilots. Yaw is calculated out so forward stays forward, no matter what orientation
<a href="#">Alt Hold</a>	<b>ALT_HOLD</b> – Good for testing your PIDs for navigation, only altitude is controlled automatically
<a href="#">Loiter</a>	<b>LOITER</b> - Position hold
<a href="#">RTL</a>	<b>RTL</b> - return to launch at <b>desired altitude</b> or <b>from current altitude</b>
<a href="#">Auto</a>	<b>AUTO</b> - runs the current loaded mission script
<a href="#">Guided</a>	<b>GUIDED</b> - similar to RTL, except the target location can be set by clicking on The Mission Planner Flight Data map: need wireless telemetry between your copter and your laptop
<a href="#">Position</a>	<b>Position mode</b> is the same as Loiter, but with manual throttle control
<a href="#">Circle</a>	<b>Circle</b> - will orbit around its current location stay at it's current altitude and points towards the center of the orbit. Throttle changes the altitude with the same caveats as in Alt Hold mode You <b>cannot land and turn off</b> the motors manually in this mode: <b>Switch to Stabilize Mode to land and disarm the motors</b>
<a href="#">Follow Me!</a>	<b>Follow me</b> need a PC and a Bluetooth GPS module <a href="#">like this</a>
<a href="#">Simple GeoFence</a>	<b>GeoFence</b> is a user-defined box that if breached forces the copter to return. It can be used as a safety measure

*All modes are fully interactive - meaning you can fly the copter normally, to override the autonomy.  
radio failsafe can be configured to RTL  
Sonar and Baro altitude support  
Attopilot Current sensor support*

## COMANDI

Con un interruttore a tre posizioni sul canale 5 si possono provare inizialmente le modalità di volo:

- 1-STABLE
- 2-ALT HOLD
- 3-LOITER.

Se necessario abbassare un poco i PID del P e R RATE della modalità STABLE e lasciare tutto il resto di default. Tutto dovrebbe funzionare bene. Il quad preferisce essere lasciato in pace...effettivamente vola meglio da solo.

L'ALT hold con i PID di default funziona bene. Il quad in assenza di vento tiene la quota nell'ambito di circa 30-50cm....magari con un po' di tuning si puo' migliorare...ma va già bene così.

Il LOITER funziona. Il quad vola da solo e se ne sta tranquillo cercando di mantenersi vicino alla home. Ogni tanto cerca di allontanarsi un po' ma poi ritorna

### Failsafe: come viene gestito

Per il **failsafe c'è l'apposito menu nel mission planner**, bisogna settare poi il failsafe sotto il segnale minimo dell'armamento, impostando il failsafe della ricevente in modo che sia al di sotto di tale segnale, quindi con il mission planner si settano le funzioni desiderate in caso di mancanza del contatto radio.

*C'è una guida dettagliata tra i wiki di arducopter: per i dettagli occorre a leggere i wiki di arducopter.*

## PID

In Arducopter e di conseguenza anche in Megapirate i pid di default sono in funzione di un quadricottero con motori da **850-880kv**, per cui per motori superiori, si deve abbassare sicuramente di un poco la "P".

*Occorre aprire il mission planner, andare in terminale e premere tre volte il tasto enter, quindi digitare setup e successivamente erase, molto importante se non è stato fatto dopo aver caricato per la prima volta il codice (serve per pulire la memoria), quindi dare il comando reset e riavviare la FC, quindi per motori intorno ai 1000KV abbassare il valore del pitch Rate P intorno al valore 0,110 e il relativo P dello stabilize a 4,000, il resto va bene per cominciare di*

default, se dovessero esserci problemi con lo yaw aumentare lievemente il valore della P rate della stessa.

Per i settings ottimali **se si ha una radio con un potenziometro, si può usare tale potenziometro ed assegnarlo al canale 6** per procedere alla taratura dei Pids

**Per configurare il potenziometro sul canale 6**, dopo aver ovviamente settato la radio per utilizzare il potenziometro sul canale 6 ci si collega al mission planner, andare su advanced **pid** e dove vi è settato di default **CH6\_do none**, selezionare la funzione che interessa assegnare, e regolare il range dei due valori scritti sotto per gli estremi del potenziometro che devono essere settati in conseguenza alla massima e minima variazione richiesta per ogni singolo parametro, poi si darà il **write parameters**. E' consigliabile poi dare un refresh e verificare che il valore del parametro che è stato assegnato al potenziometro sia vicino od uguale al valore precedente all'assegnazione al potenziometro e da lì poi fare le dovute regolazioni. Il sistema del PID tuning in-flight con il canale 6 sembra funzionare perfettamente.

**D** è molto sensibile a variazioni piccole dei valori, meglio lasciare tutto di default tranne i valori delle P del rate. Il quadricottero dovrebbe volare comunque in modo accettabile con solo i valori P settati. Se non si riesce a trovare un buon compromesso solo con la variazione delle P del rate, prima di toccare gli altri valori verificare di non aver problemi di vibrazioni (bilanciamento eliche, motori, fissaggio controller etc).

*L'eventuale oscillazione è probabilmente dovuta al RATE\_KP troppo elevato, e con motori da 1270Kv abbassandolo a 0,110 le oscillazioni scompaiono. E' anche utile abbassare il PID P dello Stabilize a 4,100 e sembra andare bene.*

### **Com'è possibile trimmare il livellamento dell'accelerometro in MegaPirate?**

Da quello che sembra in MPNG (al contrario che MultiWii) bisogna come PRIMA cosa trimmare lo STABLE mode...poi tutto il resto. Lo STABLE mode si trimma con i trim della radio.

### **PRIMO TEST (di TermicOne per Return To Landing/Return To Home).**

Impostata quota RTL = 10m (ALT\_HOLD\_RTL : 1000,00) , quota target a home position 2m (APPROACH\_ALT: 200) e una velocità di ritorno a casa di soli 2m/s per evitare guai (WP\_SPEED\_MAX: 200).

*In effetti quando attivi RTL il quad si alza fino alla quota desiderata e inizia a tornare. In questa fase il throttle non ha più effetto a meno che non lo si metta a 0. Alla home position il quad si alza di un metro e poi inizia la discesa fino alla quota target di approach.*

### **RTL**

Si ha un comportamento strano: il quad quando si muove non punta la prua nella direzione di marcia (come MultiWii e Naza) ma si muove mantenendo sempre la prua nella medesima direzione

**In config.h ci sono le define di default.** Tu puoi fare l'**override** delle define di default **aggiungendo in APM\_config.h le define che vuoi tu.** Non ci sono in APM\_config.h tutte le define possibili ma solo le più comuni...le altre, se necessario, occorre aggiungerle.

Nel nostro caso in APM\_config.h non c'è una define per il comportamento dello yaw in RTL e occorre aggiungerla altrimenti viene attivata la define presente in config.h che prevede YAW\_HOLD

A questo problema c'è una soluzione:

C'è una define per impostare l'orientamento: in config.h per il modo RTL è definita come

```
#define RTL_YAW YAW_HOLD
```

**occorre impostare in APM\_Config.h**

## **#define RTL\_YAW YAW\_LOOK\_AT\_HOME**

in modo che il quad guardi verso la home: ora in RTL il quad ruota la prua verso la home.  
Nel file **config.h** ci sono tutte le impostazioni di default nel caso che non ci siano altre define in **APM\_config.h**

Si consiglia di **non modificare i default in config.h ma di inserire sempre** la define in **APM\_config.h**

## **DEFINIZIONE DELLE VARIE MODALITA' di PRUA**

**YAW\_ACRO** - Will not hold an yaw angle, not really recommend for most flyers

**YAW\_HOLD** - Will hold a Yaw angle

**YAW\_AUTO** - Will point towards the next WP or however your mission requires

**YAW\_LOOK\_AT\_HOME** - Always points at home

## **AUTOTRIM**

**I trim della radio non si devono usare.**

**C'è la funzione autotrim**, comando ad armare per 15 secondi finché i led cominciano a lampeggiare e si tiene il mezzo in hovering correggendo con gli stick, il tutto dura una trentina di secondi.

**Meglio l'autotrim ch7**

Utilizza il **ch7** per settare in volo l'autotrim.

E' facile vedere come funziona perchè provandolo al banco, attivando il canale 7 e muovendo gli stick si muove l'orizzonte artificiale nella GUI.

**In volo** basta **attivare il canale 7**, dare aggiustamenti agli stick e disattivarlo. Se ancora non va bene basta ripetere la procedura: semplice e veloce.

**L'AutoTrim è assolutamente da usare!**

La trimmatura risulta perfetta e non occorre ritrimmare più nulla. Basta fare la calibrazione degli acc a terra e quindi quella con il canale 7.

## **WAYPOINS**

Sistemato LOITER, YAW ed RTL ora dobbiamo iniziare a studiare i waypoint

*Quelli sono veramente una chicca, semplici da impostare col mission planner, ci fai abbastanza ciò che vuoi, si può programmare una missione completamente automatica dal decollo all'atterraggio.*

## **AGGIORNAMENTI MPNG**

Riferimento **2.8 R3**.

Visto che MPNG é un porting da arducopter le novità dovrebbero essere queste

<http://diydrones.com/forum/topics/ar...r-2-8-released>

**Arducopter 2.8.1 is now in the mission planner and in the [downloads area](#)! Remember to do your MP configuration again after loading this code, since it erases your EEPROM and sets it to the new defaults.**

Note: Issues with APM1 user's level feature not working are now resolved. If you installed 2.8 we highly recommend you **upgrade to 2.8.1** as 430 extra bytes of RAM have been freed up which reduces the chance of memory corruption (although we haven't seen any cases of this on the APM2 at least) .

As per usual PIDs are optimised for the 3DR/jDrones quad with 850 motors and 10" props. If you're using more powerful motors/props **and are seeing bad flight behaviour in stabilize, start by turning down Rate Roll P in 25% steps.**

**Please note that on this release we've moved the Roll and Pitch I terms from the Stabilize controller to the Rate controller. There's some evidence that says this can lead to flips on take-off if you move**

the roll or pitch sticks around as you take-off so we recommend you leave these sticks in the middle until you're in the air.

L'ultima versione di arducopter, **la 2.9 RC2**, oltre ad integrare gli accelerometri per l'alt-hold finalmente atterra perfettamente, spegne i motori e disarma la scheda, tra l'altro è possibile programmare il rateo di discesa.

Attendiamo che Sir Alex faccia in fretta il porting su MegaPirateNG per vedere come va.

## The arducopter tuning guide (da Wiki)

This guide was written at the time of the release of Arducopter 2.8.1 FW.

### The background

Tuning your flight controller is essential for a perfect flight. Even a fully stock frame will fly better if you tune. The default params will be close, but your battery or motors might be performing differently, your air maybe 'thinner' compared to the developer that last published the default params, so you should tune.

What we are trying to achieve is control. We start to accomplish this by modifying the amount of P, I and D in our control loops. All three react to our stick inputs and the errors created by measurements from our sensors, and all three will help us fly better. If we do not have enough of any of the params the copter will not feel like it's under control, it will wander about, not resist wind and be hard to fly. Too much of any of the params and the copter will seem to have a mind of its own. It will over-react. At best you will see it oscillate, very fast, not so fast or somewhere in between, depending on which parameter has been overclocked. At worst it will simple flip over on take-off or be very unpredictable in flight.

We need to find a balance. You can tune each param up individually, *but one will have an effect on another*. Therefore *we must tune one, then another, then go back and see if we can retune the first one again*, this is the balance.

And this is where we adjust those parameters – Mission planner config page:



### Before you start:

1. Charge all of your batteries, you are going to need them.
2. Check you are using compatible ESC's, motors etc, some combinations cause problems.

3. Balance all of your props (in-situ preferably – adding bits of tape or nail varnish to one side of the prop until the motor and prop don't shake at all under throttle.
4. Do all the other sensible things like check everything over and calibrate your ESC's , radio compass etc.
5. Set your copter up as you will normally be flying it. If you tune it then add or remove weight, or change the CoG, you will have to tune it again. Use your normal flight batteries, a 4s tune will be different to a 3S tune. If you don't want to risk your expensive camera, just substitute its mass with something less valuable, like a bean bag or kids toy.
6. Be prepared to have the chopper in your hand if possible and throttled up (so gloves on, maybe a full face motorbike helmet) this way you be able to tune quickly and accurately. If you simply can't hold it in your hand, or a simple jig.
7. But read this guide anyway it will speed you right up! .

### So let's start tuning

Everything you read below will be demonstrated in the tuning video.

I've found the quickest way to tune the arducopter is to start with rate\_D, (but you simply can't do this unless you can hold your beast in hand or in a jig *as the copter won't fly without any rate\_P or I*) so a more universal way to start, and actually **a more informative way is to start with rate\_P.**

Since this is about learning for everyone, let's do it that way. I'll give a brief low-down on the 'rate\_D' tune (or Dave C tune, as it's become known) ' later.

### STEP ONE - Tuning rate P

1. Put your copter in ACRO mode,
2. Turn rate\_D and rate\_I to ZERO.
3. Set your stab\_P to about ZERO, or if you decide to avoid the jig/hand tune and go for immediate in-flight tune set it to about 3.0 as the code is currently using the stab param in acro mode), a true acro mode would not use the stab P parameter.
4. Set a range of rate\_P you would like to work with. I'd advise 0.050 or lower to 0.225 the first time you do this as, although you are unlikely to end up near either extreme, you will see what happens at, and beyond, the limits. You will be able to reduce this range as you become more familiar with tuning.
5. If you know how to do it, *set a potenziometer on your transmitter as channel 6* and use mission planner to set your limits, if you have telemetry this is a joy, if not you are going to be plugging and unplugging your usb quite a bit, it's not that bad though, and defo worth it.
6. OK, time to go, hold copter in hand above head, or place in jig. Throttle up to about hovering point. Wiggle your roll and pitch sticks. With rate\_P about 0.075 (unless you have a monster powerful chopper) it will react slowly to your sticks, it will feel relaxed, lazy. If you tried to fly this you'd be chasing it about with your sticks, you'd probably crash if it was nose in, if you know what I mean.
7. Slowly turn up rate\_P with your pot, or in MP. Move the copter about with your hand and with the sticks as you do so, raise the throttle a bit, lower it a bit, add in more rate\_P. It will start to feel and look much tighter in the air, throttle right up and flick your stick the copter should move fast, decisively but then stop quickly and smoothly. If it seems lazy you need more rate\_P, if it starts to bounce when you move the sticks or tip the copter about then you have past your limit at this point. Get to a point where you are happy, it feels tight but does not shake.

Save this rate\_P, value, our aim is now to increase it using another parameter – rate\_D so on to step two

### **STEP TWO – Tuning rate D**

You have found a nice rate\_P value (or so you think), so leave that fixed for now. For example 0.110

Set a range of rate\_D from 0.000 to 0.025.

Throttle up in hand with rate\_D at 0.000, things should be as you left them at the end of step one.

Now start turning up rate\_D, again move the sticks, shake the beast about, you will notice that at a fairly precise point you start to see very fast oscillations. This is your maximum rate\_D, I'd set it just under where you can induce these fast oscillations by moving the sticks fast of shaking a leg.

Now you can go back to rate\_P and tune that again, you should be able to increase it considerably!!! So to step three.

NOTE: once you have found your rate\_D you can try something interesting - you will have to hold onto your bird for this obviously. Set rate\_P to zero, then crank up your rate\_D as you just have.

You will discover that these fast oscillations occur at pretty much the same level of rate\_D, no matter what you have set rate\_P to. This is why I think it's more than possible to **start tuning rate\_D** then move on to P. But that's a shortcut, and this is not what this guide is all about.

### **STEP THREE – Tuning rate P even more**

You have your maximum rate\_D set, now try increasing rate\_P again.

You should be able to add at least 30%, maybe more before you get the oscillations. It will react faster to stick movement, be difficult to move with your hand and just feel very solid in the air.

Congratulations, your copter should be flying better than it ever has done before.

### **STEP FOUR**

Using your 'happy' rate\_P and rate\_D values, start tuning rate\_I. This is better done in-flight and will feature in part two of this guide. Basically just keep tuning it up until you notice a loss of 'feel'. It will hold an angle better for you, unless it's too high then things go sluggish and eventually oscillate slowly.

### **STEP FIVE - HEAD OUTSIDE**

For now set Stab\_P at 3.0 - 3.5, closer to 3.0 if you have a high power machine, just to get you flying in stab mode. Then spend your time tuning rate\_I and stab\_P.

If you find your set up immediately too harsh, back off on rate\_P and rate\_D by 10% each, and try a lower i-term, if it's still too harsh do that again!

All that's left is to watch the video below so you can see what I'm describing and head outside for fine tuning. Oh, and await part two of the guide and amendments from any discussions raised, then we will have a proper tuning guide that will be integrated into the wiki advanced tuning guide.

## **Alcune precisazioni:**

### **Rate (acro) mode tuning**

Rate will only use the gyro.

In the first step, concentrate solely on the rate 'P', 'P' is the rotation value to counter external rotation effects caused by influences like wind for example. The gyro will detect such rotations and feed the control loop.

High frequency oscillations will occur when 'P' is to high, if is feels to sluggish or it is not flyable at all (see vid) it is to low.

The rate 'I' value is normally not used, but as I explained, I like to tune the rate as good as possible so the next step is tuning the rate 'I'

The 'I' in rate is used for angle hold, if it is 0 or close to 0 the copter will never hold an angle and swing around it's axis. Increase the 'I' to a level that you feel comfortable with and the copter holds its angle.



Now that rate (acro mode) is actually flyable it will influence the stable mode PID as well. In general you can lower the 'P' because the Rate 'P' is now more prominent.

### Stable mode tuning

Stable mode uses the Gyro and accelerometers. Accelerometers will detect the direction of the acceleration. With some coding the accelerometers can measure the angle and use it to level the copter.

In the stable mode again first concentrate on the 'P' value. 'P' is the power for the motor to achieve the angle. A too high 'P' will also introduce oscillations but in a different frequency than the rate 'P'. A too low 'P' will not hold level.

I is the time needed to reach the maximum 'P'.

And ... if 'I' feels good, try to decent rapidly.

If it wobbles :

**Further increasing 'I' will try to eliminate the vortex effect (strong wobble) when descending rapidly with quads**

Experiment with different props, maybe a smaller prop diameter.

- Larger props = lower P,
- Smaller props = higher P

Oh and as promised, below is Part One of the Summary Guide.



## Summary Guide

1. Tune up just rate\_P
2. Tune up rate\_D
3. Tune up rate\_P more
4. Tune rate\_I as best you can
5. Take it outside with a stab p of 3.0 ish and tune rate I and stab p in flight, to your liking, maybe adjusting the other params too slightly, to get it just as you want it.

## Summary of what you are looking for

### RATE-P

Too much rate P will oscillate quickly, and cause to copter to sound angry under stick input, bouncing rather than smoothly following your inputs. It will also shake more at full throttle and under hard turning.

Not enough you will not feel like you have full control. It will feel sloppy and be very easy to over correct with your inputs. It will feel delayed.

Perfect is where it feels locked in, stiff in the air, but not shaky. (although if I'm sport flying I turn it up a bit for maximum 'wang' and just tolerate the slight oscillations)

### RATE\_I

Too much rate\_I will oscillate if you get high enough (a much slower oscillation than a rate\_P shake). But quite a long while before it oscillates it will have other detrimental effects on flight performance, like a sluggish feeling or a tendency to flip over on take-off. This is why I suggest tuning this in flight rather than in your hand/jig.

Not enough will cause the copter to get pushed by a constant wind, then it will fight back using just P. It will not hold a very firm angle during forward flight and will need more correction. This will not be as smooth as it could be in either case.

The perfect amount will cause the copter to lean gracefully into a constant wind, but also allow you to set a lean angle and stay there as you fly about. As you dial rate\_i in pay close attention to the feel of the copter, you are not looking to create oscillations here at all, you should notice a strange 'feel' long before this point.

### RATE\_D

Too much rate\_D will oscillate very fast, you will see a twitch forming then a fast buzzing oscillation

Not enough rate\_D will simply mean you can't dial enough rate\_P and so you will suffer the effects of having rate\_P too low.

A perfect rate\_D will help fight the wind and follow your sticks as its fast to react, but will also allow you to reach a maximum rate\_P level for you frame, thus giving better control.

## Tweaking ArduCopter parameters

(Please also read the ArduCopter Tuning Guide [here](#))

If you're using an airframe other than the official ArduCopter kit, you may need to change some PID settings (PID stands for Proportional-Integral-Derivative and is a standard control method. You can read more about it [here](#)). A full guide to PIDs is at the bottom of this page.

You can adjust PIDs interactively in the Mission Planner's Configuration tab. Start with the "[ArduCopter PIDs](#)" page, which breaks out the most important settings and gives instructions on how to use them. More configuration settings can be found in the Standard Params and Advanced Params if you need them (you probably won't).



## Basic performance troubleshooting

### Remember two things:

- Start by using Stabilize Mode for tuning, if it doesn't fly correctly in Stabilize it **will NOT** fly correctly in any of the other modes!
- First use the defaults! You shouldn't have to deviate very far from the defaults that are loaded with the firmware, if you have to change them then use small increments and one at a time. If you have to use a value that is very different then either something else is wrong or your 'copter is very different from everyone else's.

For normal flight, there is really only one term you should care about and that is the **Rate Roll P setting** (Rate Pitch is normally slaved to that, as the checkbox on the screen above shows). Use the in-flight tuning method with CH6 to nail this term in a few seconds. Make it as aggressive as you can before it begins to oscillate. You will have great performance descending straight down if this term is higher. You may want to tune it down a hair more if you like a softer response.

Here's an example: the default setting for Rate Roll P is 0.14. *This is optimized for the standard Jdrones/3DR quadcopter with 850 motors and a 10" prop. If, however, you're using the bigger 880 motors and a 12" prop, which are much more powerful, that Rate Roll P will be too aggressive and your quad will seem quite unstable. So in that case, lower your Rate Roll P to 0.1 or 0.09, and you should see much more stable flight.*

**The same is true for Altitude hold.** Throttle Rate P needs to be adjusted to get optimum altitude performance.

These two terms go a long way to making your copter perform well, and are highly dependent on the thrust to weight ratio of your copter. More thrust = lower gains.

- **My copter oscillates slowly (larger movements) when stabilized:** Lower your gain in Stabilize Roll P.

- **My copter oscillates quickly (smaller movements) when stabilized:** Lower your gain in Rate Roll P.
- **My copter gets wobbly when descending quickly:** Raise your gain in Rate Roll P. You can tune most of this out, but it's impossible to descend into your prop wash without some wobbles.
- **My copter is too sluggish :** Raise your Stabilize Roll P gain. These gains make ask the copter to respond faster to angle errors.
- **My copter yaws right or left 15° when I take off:** Your Motors are not straight or your ESCs are not calibrated. Twist the motors until they are straight. Run the ESC calibration routine.
- **My copter angles back in the opposite direction I'm flying after forward flight:** Raise your gain in Stabilize Roll P.
- **After aggressive flying my copter leans to one side 10 - 30°:** Do what ever you can to remove vibrations and isolate APM. You can even try and add a small weight to APM to keep it from vibrating. Also, just land for a few seconds, then continue flying, it will give APM a chance to correct without the vibration.
- **My copter won't stay perfectly still in the air:** Run the level command on a flat surface. (hold disarm for 15 seconds to invoke.) You can also fly in auto-trim mode in a windless (important!) environment. Any wind will cause the changes you make to work against you when the quad rotates 180°. You **can** use your radio pitch and roll trims, but remember to re-center them when you set up your radio with the config tools. Using the radio trims can have a negative effect in SIMPLE mode when yawing. Never Trim Yaw, Your copter may start spinning on it's own. (Quads are also susceptible to drafts. They will need constant corrections unless you install an optical flow sensor. One day...)
- **My copter flies well, but then dips a motor arm in a fidgety manner while hovering:** Your motor may be going bad or the bullets connectors between the motor and ESC may be at fault. Vibration from a bent shaft or unbalanced props can make bullet connections fail momentarily stopping the motor. A motor with bad bearings takes more power to spin. The ESC could be cutting out to protect itself. Or it maybe flying slow enough to stall the motor. Attach a current sensor between your battery and the motor and test the difference between a good motor and the bad one. If you are seeing higher draw in the bad one, replace or fix it.
- **My copter suddenly crashed for no good reason:** The I2C port, used to talk to the compass, has a rare but very serious flaw which will block code execution To avoid any issues be sure to solder the compass securely.
- **In Loiter, my copter constantly overshoots:** Try and increase your Nav\_P term. You can also tune down your I terms because in some cases, the Nav\_I or Loiter\_I can causes overshoot. Making the I terms 0 when in still wind is the best way to tune Nav\_P.
- **My alt hold above 10 feet is only about 1-2m accurate:** That's actually the best you can achieve. The baro sensor is sensitive to light and wind. Use a felt cover to protect it from wind and make sure you don't glue the hole and seal it up.
- **My copter increasingly swings up and down in alt hold.** It eventually get's down to the ground: Your THROTTLE\_P is too high or low. You don't need a lot of P to do alt hold. Think of how much you move the throttle to hold alt perfectly. Not much! That's what you need P to do. I will ramp up as your battery goes lower to make up the difference.
- **I have a sonar installed but I can't get it to work.** Your sonar may be picking up electrical noise from the ESCs. Be sure to move it a few inches from any ESCs and try a shielded cable, if possible. If it's cold outside, your sonar may stop working.

- **My quad loiters by rotating in a CW or CCW circle:** Adjust your compass declination until it stops circling.

---

## In-flight tuning

You can tweak parameters in flight from Mission planner (needs a telemetry link or USB cable) or with the Transmitter (Tx) channel 6 knob or lever (You need an extra free channel that is controlled by a knob or slider on your transmitter(Tx) to use the CH6 fine-tuning function, you can use any free channel on your Tx but it must be plugged in to the CH6 option on APM. So CHx on Tx > CH6 position on APM.)

**Method 1:** (See also Method 2: Flying Attitude Tuning below)

Select the "ArduCopter Configuration" tab in the Mission Planner, choose the CH6 option you'd like to tune. e.g. CH6\_RATE\_KP.

From MP v1.1.44 onwards these high and low values can be adjusted in boxes just below the CH6 options. Alternatively you can also adjust TUNE\_HIGH and TUNE\_LOW in the left hand pane of the Configuration > Adv Parameter List.

Input the high and low limit of the parameter you'd like to tune, choose a value of + and - around 30-40% of default, (so for CH6\_RATE\_KP set 'low' to 0.100 and 'high' to 0.180 from the default of 0.140)

Then click 'Write Params'. To check the values click 'Refresh Params', your high and low values should be correct in both the Parameter List tab (under TUNE\_HIGH & TUNE\_LOW) and in the adjustment boxes. If not enter them again.

To test that the values are changing: move the lever or knob then click 'Refresh Params', the value should be displayed in the center pane in the respective box, move it again and click 'Refresh Params' again, the value should've changed. Alternatively select the Terminal tab, type 'test' and then 'tune' and your value will scroll down in the window, adjust the knob or lever to see the values change.

Now set your CH6 knob or lever to between 0 and 25% on your transmitter, then carefully test fly, land and tune the parameter with the knob or lever, test fly again.

To read your final parameter, leave the knob or lever on your Tx in your chosen position, connect to the MP then click 'Refresh Params', the value should be displayed in the in the respective box in the Parameter List tab (under TUNE\_HIGH & TUNE\_LOW). Alternatively select the Terminal tab again, type 'test' and then 'tune' and your value will scroll down in the window. Enter this value into the correct position in the Configuration tab.

Now tune another parameter or, if you're finished, set the parameter to CH6\_NONE (important!)

The main value to tune is your Rate\_KP value. This value adjusts whether your copter is:

- no response to control inputs (too low)
- sluggish (a little too low)
- stable (just right)
- over-active (a little too high)
- or oscillating (too high)

**Here are all the parameters one can tune:**

CH6\_NONE 0  
---Attitude  
CH6\_STABILIZE\_KP 1  
CH6\_STABILIZE\_KI 2  
CH6\_STABILIZE\_KD 29 // duplicate with CH6\_DAMP  
CH6\_YAW\_KP 3  
CH6\_YAW\_KI 24  
---Rate  
CH6\_ACRO\_KP 25  
CH6\_RATE\_KP 4  
CH6\_RATE\_KI 5  
CH6\_RATE\_KD 21  
CH6\_YAW\_RATE\_KP 6  
CH6\_YAW\_RATE\_KD 26  
---Altitude rate controller  
CH6\_THROTTLE\_KP 7  
---Extras  
CH6\_TOP\_BOTTOM\_RATIO 8  
CH6\_RELAY 9  
---Navigation  
CH6\_TRAVERSE\_SPEED 10 // maximum speed to next way point  
CH6\_NAV\_KP 11  
CH6\_LOITER\_KP 12  
CH6\_LOITER\_KI 27  
---Trad Heli specific  
CH6\_HELI\_EXTERNAL\_GYRO 13  
---altitude controller  
CH6\_THR\_HOLD\_KP 14  
CH6\_Z\_GAIN 15  
CH6\_DAMP 16 // duplicate with CH6\_YAW\_RATE\_KD  
---optical flow controller  
CH6\_OPTFLOW\_KP 17  
CH6\_OPTFLOW\_KI 18  
CH6\_OPTFLOW\_KD 19  
  
CH6\_NAV\_I 20  
CH6\_LOITER\_RATE\_KP 22  
CH6\_LOITER\_RATE\_KI 28  
CH6\_LOITER\_RATE\_KD 23  
  
CH6\_AHRS\_YAW\_KP 30

**Method 2: Flying Attitude Tuning with CH6:**

In this case you will tune your 'copter while it is actually flying.

This assumes that the 'copter is fairly flyable/controllable and that the pilot is experienced enough to easily keep the 'copter at eye level and in the same place for at least 60 secs. You will also preferably need an open space of perhaps 10mx10m (30ftx30ft).

Follow the setup in Method 1 above.

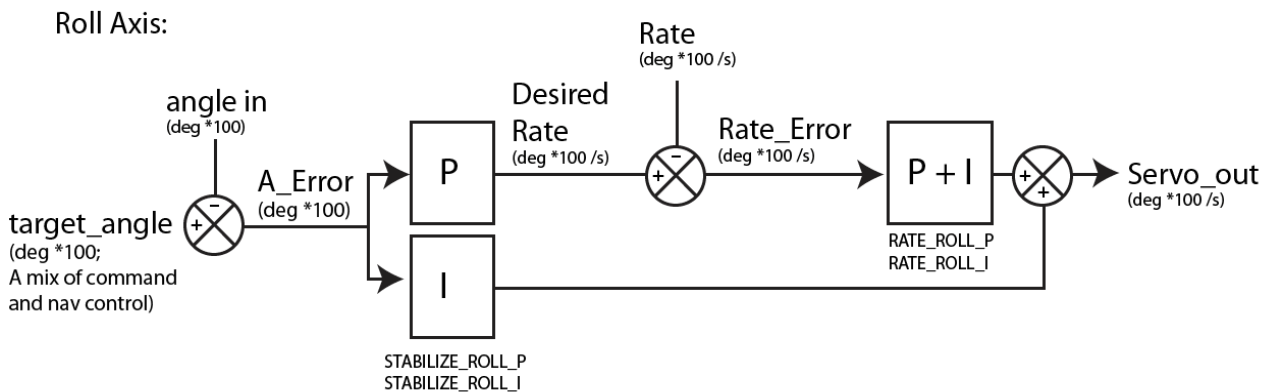
Do Rate\_P first, choose a value of + and - around 30%-40% of default, (so for Rate\_P set 'low' to 0.100 and 'high' to 0.180 from the default of 0.140. Be careful not to set these values incorrectly - don't set 0.100 as 1.000 or 0.010!

Hover at about 1.5m (5ft) and bank the quad briefly left and right, then either land and turn the knob/lever that controls the channel, or if you can fly and turn the knob/lever then do so, turn it up and down, note the difference and choose the flying style which "feels" best to you. Higher values might cause oscillations so back off a little until it flies without oscillations but isn't too sluggish. To read your final parameter, leave the knob or lever on your Tx in your chosen position, connect to the MP then click 'Refresh Params', the value should be displayed in the in the respective box in the Parameter List tab (under TUNE\_HIGH & TUNE\_LOW). Alternatively select the Terminal tab again, type 'test' and then 'tune' and your value will scroll down in the window. Enter this value into the correct position in the Configuration tab.

Now tune another parameter or, if you're finished, set the parameter to CH6\_NONE (important!)

Note: (Set I values generally low, eg 0.000-0.100.)

## How PIs work:



The new controller for AC2 is a variation on the above idea. We take the angle error and generate a desired rate of rotation. This rate of rotation (in degrees) is compared to the current rate and multiplied times Rate\_P. The value is converted into PWM and sent to the motors. This gives us numerous benefits such as being able to specify the desired rate of travel, rotation or altitude change. It also gives us a second I term to control how must we compensate to achieve the desired rate.

## What PIDs mean for performance

Stabilize is used for all modes except Acro. It's intended to be symmetrical, but pitch and roll have been broken out so you can fine tune a copter that is bearing a load in 1 axis. This might be a result of using two battery packs hanging on either side of the frame, or a very long thin pack. If one axis has a higher moment of inertia than the other axis, your ideal tuning parameters will not be symmetrical.

- STABILIZE\_P: The rotation rate at which you want to correct any errors. The higher this is faster the copter will attempt to achieve the desired attitude.

- STABILIZE\_I: Used to account for CG variations, weak motors or persistent external forces.
- STABILIZE\_I<sub>max</sub>: Maximum amount the copter can compensate for these imbalanced forces.
- RATE\_P: The most important value! This gain controls how much thrust you need to output to achieve the desired rate of rotation. High thrust/weight copters will require a lower value, lower thrust/weight will require a higher value. A too-high value will oscillate around 5-10hz. You want this value to be slightly lower than the value that causes the oscillation. Aggressive tuning may result in exactly 1 overshoot and return, which is considered acceptable. Use CH6 tuning to adjust in the air for best performance.
- RATE\_I: Used to help a copter achieve a desired roll rate. Not used by default as this can be very difficult to tune properly and can be confusing. If you are just starting out, set this term to 0.
- RATE\_I<sub>MAX</sub>: The maximum amount of Rate\_I that can build up. This is also not used in a basic setup. Having a 0 iMax will make Rate\_I completely ineffective, no matter how high the Rate\_I is.

Yaw is used to hold a particular Yaw angle. If your copter wants to spin naturally, you won't be able to hold an exact heading. You will instead drift a few degrees until P gets significantly high to stop rotation.

- STABILIZE\_YAW\_P: The desired rate at which the copter will return to the target heading. If this is too high, it could cause an oscillation.
- STABILIZE\_YAW\_I: Acts like a trim to overcome poor copter balance. Defines time it takes to achieve max value. Higher = faster.
- RATE\_YAW\_P: Used to the amount of control authority the AC2 can use to achieve zero yaw rate. If this is too low, you will never be able to stop a rotation. If this is too high, it will yaw-oscillate.
- RATE\_YAW\_I: Not used

LOITER is used to control how much the copter will pitch towards the Loiter target while trying to hold a position.

- LOITER\_P: The rate at which the copter will move towards the target point. If this is not high enough, the copter will not be able to fight high winds and will drift. If it's too high, it will oscillate around the target.
- LOITER\_I: This will help the copter fight winds while having a zero error. However use it with caution because it will also cause an oscillation if it's too high.
- LOITER\_I<sub>MAX</sub>: The maximum possible buildup of Loiter\_I

NAV\_WP is used to control the rate of speed of the copter towards the target.

- NAV\_WP\_P: we use our speed (4m/s as defined by WP\_SPEED\_MAX) offset as the error. high numbers = more pitch to achieve the desired speed
- NAV\_WP\_I: Allows us to ramp up against the wind. Higher value ramps faster.
- NAV\_WP\_I<sub>MAX</sub>: Amount of Pitch we can add to overcome wind

Alt Hold is used to hold a position using the relatively noisy Barometric pressure sensor.

- Altitude hold P: Used to convert altitude error in centimeters to a desired climb\_rate in centimeter/second. Higher = faster climb rate.
- Altitude hold I: Used to account for a copter having trouble holding altitude, usually due to a low voltage battery.
- Altitude hold I<sub>MAX</sub>: Amount of throttle we can adjust (units: 1000 = 100%)

Throttle Rate is used to dampen the copter and control the rate of climb

- Throttle Rate P: amount of throttle output used to change the climb rate

- THROTTLE\_I: compensates for error in achieving desired climb rate (zero by default. We use Altitude hold I to do most of the work.)
- THROTTLE\_IMAX: Amount of Throttle\_I we can add or remove to achieve desired climb rate.

## ReturnToLand

**Nella versione 2.8r2 è impostata l'altitudine a 1500 cm**, ovvero 15metri.

RTL\_HOLD\_ALT é l'altezza minima che il quad tiene quando é in RTL ed é altezza dal suolo dal quale parte.

Per le prove 15 metri sono tanti. Meglio abbassare a 8 metri le prime **volte e diminuire anche la velocità di RTL**. Le manovre di RTL sono assai piú violente delle analoghe manovre in MultiWii.

*#define RTL\_HOLD\_ALT 800*

*Per tutti quelli che vorranno testare per la prima volta il RTL, ricordatevi di impostare un'altezza piú bassa rispetto a quella di default.*

**Oppure impostate il valore a zero** per mantenere sempre la stessa quota, in stile multiwii. Però **la possibilità di alzarsi e tornare in home** è molto meglio, vedi possibili ostacoli come ad esempio gli alberi ...

## TEST 2

-Calibrato la trasmittente, configurato i modi di volo tramite il **canale 5** e gli switch a 3 posizioni + switch a 2 posizioni per ottenere le **6 diverse modalità** di cui per ora configurate solo 3 (stable, alt-hold, loiter), impostato il **canale 6 ad un pot. della tx per regolare il P\_rate** ed il **canale 7 per attivare l'RTL** (settato per ora ad **altitude 0**, ovvero **rientra in RTL all'altitudine alla quale si trovava**).

Settata la **declinazione magnetica, autotrim**, etc. si comincia a volare sul serio.

Con le regolazioni PID di **default oscilla parecchio**, la classica oscillazione da sovracompensazione. Abbasso il *pitch Rate P con il potenziometro della TX verso 0,10* e lo **stabilize P a 4**, le oscillazioni si riducono sensibilmente.

**Provo l'alt-hold.** *Questo non va, nel senso che anche se abilitato il quadricottero non tiene molto bene l'altitudine, tende salire ma piú frequentemente a scendere, diverse volte fino a toccare terra, insomma non è molto stabile. Anche in loiter non è che sia proprio stabilissimo, tende a derivare lentamente e stranamente sempre verso la mia sinistra e prima che si schianti ritorna in stable.*

Pure il pilotaggio appare un po' "gommoso" nel senso che i comandi non rispondono istantaneamente.

Rispetto ad una multiwii in acro effettivamente il pilotaggio di MPNG in stabilize mode appare particolare. Sembra che la scheda voglia pilotare il quad e tu la disturbi con i tuoi comandi. Da solo sembra volare meglio.

Nei miei test il baro non va male, ma occorre attivarlo quando il quad é stabile.

Prima di attivare il gps (loiter o rtl) mi sono accertato che il quad volasse quasi da solo senza toccare la radio ripetendo due o tre volte la procedura di autotrim.

In ogni caso il risultato é documentato ai post 114 e 115 di questo 3d.

ho fatto qualche altro voletto per regolare meglio i PID rate. C'è da smanettarci un po' ma alla fine le cose migliorano! Dopo una serie di prove ho aumentato:

**pitch/roll rate\_I** portandoli al valore **0,016** (da 0,010 default)

**pitch/roll rate\_D** a **0,007** (da 0,004 default)

questo mi ha consentito di riportare il



**pitch/roll rate\_P** al valore di **0,148**

senza fastidiose oscillazioni nella risposta e con un quadri più stabile e reattivo ai comandi.

Ho innalzato anche un pochino i rate\_P e I dello yaw per ottenere un puntamento più pronto. Già che c'ero ho cercato di migliorare ulteriormente la protezione del barometro alle turbolenze delle eliche.

Ora la situazione è sensibilmente migliorata su tutti i fronti: sia in stabilize che in acro, sia l'alt-hold e loiter. Era così stabile che sono salito penso attorno ai 30-40 metri senza patemi.

*Domani proverò a ridurre i P dello stabilize da 4 a 3.5 per vedere le differenze.*

*Luciano osserva:*

Io ho toccato **solo il rate P portandolo direttamente a 0,12 lasciando tutto il resto di default**. Devo dire che rispetto a multiwii ho trovato MPNG più semplice da settare.

Dal punto di vista del pilotaggio trovo invece che multiwii sia più fluido e comodo da pilotare, soprattutto in ACRO. MPNG in stable mode tenta sempre di ritrimmare il modello per metterlo piano e devi sempre combattere per pilotarlo. Anche in ACRO non mi piace molto e preferisco ampiamente la modalità ACRO di Multiwii. **La modalità STABLE di MPNG è interessante** soprattutto se vuoi **fare volare il modello da solo (LOITER, RTL, WP)**. E' anche utile se stai imparando perchè se molli i comandi lui torna perfettamente orizzontale. Trovo ottimo il **volo pilotato lasciando attivo il LOITER**. Ti consente di pilotare il mezzo meglio che in Stabilize **e se molli i comandi lui si ferma dove è**.

La discesa veloce destabilizza il quad. Nel senso che in modalità alt-hold o loiter, scendendo rapidamente il quad oscilla parecchio, come se tentasse di correggere disperatamente, eppure con il throttle sto sotto il 10%, al di sotto della banda morta, quindi il comando di discesa dovrebbe essere chiaro.

In ogni caso dipende sicuramente dal frame e dalla configurazione eliche-motori, megapirate mi ha costretto a fare un po' di PID tuning in più del solito.

Una cosa che ho notato nel RTL, oltre che alla rapidità di rotazione dello YAW mentre sale in quota per puntare verso la home, il quad parte bello lanciato per poi fermarsi circa 3, 5 metri prima della home ed avvicinarsi piano piano fino al punto esatto.

Con multiwii impostando una velocità elevata arrivava al punto di home a tutta velocità per poi superarlo ed immediatamente frenare e cercare di correggere la posizione.

Giorni fa ho avuto un problema con il ReturnToLand. Il quad alla fine è salito a 100metri! per poi tornare a casa. Mi sono accorto che nel MissionPlanner, nella sezione per pianificare i WayPoints, c'è l'**altezza di default** impostata a **100 metri** ed io sbadatamente ho provato a scrivere i waypoints nell'EEPROM e successivamente rileggerli per semplice test con i relativi pulsanti.

Così facendo ho sovrascritto anche l'altezza di RTL.

Me ne sono reso conto solamente ieri rileggendo il wiki del MissionPlanner.

## Collaudo dei WayPoints

Oggi ho finalmente provato il volo con WP.

Ho impostato l'interruttore a 3 posizioni **su Stab/Loiter/Auto e uno a due per attivare RTL sul canale 7**.

### Successivamente:

#### CH5

- 1 - STABLE
- 2 - LOITER
- 3 - AUTO

#### CH7

- 1 - OFF
- 2 - RTL

L'ALT Hold da solo alla fine l'ho tolto (o vado in stable o vado in Loiter)

Quando vado in AUTO, nel caso voglia abortire la missione, attivo il canale 7 che mi fa l'RTL

Se non volo in AUTO sostituisco con BT su telefonino AUTO con RTL in modo da avere

## CH5

1 - STABLE

2 - LOITER

3 - RTL

che in fondo sono i tre modi più importanti...

Ho programmato una missione a 5 punti portando il quad un po' più lontano.

Messo il quad in Loiter. Attivato Auto. Il quad è partito verso la sua missione che ha compiuto perfettamente fermandosi in Loiter a fine percorso.

Ho rifatto diverse volte il medesimo percorso ed ha SEMPRE funzionato tutto bene. Ho provato ad abortire la missione attivando RTL e il quad è tornato a casa buono buono. Ho provato anche ad abortire la missione riportando il quad in Loiter per poi rifarlo partire in Auto e il quad è ritornato al WP1 rifacendo il suo percorso.

Per l'atterraggio automatico attenzione che **non spegne i motori da solo** ma va fatto **a mano rimettendo il stabile e togliendo gas**, almeno su arducopter fino alla 2.9, con le nuove versioni dovrà spegnerlo da solo.

Un'altra funzione bella è il **fly-by-wire, cliccando un punto sulla mappa** lui lo raggiunge e si mette in LOITER aspettando il comando successivo ma naturalmente per il fly-by-wire occorre una connessione radio con mavlink... Si trova a 25\$ da RCTimer e da Sunsky

Sicuramente bisogna lavorare bene sui PID. Man a mano che ci si avvicina ai valori giusti per la propria configurazione di quad, le cose migliorano sensibilmente.

Oggi per esempio ho ridotto di 0.5 il P dell'Alt-hold ed ora in discesa rapida nelle modalità alt-hold e loiter sbalanzola molto meno di prima.

## Altre missioni sempre più complesse.

Questi gli eventi:

- **WAYPOINT** (a diverse altezze)
- **LOITER\_UNLIM** (buono come punto di arrivo della missione)
- **RETURN\_TO\_LAUNCH** (funziona...ma preferisco attivarlo da CH7)
- **TAKEOFF** (funziona....ma non sempre...a volte mi ribalta il quad)
- **LOITER\_TIME** (buono per fermarsi ad un waypoint per un po' di tempo)
- **LOITER\_TURNS** (orbita in modo un po' buffo intorno al waypoint con la prua sempre verso il waypoint)

Visto che i percorsi si sono allungati ho provato ad **alzare la velocità a 7m/s**. Effettivamente è parecchio veloce ...ma riesce comunque a rispettare i vari waypoint. Con quella velocità non riesce però a tenere la quota e in accelerazione tende ad alzarsi per poi scendere quando

raggiunge il WP. **A 5m/s va sicuramente meglio. A 3m/s è simpatico perchè lo puoi seguire a piedi** ma è troppo lento per percorsi lunghi.

## La velocità di navigazione in RTL e in AUTO si può cambiare almeno in due modi:

- in Mission Planner (o in Copter-GCS) c'è il parametro **WP\_SPEED\_MAX** in cm/sec che imposta la velocità di navigazione in volo autonomo

**Gestione della velocità:** nella pianificazione di una missione il metodo più comodo è sicuramente l'evento **DO\_CHANGE\_SPEED** che cambia la velocità (in metri/sec) da quel punto in avanti. Non è un WayPoint e l'evento va aggiunto senza coordinate, con solo l'indicazione della nuova velocità. Ho provato a variare diverse volte la velocità dopo diversi WP e funziona benone. L'ultima velocità impostata rimane attiva fino al prossimo reboot o fino al prossimo DO\_CHANGE\_SPEED

**Gestione dello YAW:** l'evento **DO\_SET\_ROI** (solo ROI sul Mission Planner) funziona proprio bene. Usando l'opzione 2 o 3 e mettendo le coordinate di un punto il multicoeso mentre si muove mantiene la prua verso quel punto e lo fa in modo molto dolce; adattissimo per fare riprese di un soggetto con il quad in movimento automatico. Naturalmente per comodità si fissa un punto waypoint sul Mission Manager e poi lo si cambia in ROI. Su MP il punto ROI rimane visibile con colore diverso. Anche in questo caso l'ultima l'impostazione dello YAW rimane attiva fino al prossimo reboot o fino al prossimo ROI

Ho provato anche **il takeoff automatico**. Non so se ho fatto tutto giusto. Ho messo il primo waypoint come takeoff, ho armato la scheda e **attivato AUTO**. A questo punto il quad è ancora fermo; basta **spostare un pochino dal minimo lo stick del motore e il quad decolla**. Un paio di volte è andato bene ed è veramente interessante vedere il quad che decolla sbalanzolando da solo. Una volta invece non è riuscito a decollare e tentava di ribaltarsi...ho rimandato queste prove dopo aver letto un pochino di più su questa procedura di auto-takeoff.

Per **l'autolanding aspetto la 2.9** che dovrebbe spegnere i motori perchè così com'è nella 2.8 non mi piace.

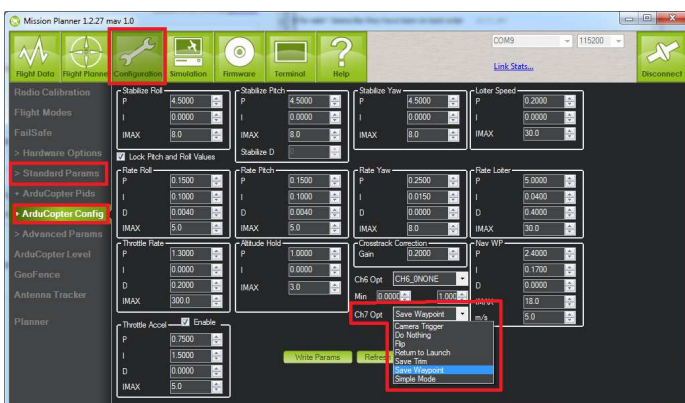
## Channel 7 Options: WAIPONTS

Si possono settare i waypoint al volo con il canale 7 anche senza collegare il PC.

Per prima cosa bisogna **settare la funzione nel MP, channel 7 save\_wp**, poi mentre si vola quando si attiva lo switch del canale 7 il sistema salva il punto memorizzato in altezza e posizione.

Meglio collegare il canale ad un pulsante che torna indietro da solo, **quando lo si tira memorizza il punto il cui è la scheda**.

Ovviamente quando si mette in auto ripercorre tutto il percorso con relativi wp.



Option	Description	Notes
Camera Trigger	Trigger the camera shutter	Set-up instructions <a href="#">here</a>
Do Nothing	Disables the channel 7 switch	
Flip	Triggers a flip on the roll axis	<b>Not working in AC 2.8.1.</b> Copter will rise for 1 second and then rapidly flip on it's roll axis. The copter will not flip again until the switch is brought low and back to high. Give yourself at least 10m of altitude before trying flip for the first time!
Multi Mode	Flip if CH6 < 1200, RTL if CH6 in middle, Save Waypoint if CH6 is > 1800	Not available in drop-down list in Mission Planner so can only be enabled by manually setting CH7_OPT parameter to "8"
<b>RTL</b>	Return to Launch	<b>Returns to the location that the motors were armed</b>
Save Trim	Save radio trims to APM	Detailed instructions are <a href="#">here</a>
<b>Save Waypoint</b>	Saves current location including altitude as waypoint	User must toggle to save WP. <i>If a mission is currently running it clears the mission and invokes a Return-to-launch</i>
Simple Mode	Simple Mode on when switch is high, off when low	Overrides the control mode settings setup on the Flight Modes screen

Note that your channel 7 switch must reach at least 1800 pwm for the auxiliar function will not be activated. You can check that the range of your channel 7 switch surpasses 1800 on the Mission Planner's radio config page



## MONITOR DELLA BATTERIA

Collegando alla AIOP un sensore **AttoPilot** si avrà il monitoraggio del livello di tensione della batteria.

--	--

Per la corrente si deve provvedere a collegare un carico (che sarà il quadri) e verificare con un Wattmetro/Voltmetro/Amperometro esterno se gli assorbimenti si eguagliano.

Il sensore è stato **collegato ad A0 per la corrente ed A1 per la tensione.**

In Mission Planner, nel menu Configuration->Battery Monitor si sceglie (anche perchè solo così ha funzionato):

Monitor - 4: Voltage & Current

Sensor - 0: Other

APM ver- 1: APM 2-2.5 non 3Dr

Battery Capacity - 2200 mAh (è la batteria da battaglia per le prove al banco)

Il parametro Sensor ho dovuto impostarlo ad **Other** perchè con il valore 2:AttoPilot 90A c'era un volt secco in meno rispetto alla tensione effettiva della batteria. Non ho capito il perchè ma credo che sia un problema del porting sotto MPNG.

Nel menu **Adv Parameter list** modificare il parametro BATT\_CUR\_PIN ponendolo a 0 (A0) mentre BATT\_VOLT\_PIN è stato posto ad 1 (A1).

Dopo aver misurato la tensione effettiva della batteria, ad es 11.3V, ed inserita nella casella Measured Battery Voltage, la AIOP si calcola il Voltage Divider (Calced), nel ns. caso 16,86334, restituendo in Battery Voltage il valore di tensione corretto a meno di qualche decimale.

### TARATURA ACCELEROMETRO

Ultima nota che riguarda **Mission Planner**: con l'aggiornamento alla versione **1.2.33** non funziona con la AIOP la funzione **ArduCopter Level** per tarare l'accelerometro.

Livellare il quadri ed azionare "Calibrate Accel" ([video demo here](#)).

Seguire le istruzioni che richiedono di piazzare il mezzo in 6 posizioni differenti.

Alla fine appare la scritta "DONE".

Final accelerometer offsets and scaling are stored in the INS\_ACOFFS\_X, Y, Z and INS\_ACCSCAL\_X, Y, Z parameters.


I valori sono specifici della scheda e possono essere ottenuti anche senza il frame.

Non è importante che i livelli siano perfetti (90°) ma che la scheda sia sempre perfettamente ferma.

*Funziona bene con ArduFlyer e firmware 2.9+ ma il metodo di taratura è complicato; prima bisogna mettere la board in piano, poi sul lato sinistro, poi su quello destro, poi naso in basso, poi naso in su ed infine capovolta*

non è opportuno aggiornare il Mission Planner fino a quando non sarà disponibile MegaPirateNG 2.9

## Kit per la telemetria a 433MHz da RCTimer

<p><b>433Mhz Radio Telemetry Kit</b></p> <p>Item : KIT-433</p> <p>Brand : RCTimer</p> <p>Weight : 0.150 kg</p> <p>Price : <b>\$38.99</b></p>	
--	--

Per le informazioni sull'impostazione della radio, riferirsi alla documentazione presente nella [wiki](#) di ardupilot-mega.

Ho inserito il Ground Module alla porta USB ed è stato subito riconosciuto dal sistema ed installato come una normale porta COM. Se si usa Win7 non c'è bisogno di caricare nessun driver, in caso contrario i driver possono essere scaricati dal sito di [ftdichip](#). E' possibile scaricare il tool di configurazione della radio da [RCTimer](#) (vedi mia DIR) ma non è strettamente necessario poichè le stesse cose possono essere fatte da MP che contiene, come documentazione aggiuntiva, gli schemi elettrici dei due moduli radio.

Per la connessione fisica dell'Air Module si può usare sia la USART3, già predisposta in MPNG, che la USART0/FTDI con l'accortezza però di impostare la velocità del modulo aereo a 115200, ma perdendo così la possibilità di utilizzare in contemporanea la USB on-board (tra l'altro necessaria se si utilizza la modalità **Terminal** di MP).

L'alimentazione del modulo aereo è stata prelevata, per mia comodità, dal connettore FTDI; ovviamente rx e tx tra modulo e AIOP vanno incrociati.

Al primo collegamento con Mission Planner è possibile verificare le impostazioni dei due moduli radio in **Configuration->Hardware Options->3DR Radio**, da dove è possibile fare l'upload del firmware. Per verificare i parametri dei moduli premere **Load Settings**; il **Connect** non

deve essere attivato altrimenti la porta COM risulta occupata e non si possono vedere i settings.

Infine, terminata la verifica, premere **Connect** impostando prima la COM corretta e la velocità a 57600 baud.

## Copter-GCS su Android

Si possono impostare sul campo le missioni utilizzando unicamente **Copter-GCS sul telefonino Android**

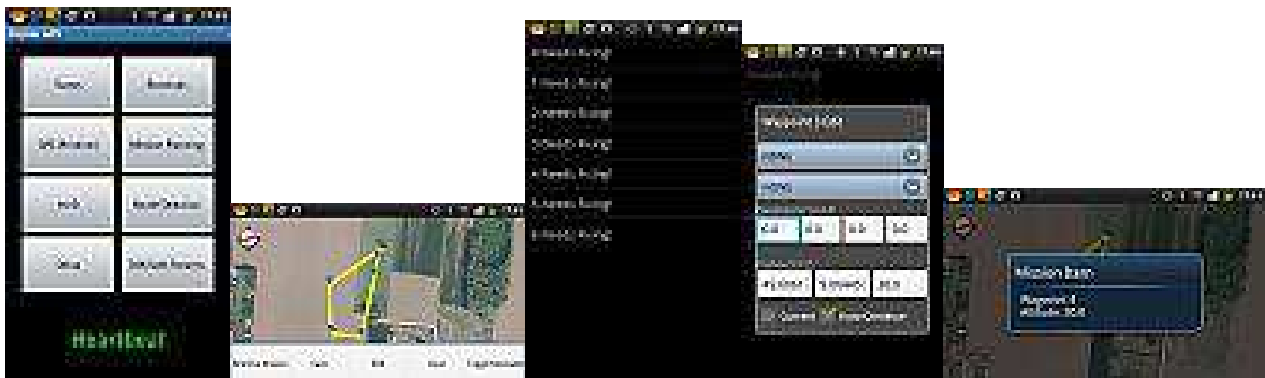
In effetti le funzioni non sono proprio come quelle che si trovano sul wiki di Copter-GCS: per far funzionare questo coso con il quale tutto sommato si puo' programmare una semplice missione solo con WP e altezze senza PC al seguito.

Dalla schermata gcs1 si seleziona Mission Planning. Se c'è già impostata una missione appare la schermata con la missione oppure ti chiede di caricarla da MPNG. Per caricare da MPNG puoi anche premere menu' e caricare la missione (vedi gcs3).

I WP si impostano semplicemente premendo sul posto dove vuoi i WP (come su Mission Planner) solo che non si riesce a specificare il tipo di WP (wp, loiter, rtl, ecc.). Dalla schermata gcs3 premendo edit si finisce nella criptica schermata gcs4...non si capisce cosa voglia dire ma premendo i vari waypoints appare la schermata gcs5 dove almeno puoi (devi) impostare l'altezza (è il parametro a fianco delle coordinate gps). Finito il giochino del settaggio dei wp e relative altezze dalla schermata gcs3 si puo' salvare la missione nel multicoso. Per sicurezza si puo' rileggere la missione dal multicoso e verificare premendo sui wp sulla mappa che le altezze siano quelle giuste (schermata gcs6).

...beh...non è la fine del mondo ma oggi pomeriggio nella triste e uggiosa giornata invernale mi sono fatto un po' di missioni senza portarmi il PC sul campo...

...se qualcuno riesce a trovare come definire il tipo di WP ce lo dica!

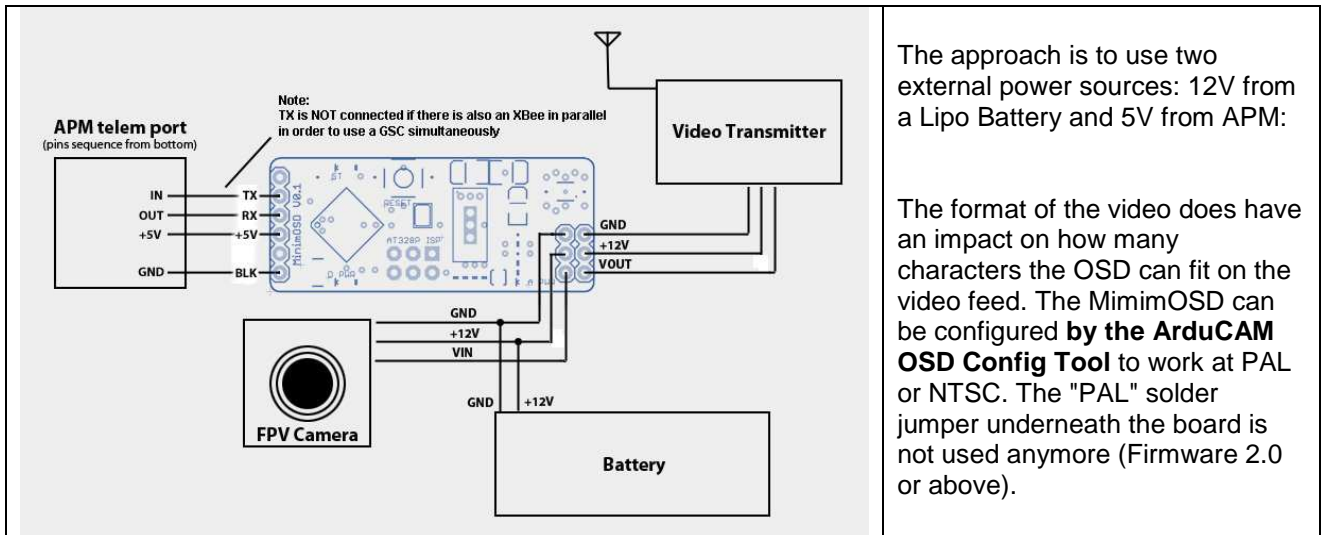


## MINIMOSD

Atto fornisce i dati ad AIOP, attraverso gli ingressi analogici A0-A1, che li inserisce nel flusso MAVLink. MinimOSD legge il flusso MAVLink ed intercetta i campi relativi a tensione e corrente e li sovrappone al video.

MinimOSD non ha ingressi analogici oltre a quelli relativi alla camera, ma è in grado, tramite la sua porta seriale, di interpretare i messaggi MAVLink che escono sulla porta di telemetria della AIOP. Questo schema da' un'idea di come funzionano le cose





The approach is to use two external power sources: 12V from a Lipo Battery and 5V from APM:

The format of the video does have an impact on how many characters the OSD can fit on the video feed. The MinimOSD can be configured **by the ArduCAM OSD Config Tool** to work at PAL or NTSC. The "PAL" solder jumper underneath the board is not used anymore (Firmware 2.0 or above).

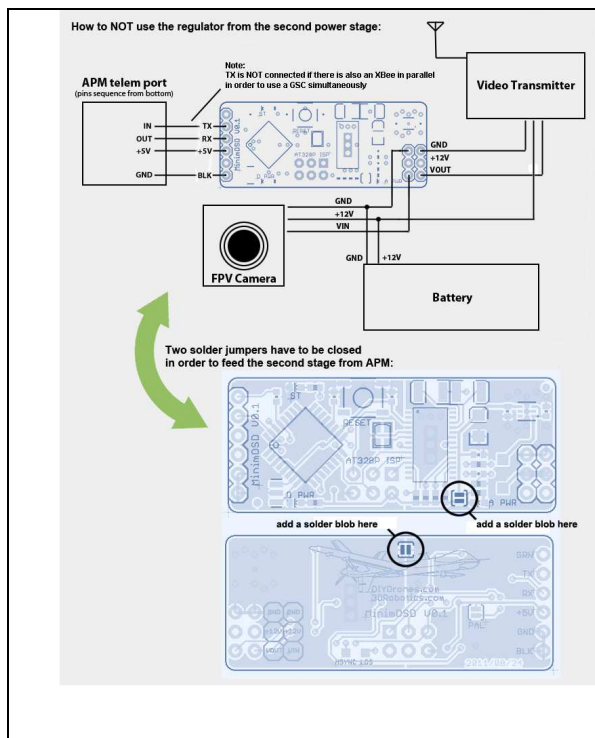
MinimOSD, compact and minimal Arduino-based on-screen display board with ArduPilot Mega and the MAVlink protocol.

**Features:**

- ATmega328P 8 bit microcontroller with an Arduino bootloader
- MAX7456 monochrome on-screen display
- Combine the two power and ground sections with solder jumper
- FTDI cable compatible and 6-pin ISP header
- +5V 500mA regulator for up to +12V supply input

**More information:** [Instructions](#)

Important note: **You cannot connect the OSD when your APM 2 is also connected via USB (they share the same port).** Make sure you disconnect your USB cable from the APM 2 board before attempting to use the OSD.



**OPTIONAL SETUP (for critical cooling conditions)**

The second stage regulator from the OSD board gets too hot on 12V video setups. If your frame has not a good air flow for cooling the OSD board you may want to feed the OSD entirely from APM. Probably it will add some noises from servos, but you'll be more safe by this way:

**Using a GSC simultaneously**

MinimOSD's TX is NOT connected if there is also a "wireless serial link" in parallel (E.g.: XBee) **in order to use a Ground Control Station simultaneously. To use a wireless serial link and MinimOSD at same time, you need to make an "Y" cable or prepare your APM IMU board to have a parallel "mirror" connection.**

Bellow is a picture of a nice solution using bent 20mm long male pin-headers.

Upper mirror output: "TX" from MinimOSD is NOT connected on the APM's telem "OUT".

Lower mirror output: all the four wires are connected to the wireless serial link.

## FAILSAFE

[APM2xFailsafe - ardupilot-mega - Official ArduPlane repository - Google Project Hosting](#)

**How it works.** Your RC transmitter outputs a PWM signal that is captured by your receiver and relayed to the APM. Each channel on your transmitter has a PWM range usually between 1100 - 1900 with 1500 being its neutral position. When you start your radio calibration on the mission planner, all your values will be at 1500. By moving your sticks, knobs and switches you will set your PWM range for each channel. *APM monitors your throttle channel and if notices a drop lower than **THR\_FS\_VALUE (Default is 950)** it will go into **failsafe mode**.*

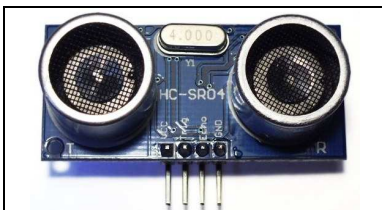
RC transmitters usually have a default range for each channel that goes from -100% to 100%, however most transmitters will allow you to extend this to -150% and 150% respectively. In the default setup, bringing your throttle to -100% will translate to a value close to 1100 and bringing it to -150% will translate to a value closer to 900. What we want to achieve is to let your receiver know that the throttle can go as low as -150% but keep the APM control range between -100% and 100%. Meaning that when flying, our throttle values will range between 1100 - 1900. But if we lose RC range, the receiver if set up properly, will drop to the lowest known throttle value of ~900. This value falls below the THR\_FS\_VALUE and will trigger the APM to go into failsafe mode.

First the APM will go into short failsafe (**FS\_SHORT\_ACTN, 0=Disabled, 1=Enabled**) when it detects loss of signal for *more than 1.5 sec*. The default setting for short failsafe is **Circle mode**. *If the loss of signal is longer than 20 sec the APM will go into long failsafe (**FS\_LONG\_ACTN, 0=Disabled, 1=Enabled**)*. The **default** setting for long failsafe is **RTL** (Return to Launch).

### Setup.

1. Enable throttle failsafe by setting THR\_FS\_Value to 1 (0=Disabled, 1=Enabled).
2. First turn on your transmitter and enable the throttle range to extend past -100%, we want to extend the throttle range past its low threshold.
3. Once this is done, bind with your receiver. This will let your receiver know the lowest possible value for your throttle channel.
4. Next revert the first change you made to the transmitter to limit the throttle to the original range.
5. Do the radio calibration using the Mission Planner.
6. Once the radio calibration is completed, drop the throttle on your transmitter and read what PWM value is being outputted to the mission planner on that channel.
7. Turn off the transmitter. You should see the value drop significantly. This will be the PWM value relayed to the APM in the event RC link was lost during flight.
8. Make sure THR\_FS\_VALUE is an adequate number to trigger the failsafe function on the APM.
9. Make sure FS\_SHORT\_ACTN and FS\_LONG\_ACTN are both enabled (set to 1).
10. Connect on the mission planner with your RC transmitter on. Verify on the bottom right corner of the HUD that you are "flying" in a non auto mode (Manual, Stabilize, FBW are ok).
11. Turn off your transmitter. After 1.5 sec the flight mode should switch to Circle. After 20 sec the flight mode should switch to RTL. If you observe this behavior, your failsafe function has been set up correctly.

## SONAR



Attivare il sonar nell'area **Hardware su Mission Planner** (inutile specificare il tipo perche su MPNG funziona solo HC-SR04). Provare il sonar su CLI (terminal mode) usando il comando *test* e quindi *sonar*. In effetti si vede molto bene la distanza tra il quad e la base di appoggio e se si alza il quad la distanza varia immediatamente e con precisione.

Vista la precisione del sonar pensavamo che a basse quote tenesse la quota molto meglio mentre invece non si vedono grosse differenze. Il sonar funziona e viene gestito da MPNG in quanto se si mette il quad in LOITER a 1,5m di quota, e si mette una mano sotto il sensore (senza toccare il quad) immediatamente si alza per poi ridiscendere quando si toglie la mano.



Utile magari nel caso di sorvolo di ostacoli pericolosi, tipo un albero oppure nell'autolanding, ma certe volte se sta volando in loiter a 2,3 metri di altezza, e c'è una piccola siepe il quad giustamente si alza di altri 2, 3 metri.

Come funziona sull'Arducopter è mostrato chiaramente in questo video: [Arducopter testing sonar altitude hold - YouTube](#)

essendo un misuratore di spazio, copia pedissequamente le variazioni di livello della superficie all'interno dell'area di captazione del sonar sopra la quale si trova il multirottore. Nello specifico anche le aiuole...

La Wiki Arducopter indica che le schede compatibili Megapirate utilizzano il classico sonar ultrasonico a trasduttori separati stile Arduino, che però essendo meno potente **viene fatto funzionare dai 2 metri in giù.**

#### Ultrasonic Wave Detector Ranging Module Distance Sensor

- Probe Main Technical Parameters
- Center resonant frequency: 40kHz  $\pm$  2kHz
- Static capacitance: 3300P  $\pm$  300P
- Resonant impedance: 120 $\Omega$   $\pm$  20 $\Omega$
- Frequency bandwidth (-3dB):  $\Delta f$ -3dB  $\geq$  2kHz
- Operation voltage: 300 ~ 500VP-P
- Limit voltage  $\leq$  1000VP-P
- Transmitting beam angle: 60 degrees
- Operation temperature: -40 ~ +80 °C
- Protection class: IP65

#### Module Performance:

- Voltage: DC5V;
- Static current: Less than 2mA;
- Output signal: Electric frequency signal, high level 5V, low level 0V;
- Sensor angle: Not more **than 15 degrees**;
- Detection distance: **2cm-500cm**
- High precision: Up to 0.3cm

It's important that the sonar sensor be mounted **at least three inches away from the body**, to avoid electrical noise from the ESCs and other interference. One good way is just to mount it on a strip of plastic attached to the body.

Qual'è la variabile utilizzata dal sistema di controllo per stabilire quando utilizzare l'uno più l'altro sistema di rilevazione? Se è il valore del barometro (eventualmente integrato dall'accelerometro Y) essendo questo un sensore di pressione assoluta, anche passando sopra il tetto di casa non dovrebbe far entrare in funzione il sonar poiché il barometro dovrebbe fissare come riferimento di quota zero il livello di pressione all'arming, quindi a terra. Però essendo il valore del barometro piuttosto variabile ed afflitto da una serie di disturbi legati a turbolenze e temperatura, potrebbe capitare che nell'hovering a 8-10 metri, possa entrare sporadicamente in funzione il sonar (anche se non dovrebbe) e se in quel momento il quad è sul tetto di casa il sonar potrebbe agganciarsi, con la conseguenza di far salire nuovamente di quota il modello.

Può essere che il sonar sia sempre in funzione e nel caso in cui venisse ricevuta una eco il main code integri con maggior peso anche quella variabile al fine di stabilire la quota. Occorre dare un'occhiata più approfondita al codice MegaPirate per capire come viene gestito il range\_finder.

Comunque non pare che il sonar possa salvare dallo scontro con un albero. Il cono di rilevazione non è molto ampio e qualora i rami più bassi dovessero attivare il sonar, probabilmente si è già troppo vicini per avere il tempo necessario di salire, questo specialmente nei voli a missione.

## Notizie per ArduCopter 2.9 (non MPNG)

The major improvement is we use inertial navigation to improve altitude hold. This increased reliance on the accelerometers means you must do some additional set-up before flying:

1. Perform the new accelerometer calibration in the mission planner ([video](#)). The auto-trim metho has also changed ([video](#)) Usa solo i trim della radio e poi atterra e attiva il CH7 per 3 sec.



Fate attenzione che la nuova versione, la 2.9.1 di arducopter ed il suo porting, soffrono molto di più le vibrazioni delle versioni precedenti poiché utilizzano anche gli accelerometri per integrare il barometro nella gestione della quota, è da quando è uscita la prima 2.9 beta che combattiamo con le vibrazioni e la soluzione migliore è quella di sospendere l'elettronica con oring.

Se le vibrazioni sono troppo alte in alt-hold o in loiter il mezzo non smette mai di salire finché non si ritorna in auto o manual.

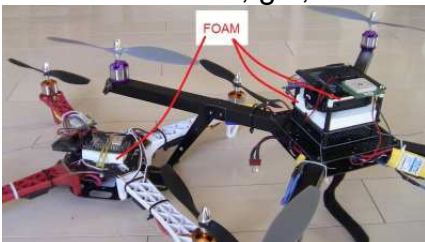
Un link ad un blog specifico di diydrones :

<http://diydrones.com/profiles/blogs/...age=1#comments>

Per verificare le vibrazioni si deve abilitare il log dei dato raw, in mp, in finestra terminal, logs, enable raw.

Dopo il volo analizzate il log e controllare i valori dell'asse z, devono stare tra -5 e -15 per essere accettabili ma più le creste dell'onda sono vicine a -10 e meglio è, comunque trovate tutte le info nel gigante post

2. Add vibration dampening foam between your frame and the APM. Some suggested materials: DuBro, gel, hk foam.



3. If upgrading from **2.8.1**, modify the throttle and altitude PID values:

Increase your Throttle Rate P, reduce I to zero, increase D

Increase Altitude Hold P, reduce I to zero

Tune Throttle Accel P and I terms but try to keep P about 1/2 the size of I



**Here is the list of major changes** (a more detailed list can be found in the release notes):

**Alt hold using inertial navigation** ([Leonard, Randy, Jonathan](#))

AUTO\_VELZ\_MIN, AUTO\_VELZ\_MAX parameters control the max climb/descent rate for the autopilot (cm/s)

PILOT\_VELZ\_MAX controls max climb/descent rate for the pilot (in cm/s)

**Landing improvements.** Copter will descend to 10m or until an object is sensed with the sonar. Then slows to 50cm/s descent (speed can be adjusted with LAND\_SPEED parameter). ([video](#)).

**Surface tracking with sonar.** Copter will attempt to maintain current distance from objects in front of sonar regardless of altitude. Only used in alt-hold and loiter, *not used for missions*. Sonar can be enabled/disabled with CH7 switch. ([video](#))

**Failsafe improvements.** including bug fixes, additional check for PPM encoder failure and implementation of battery failsafe. Set-up instructions are [here](#).

**Mediatek gps driver accuracy improvements and use of SBAS.** Instructions on upgrading your mediatek to firmware 1.9 are [here](#).

**Acro trainer.** Copter will return to be generally upright if you release the sticks in acro mode.

ACRO\_TRAINER : set to 1 to enable the auto-bring-upright feature

ACRO\_BAL\_ROLL, ACRO\_BAL\_PITCH : controls rate at which roll returns to level

**Camera control improvements:** (a) AP\_Relay enabled for APM2 (b) Trigger camera with CH7 or DO\_DIGICAM\_CONTROL command (c) Allow pilot override of yaw during missions and fixed CONDITIONAL\_YAW command.

**PPM sum support** for transmitters with as few as 5 channels

Performance and memory usage improvements.

As per usual **PIDs are optimised for the 3DR/jDrones quad with 850 motors and 10"**

**props.** If you're using more powerful motors/props and are seeing bad flight behaviour in stabilize, start by turning down Rate Roll P in 25% steps.

Special thanks to our testing team lead Marco and the dedicated bunch on the 2.8.1

release thread who put their copters at risk while testing the pre-release version. Some of their videos are here: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#)

## ACRO MODE con MPNG

Per volare in ACRO con MegaPirateNG bisogna continuare a mantenere la pressione sugli stick (come in STABILIZE) altrimenti lui livella velocemente e questo anche con un alto valore di I.

### There are two Acro modes available:

- the default is the Stabilized Earth Frame Rate controlled (**AXIS\_ENABLE = 1**),
- the second is Body Frame Rate controlled with no stabilization (**AXIS\_ENABLE = 0**).

We don't recommend flying in the un-stabilize mode as it will not self balance and is very difficult to fly.

When **AXIS\_ENABLE = 1** the controller is basically a rate controlled stabilised mode with a number of helpful features. There are two parameters that control **how fast the copter will move back towards hover if you let go of the sticks**.

**ACRO\_BAL\_ROLL** should be set to:

**Beginner** 200

**Intermediate** 100

**Expert** 0.

**ACRO\_BAL\_PITCH** should be set to:

**Beginner** 200

**Intermediate** 0

**Expert** 0

*These things are first and foremost for people to learn to fly ACRO.*

A beginner can adjust these settings to get flying characteristics **from a smooth STABILIZE to a rock solid ACRO** by adjusting the auto balance percentage.

For your taste I would recomend:

ACRO\_BAL\_PITCH of 0

ACRO\_BAL\_ROLL of 0

ACRO\_P of 4.5 to 9 (202.5 deg/s to 405 deg/s)

Le prove al banco confermano che i parametri **ACRO\_BAL\_PITCH** e **ACRO\_BAL\_ROLL** controllano la velocità con cui i rispettivi assi ritornano alla loro posizione di hovering. Un valore di 200 (valore max 300) si avvicina moltissimo allo STABILIZE ed è difficile notare la differenza, un valore di 0 esclude totalmente il ritorno.

Si possono utilizzare tutti i valori intermedi quindi è possibile regolare la velocità di ritorno oppure escluderla del tutto.

E' anche possibile scegliere velocità di ritorno differenti per ognuno degli assi, forse è preferibile avere il pitch bloccato ed un leggero ritorno sul roll.

## RISOLUZIONE PROBLEMI

Ad esempio i motori non rispondono più (ad es si armano ma non accelerano)

Bisogna aprire Mission Planner andare in Terminal e digitare:

Setup

Setup) ERASE

risponde Erasing EEPROM

Done

poi digitare

Setup) RESET

Ed alla risposta Y =factory reset

Digitare Y

e arriva la conferma  
*Reboot APM*

Poi è necessario ricalibrare la radio ecc. ecc.

## MegaPirate NG 2.9 RG

### Release Notes for MegaPirateNG 2.9 R6 (ArduCopter 2.9.1)

#### === How to compile MegaPirateNG ===

1. Download and install Arduino 1.0.1 - 1.0.3!!!!
2. See additional requirements for BlackVortex boards in the release\_notes\_BlackVortex.txt
3. Delete original libraries folder in Arduino folder
4. Copy libraries folder from MegaPirateNG distributive into Arduino folder
5. Select your Frame, Sensor, Tranmitter type in the APM\_config.h (**Quad X, ALLINONE by default**)

#### **MEGLIO**

*Creare una dir ARDUINO in Documenti.*

*Scompattare i files originali:*

*Arduino 1.3,*

*MPNG\_2.9\_R6 (da cui Arducopter, Libraries, tools, txt vari)*

*Trasferire da Arducopter, libraries e tools in radice*

Per cui si ha **Documents/ Arduino**

Arduino contiene:

Arducopter, Arduino\_1\_0\_3, Libraries, Tools

Lanciare Arduino, verificare sia Arduino Mega 2560 o Mega ADK.  
Quindi "Open" Arducopter.pde (potrebbe diventare .ino).

#### === Arduino board pin mapping ===

\*\*\* RC channels (can be changed in APM\_RC\_PIRATES.cpp) \*\*\*

\* PPM SUM signal must be connected to A8 pin

\* Default is TX\_mwi - MultiWii set

A8 - ROLL  
A9 - THROTTLE  
A10 - PITCH  
A11 - YAW  
A12 - AUX1  
A13 - AUX2  
A14 - CAMPITCH  
A15 - CAMROLL

A4-A7 - Copter LEDs. Connect your LED strips to this pins via ULN2003

A0 - Voltage sensor pin

A1 - Current sensor pin

\*\*\* Camera stabilization \*\*\*

d44 - Camera stabilisation Roll servo

d45 - Camera stabilisation Pitch servo

d46 - Camera trigger relay (5V - On, 0v - Off)

Serial ports:

Serial0 (RX0,TX0) - USB/Console/Mavlink

Serial1 (RX1,TX1) - **OSD** (Remzibi, E-OSD, FrSky)

Serial2 (RX2,TX2) - **GPS**

Serial3 (RX3,TX3) - Telemetry (3DR, Xbee, **Bluetooth**)

\*\*\* Sonar \*\*\*

d9 - Sonar Tx  
d10 - Sonar Echo

\*\*\* Status LEDs \*\*\*

d13 - RED  
d30 - YELLOW  
d31 - GREEN

\*\*\* Motor mapping \*\*\*

motor mapping (maximize your text viewer or disable word wrapping to avoid line breaks)

```

=====
Pin   D2  D3  D5  D6  D7  D8  D11 D12 - Arduino pins
CH    3  4  1  2  7  8  10  11 - MegaPirate output channel
MOT_* 3  4  1  2  5  6  7  8  - Arducopter motor mixer mapping
TIMER 3  3  3  4  4  4  1  1  - ATMEGA Timer used to generate PWM signal
=====
TRI   LC  BC  RC  -  S  -  -  -
QuadX LFW RBW RFC LBC - - - -
Quad+ FW  BW  RC  LC  -  -  -  -
Hexa+ BLW FRC FW  BC  FLC BRW -  -
HexaX FLW BRC RW  LC  FRC BLW -  -
Y6    LDC BDW RDC LUW RUW BUC -  -
Octo+ FRC BRC FW  BW  FLC BLC LW  RW
OctoX RFC BRC FRW BLW FLC LBC LFW RBW
OctoV BLC BBRC FLW BRW FFLC FRC FFRW BBLW
Quad8X BLUC BRUW FRUC FLUW FLDC FRDW BRDC BLDW
Quad8+ BUC RUW FUC LUW LDC FDW RDC BDW <<< Not verified
=====

```

**Motors description:**

- B- back
- R- right
- L- left
- F- front
- U- upper
- D- lower
- W- clockwise rotation
- C- counter clockwise rotation (normal propeller)
- S- servo (for tri)

Example: FLDW - front-left lower motor with clockwise rotation (Y6 or Y4)

MegaPirateNG 2.9 R6 è ancora in dev

Inizialmente c'è qualche problema con la nuova procedura di calibrazione degli accelerometri perchè sulla scheda richiedevano un offset sull'asse Z maggiore di 3.0 e quindi la procedura falliva.

Ora c'è una modifica del codice per ampliare i limiti di offset.

Edit:

your place to play with is

**AP\_InertialSensor.cpp**

line 421

```

if( accel_offsets.is_nan() || fabs(accel_offsets.x) > 2.0 || fabs(accel_offsets.y) > 2.0 ||
fabs(accel_offsets.z) > 3.0 )

```

Try to change to >5.0 or what you find in the failed calibration data.

### Per calibrare usare una versione di MP dalla 1.2.30 in poi

Il volo in Stabilize è abbastanza simile a quello della 2.8

L'Alt Hold è ora fantastico. L'integrazione con gli accelerometri garantisce una tenuta di quota simile a quella del NAZA e, come il NAZA, se metti il quad in loiter e lo spingi verso il basso senti la reazione immediata dei motori che contrastano la spinta mentre se lo spingi con forza verso l'alto i motori quasi si fermano... Unica accortezza è che **occorre tenere il gas bene a metà** altrimenti il quad guadagna o perde **quota perchè interpreta la posizione dello stick non centrale come un comando cambio di quota**.

Il volo autonomo in **RTL** è molto più dolce. Le salite e le discese sono belle lente (**si settano con un parametro**) e il quad non traballa più quando scende. Anche la velocità di rotazione è più lenta e meno scattosa.

Fantastico l'autoland. Mettendo a **0 l'altezza finale di RTL** e attivando RTL si può mettere il gas a 0 (in RTL durante il volo lui non lo sente) e aspettare che il quad torni. Quando è sulla verticale della home si ferma un attimo e poi comincia a scendere molto lentamente, atterra dolcemente, spegne i motori e disarma la scheda....tutto da solo.

Unico problema che è il sonar. Attivando il sonar il quad saltella su è giù come un matto e non si riesce a farlo funzionare (con la 2.8 funzionava). Disabilitando il sonar tutto OK.

**Con la 2.9R6 è cambiata la gestione della quota.** Quando metti in loiter il gas devi subito metterlo a centro corsa altrimenti se lasci il gas un po' più alto mpng lo interpreta come comando a salire e il modello lentamente sale. Se lo metti al centro si ferma. Se lo abbassi dal centro scende lentamente (c'è un parametro per regolare il rateo di salita/discesa).

## MOST COMMON MISSION PLANNING EVENTS (MPNG2.9 R6)

---

### WAYPOINT (NAV\_WAYPOINT)

Will cause the craft to move to the position specified by Lat and Lon parameters at the speed defined in WP\_SPEED\_MAX parameter.

Delay option is triggered after the craft reaches the waypoint radius (defined in Mission Planner).

The Next command is loaded after the delay.

Delay time is in seconds, the default is 0

Option Alt Lat Lon

Delay (seconds) Altitude (meters) Latitude Longitude

---

### LOITER\_UNLIM

Option Alt Lat Lon

N/A N/A N/A N/A

Will cause the craft to loiter unlimited at the current location when the command is invoked.

---

### LOITER\_TIME

Will cause the craft to begin to Loiter at the current location when the command is invoked.

Optionally give a location to loiter at that wp – The clock starts ticking after the WP is reached.

Option Alt Lat Lon

Time in seconds Altitude (in meters) Latitude Longitude

---

### LOITER\_TURNS

Will cause the craft to begin to orbit the current location when the command is invoked. Be careful as copter will not simply "turn" at the current location but it will "orbit" around it.

Option Alt Lat Lon

Number of turns (N x 360°) N/A N/A N/A

---

## RETURN\_TO\_LAUNCH

Will cause the craft to return to the home position set when the craft is armed (requires GPS Lock!)  
If Altitude is 0, craft will return to home at the altitude specified by the Mission Planner  
See also RTL\_ALT parameter  
Option Alt Lat Lon  
N/A Altitude (meters) N/A N/A

---

## LAND

Will cause the craft to land at the current location when the command is invoked at speed defined in LAND\_SPEED parameter

---

## TAKEOFF

Will cause the craft to take off and hold position until the altitude is reached.  
Option Alt Lat Lon  
N/A Altitude (meters) N/A N/A

---

## DO\_CHANGE\_SPEED

Will cause the craft to change his speed (previously defined in WP\_SPEED\_MAX parameter)  
Option Alt Lat Lon  
Speed (in meters/s) N/A N/A N/A

---

## DO\_SET\_ROI

Yaw tracking option  
Setting will persist until reboot  
The location is optional  
The option is a number from 0 to 4  
MAV\_ROI\_NONE = 0: Yaw will hold it's current angle  
MAV\_ROI\_WPINDEX = 1: Yaw will point at next WP  
MAV\_ROI\_WPINDEX = 2: Yaw will point at the desired WP at the index #  
MAV\_ROI\_LOCATION = 3: Yaw will point at the indicated location (The location in the command)  
MAV\_ROI\_TARGET = 4: Not implemented  
This event was working in 2.8 R2 but it is not working in 2.9 R6

---

## MOST COMMON MAV LINK PARAMETERS (MPNG 2.9 R6)

---

### RTL\_ALT RTL Altitude

The minimum altitude the model will move to before Returning to Launch.  
Set to zero to return at current altitude.  
Range: 0 4000  
Increment: 1  
Units: **Centimeters**  
Suggested setting: *set according to the obstacles to stay above during a RTL - i.e. 1500 (15 meters) for returning abovetrees)*

---

### RTL\_ALT\_FINAL RTL Final Altitude

This is the altitude the vehicle will move to as the final stage of Returning to Launch or after completing a mission.  
Set to zero to land.  
Range: -1 1000  
Increment: 1  
Units: Centimeters  
**Suggested setting:** *0 for autolanding or 300 for loiter at 3 meters. I do not suggest setting 0 but it is better to use LAND event.*

---

**WP\_SPEED\_MAX** Waypoint Max Speed Target



Defines the speed which the aircraft will attempt to maintain during a WP mission or during RTL.  
Increment: 100  
Units: Centimeters/Second  
Suggested setting: 500 (default) is pretty fast; *it is good for long missions - for short or test missions use 300*

---

**RTL\_LOIT\_TIME** RTL loiter time  
Time (in milliseconds) to loiter above home before beginning final descent  
Range: 0 60000  
Increment: 1000  
Units: ms  
*Suggested setting: 5000 (default) 5 seconds before landing*

---

**LAND\_SPEED** Land speed  
The descent speed for the final stage of landing in cm/s  
Range: 10 200  
Increment: 10  
Units: Centimeters/Second  
*Suggested setting: 50 (default) is OK - lower values can cause copter to land too slow in windy weather*

---

**AUTO\_VELZ\_MIN** Autopilot's max descent speed in cm/s  
Maximum descent speed the autopilot may request in cm/s  
Range: -500 -50  
Increment: 10  
Units: Centimeters/Second  
*Suggested setting: -80 to slow down descending speed (default -125 can be too fast)*

---

**AUTO\_VELZ\_MAX** Auto pilot's max vertical speed in cm/s  
Maximum vertical velocity the autopilot may request in cm/s  
Range: 50 500  
Increment: 10  
Units: Centimeters/Second  
*Suggested setting: 125 (default) is OK*

---

## ROI

E' un evento che si imposta nella missione con Mission Planner o android.

Impostato a 3 fa in modo che il multicoso si muova, da quel momento in avanti, tenendo la prua verso il punto specificato nel ROI. Serve per poter, per esempio, mantenere inquadrato con la telecamera uno specifico target intanto che ci si muove con il multicottero verso waypoints intorno al target medesimo. Funziona molto bene perchè la rotazione è molto lenta e continua, senza scatti.

*DO\_SET\_ROI*

*MAV\_ROI\_NONE = 0: Yaw will hold it's current angle*

*MAV\_ROI\_WPNEXT = 1: Yaw will point at next WP*

*MAV\_ROI\_WPINDEX = 2: Yaw will point at the desired WP at the index #*

*MAV\_ROI\_LOCATION = 3: Yaw will point at the indicated location (The location in the command)*

*MAV\_ROI\_TARGET = 4: Not implemented*

Con la vecchia versione 2.8 il multicoso parte da (WP 1) e va a (WP 2). Quando arriva a (WP 2) si gira verso il (ROI 3) e intando che va a 4, 5 e 6 (tutti WP) tiene la prua sempre direzionata verso il 3 ROI 3).

Con la nuova versione **2.9 R6** l'evento **ROI** purtroppo non funziona più come prima.

Sembra essere un problema di Arducopter 2.9. Su DIYdrones Randy ha consigliato di provare mettendo il parametro **YAW\_OVR\_BEHAVE** a 1. In effetti con il parametro a 1 il ROI funziona nel senso che quando il multicoso arriva al ROI mantiene correttamente la prua verso il ROI ...peccato però che prima del ROI non c'è verso di far puntare la prua verso il prossimo WP. in qualche modo, in attesa che la sistemino, si riesce comunque a fare qualcosa, rinunciando all'orientamento verso il prossimo WP.

Senza, se si mette un evento ROI, il multicoso quando arriva a elaborare l'evento si gira un attimo verso la location specificata nel ROI e poi ritorna a girarsi verso il prossimo WP e se ne va allegramente continuando la missione come se nulla fosse.

Mettendo **YAW\_OVR\_BEHAVE=1** allora il multicoso fino a (WP 2) tiene la prua come cavolo vuole lui (devo capirne ancora la logica) ma appena arriva a (WP 2) si gira miracolosamente verso (ROI 3) e si fa tutto il giretto successivo per 4, 5 e 6 inquadrando sempre (ROI 3) come faceva una volta.