Plane

Plane Home

Tip

The ArduPilot Developer Ecosystem is Evolving! Find out more here …

The free Plane firmware running on a compatible controller board gives any fixed-wing aircraft full autonomous capability.

Plane provides advanced functions such as support for hundreds of three-dimensional waypoints, automatic take-off and landing as well as sophisticated mission planning and camera controls. It works with a variety of Ground Control Station (GCS) software for programming and mission operations and offers a complete UAV solution.

The entire package is designed to be easily approachable for the novice, while remaining open-ended for custom applications, education, and research use.

Note

This is the platform that won the prestigious Outback Challenge UAV competition in 2012, 2014 and 2016 as well as the 2014 Sparkfun Autonomous Vehicle Competition.

System components

- APM autopilot loaded with the latest version of Plane firmware
- Ground Control Station software gives you an easy point-and-click setup/configuration, along with a full-featured ground control interface.
- A suitable airframe for your mission.
- Support tools and hardware for operating and maintaining your aircraft.
- Plus many other useful options, such as two-way telemetry radios which allow in-flight communication and control between the aircraft and your computer.

Fixed-wing aircraft

A fixed-wing aircraft has both advantages and disadvantages in comparison with rotor-craft. Fixed-wing aircraft tend to be more forgiving in the air in the face of both piloting and technical errors, as they have natural gliding capabilities with no power. Fixed-wing aircraft also are able to carry greater payloads for longer distances on less power.

There is a huge variety of fixed wing aircraft from electric battery powered small foam planes to large scale wooden replicas with multi liquid fuel engines and everything in between. You are bound to find a plane that suits your flying style and needs.

When precision missions are required, fixed-wing aircraft are at a disadvantage, as they must have air moving over their wings to generate lift. This means they must stay in forward motion, which means they can’t hover in one spot the way a copter can and as a result cannot provide the same level of precise camera positioning.

So for longer missions and more payload, a fixed-wing is your best choice. But for keeping a camera in one place, consider switching to a copter instead. The same APM autopilot can control that equally well by simply loading the Copter code.

Announcements

- Mar 26, 2016: Plane 3.5.2 released.
- Mar 21, 2016: Plane 3.5.1 released.
- Feb 02, 2016: Plane 3.5.0 released.
- Sept 24, 2015: Plane 3.4.0 released.
- May 20th 2015 - Plane 3.3.0 released
- February 4th 2015 - Plane 3.2.1 released
- November 25th 2014 - Plane 3.2.0 released

Full Table of Contents

User Manual Introduction

This User Manual is designed to work for the novice and experienced users. It should be easy to run through, but there is also a nice path for a leisurely walk.

Please do not be tempted to skip over steps and rush into flight. This is robotics and aviation combined – both of which are quite complex on their own. Take some time to become familiar with this user manual. Follow the steps patiently, and you will have your robot safely up in the air.

- Setup- Installing: hardware, software, and firmware. Plugging in: Sensors, radios, batteries, motors, etc.
- Configuration- Establish telemetry and control between: robot, ground-station, and RC-control transmitter. Set up flight modes, program failsafe behavior, calibrate sensors and motor-controllers, verify correct motor rotation and prop orientation.
- Tuning- Verify performance and behavior, adjusting parameters to suit
- Mission planning and analysis - Programming missions, logging and analyzing telemetry data, ground-control set up.
- Flying- Safe-skillful piloting and rules of the air, weather and location considerations, pre-flight checks, emergency procedures, operation examples

Warning

The topic of safety should be foremost on your mind.

Autonomous robots, flying machines, high-energy power systems driving rotating blades – you are in charge of these potential hazards.
Note

APM is designed to easily integrate with most standard RC aircraft, from simple high-wing trainers to high-speed swept wing fighters and flying wings. If you have something out of the ordinary please assume that a bit more tuning will be required to get things dialed in.
All of us involved with this project care a great deal about the privacy and safety of those whom we share this planet with. Please be a good steward of this technology. It is the product of many evenings and weekends, we make it available for benevolent use.

Video Guides

This page is for video guides on using Plane. *Please add any useful guides that you create (or discover)*:

Basics of Radio Control Aircraft (Mar 24, 2011)

3DR Aero-M Video Tutorials:

Setup for Plane

Just want to get up and running as fast as possible? Here's a guide to the simplest approach. There's a lot more Plane can do, but this will give you a taste:

Assembly Instructions for Plane

This section contains the wiring instructions for assembling the flight controller and other *essential components* of Plane (for several flight controllers):

The instructions for adding other hardware are covered in Optional Hardware.

Plane Configuration

This section contains topics related to configuring and testing Plane components, including those required for the operation of the autopilot.

Note

In addition to the configuration discussed in this section, you may also choose to Configure Optional Hardware including battery monitor, sonar, airspeed sensor, optical flow, OSD, camera gimbal, antenna tracker etc.

Mandatory Hardware Configuration

This topic explains how to configure the components that are required for correct operation of the autopilot.

The linked articles describe the process for configuring the RC transmitter/receiver, compass, and accelerometer and failsafes using *Mission Planner*.

Plane Failsafe Function¶
Plane has a limited failsafe function which is designed to do three things:

1. Detect complete loss of RC signal (if the RC receiver is able to generate a predictable signal-loss

behavior) and initiate a defined auto-mode response, such as returning to home. Some RC equipment can do this, and some can't (see below for details on how to use it if yours supports this function).
2. Detect loss of telemetry for more than FS_LONG_TIMEOUT sec and switch to return to launch (RTL) mode (GCS Failsafe).
3. Detect loss of GPS for more than 20 seconds and switch into Dead Reckoning mode until GPS signal is regained.

Here's what the failsafe **will not do**:

1. Detect if one more more individual RC channel has failed or become disconnected
2. Detect if you're flying too far away or are about to hit the ground
3. Detect autopilot hardware failures, such as low-power brownouts or in-air reboots
4. Detect if the Plane software is not operating correctly
5. Detect other problems with the aircraft, such as motor failures or low battery situations (although the latter can be set up through the main code if you have the right voltage/current sensor)
6. Otherwise stop you from making setup or flight mistakes

Plane Failsafe Documentation¶

Throttle Failsafe¶

How it works. Your RC transmitter outputs a PWM signal that is captured by your receiver and relayed to the autopilot. 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. The autopilot monitors your throttle channel and if it 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 autopilot control range between -100% and 100%. Meaning that when flying, our throttle values will range between 1100 - 1900.

- If we lose RC communication, the receiver if set up properly, will drop to the lowest known throttle value of ~900. This value falls bellow the THR_FS_VALUE and will trigger the autopilot to go into failsafe mode.
- First the autopilot will go into short failsafe (FS_SHORT_ACTN, 0=Disabled, 1=Enabled) when it detects loss of signal for more than FS_SHORT_TIMEOUT sec. The default setting for short failsafe is Circle mode.
- If the RC signal is regained during the short failsafe, the flight will return to auto mode.
- If the loss of signal is longer than FS_LONG_TIMEOUT sec the autopilot will go into long failsafe (FS_LONG_ACTN, 0=Disabled, 1=Enabled).
- The default setting for long failsafe is RTL (Return to Launch).
- Once the long failsafe (RTL mode) has been entered at the conclusion of the short failsafe the RTL mode will continue even if your RC signal is reacquired.

```
       Ext. Range       Normal Range       Ext. Range
```

```
   |----------------|----------------|----------------|
 -150%           -100%             100%             150%

   |_______________|
           |
        Failsafe
```

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 output 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 autopilot 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 autopilot.
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 S_SHORT_TIMEOUT sec the flight mode should switch to Circle. After FS_LONG_TIMEOUT sec the flight mode should switch to RTL. If you observe this behavior, your failsafe function has been set up correctly.

Transmitter Tutorials:

Spektrum Setup

GCS Failsafe¶

How it works. When flying while using telemetry on the GCS, the autopilot can be programmed to trigger into failsafe mode if it loses telemetry. In the event that the autopilot stops receiving MAVlink (telemetry protocol) heartbeat messages for more than FS_LONG_TIMEOUT sec, the GCS failsafe (FS_GCS_ENABL, 0=Disabled, 1=Enabled) will trigger the autopilot to go into long failsafe and change the flight mode to RTL.

Setup.

1. Set FS_GCS_ENABL to 1 to enable it.
2. Connect to the Mission Planner via telemetry. Verify on the bottom right corner of the HUD that you are "flying" in a non auto mode (Manual, Stabilize, FBW are ok).
3. Unplug one of the telemetry radios. After a few minutes power off your autopilot. (Remember the autopilot will not go into failsafe until FS_LONG_TIMEOUT seconds of MAVlink inactivity have passed).
4. Connect your autopilot to the mission planner and pull the logs. Verify on the log that the autopilot

went into RTL after FS_LONG_TIMEOUT sec of MAVlink inactivity.

Failsafe Parameters and their meanings¶

Short failsafe action (Plane:FS_SHORT_ACTN)¶

The action to take on a short (FS_SHORT_TIMEOUT seconds) failsafe event in AUTO, GUIDED or LOITER modes. A short failsafe event in stabilization modes will always cause a change to CIRCLE mode. In AUTO mode you can choose whether it will RTL (ReturnToLaunch) or continue with the mission. If FS_SHORT_ACTN is 0 then it will continue with the mission, if it is 1 then it will enter CIRCLE mode, and then enter RTL if the failsafe condition persists for FS_LONG_TIMEOUT seconds.

VALUE	MEANING
0	Continue
1	Circle/ReturnToLaunch

Long failsafe action (Plane:FS_LONG_ACTN)¶

The action to take on a long (FS_LONG_TIMEOUT second) failsafe event in AUTO, GUIDED or LOITER modes. A long failsafe event in stabilization modes will always cause an RTL (ReturnToLaunch). In AUTO modes you can choose whether it will RTL or continue with the mission. If FS_LONG_ACTN is 0 then it will continue with the mission, if it is 1 then it will enter RTL mode. Note that if FS_SHORT_ACTN is 1, then the aircraft will enter CIRCLE mode after FS_SHORT_TIMEOUT seconds of failsafe, and will always enter RTL after FS_LONG_TIMEOUT seconds of failsafe, regardless of the FS_LONG_ACTN setting.

VALUE	MEANING
0	Continue
1	ReturnToLaunch

Failsafe battery voltage (Plane:FS_BATT_VOLTAGE)¶

Battery voltage to trigger failsafe. Set to 0 to disable battery voltage failsafe. If the battery voltage drops below this voltage then the plane will RTL

- Units: Volts

Failsafe battery milliAmpHours (Plane:FS_BATT_MAH)¶

Battery capacity remaining to trigger failsafe. Set to 0 to disable battery remaining failsafe. If the battery remaining drops below this level then the plane will RTL

- Units: mAh

GCS failsafe enable (Plane:FS_GCS_ENABL)¶

Enable ground control station telemetry failsafe. Failsafe will trigger after FS_SHORT_TIMEOUT and / or FS_LONG_TIMEOUT seconds of no MAVLink heartbeat messages. WARNING: Enabling this option opens up the possibility of your plane going into failsafe mode and running the motor on the ground it it loses contact with your ground station. If this option is enabled on an electric plane then either use a separate motor arming switch or remove the propeller in any ground testing.

VALUE	MEANING
0	Disabled
1	Enabled

Configuration Values for Common Airframes

This page provides some approximate configuration values for common airframes.

Note

These values are good enough to “get you off the ground”. For optimum performance you will still need to tune these parameters for your aircraft.

Configuration values - not complete files¶

We do not not provide complete configuration files because we do not want to encourage sharing complete configuration files between APM users.

Sharing configuration files is often a bad idea because the correct settings for a specific plane are dependent on a lot more factors than just what base airframe type is being used:

- the right tuning values depends on what type of servos you have installed (different servos respond at different speeds), and which hole in the servo horns you connect the control arms to
- the right values also depend on your RC transmitter setup, especially the range of movement of each control axis
- the right tuning values also depend on what motor and battery you have installed, and depends a lot on the total takeoff weight of your aircraft. If one user has a camera installed and another user doesn’t then they will probably need different tuning values.
- manufacturers often make changes to airframes, servos and motors without re-branding the airframe as a new model. Sometimes they even change the direction that servos move
- different users have different optional sensors, which changes what configuration options are needed. For example, some users install an airspeed sensor or an external compass.
- new APM firmware releases frequently change the defaults for some critical parameters to avoid problems found by users. If you load a config file from a user of an older firmware then that will revert those changes

For all of these reasons you should not just download a configuration file from another user and expect it to work.

So what can you do?¶

If another user has tuned a similar aircraft, then you can look in their configuration file to find the key tuning parameter they used, and manually put those values into your APM as a starting point for tuning. The key parameters you should look for are:

- RLL2SRV_P
- PTCH2SRV_P

- NAVL1_PERIOD

Setting those to an approximately correct value will give you a reasonable starting point for tuning your airframe. After that you should follow the tuning guide.

Also note that the Plane parameters default to zero 'I' values. The reason for this is that you need to tune the P value first, then tune the I value. If you put in a non-zero I value when the P value is a long way off the aircraft may oscillate badly and could crash.

Once you have good P values then you should raise the I values a bit, as described in the tuning guide. That will allow APM to cope better with wind. For example, once the P values are setup correctly then setting the following will help with most aircraft:

- RLL2SRV_I: 0.05
- PTCH2SRV_I: 0.05

HobbyKing Bixler v1 or v2¶

Bixler v2 has a slightly bigger wing and sturdier fuselage.

HobbyKing Bixler

- Bixler 1: In ARF form from HobbyKing's USA warehouse or rest of world
- Bixler 2 kit: USA Warehouse and rest of world

Key parameters¶

- RLL2SRV_P: 0.9
- PTCH2SRV_P: 1.0
- NAVL1_PERIOD: 18

Bixler Tips¶

See this posting for how to fit an APM to a Bixler2. Also see High Quality Bixler 1.1 Build (APM2.x).

1. Heavy Bixler’s fly much faster and are prone to tip stalling if slowed up too much, so keep things as light as possible! A stall in FBWA can be tricky to get out of. Autopilots do not handle stalls well.
2. Set manual level with a little bit of positive pitch - take your time to set this, fly and repeat as many times as it takes to get a super stable cruising speed with the lowest throttle setting possible. It takes a little while to work out what pitch and throttle combination give the most stable cruising flight. Set closer to 50% throttle for cruise (but obviously depends on your set up).
3. Perform some test flights in FBWA mode and figure out minimum throttle setting /speed that is still stable. You can “feel” a tip stall coming with enough practice. To recover from the stall, change quickly to Manual mode and then: wings level and pull out slowly. Test with enough height to see how fast the plane needs to be going in each mode. Also be sure to always add sufficient throttle when heading downwind to keep airspeed up.
4. If you prefer to keep the airframe flat you can, in FBWA mode, use only rudder for turning. APM does a great job of keeping the the plane level in turns (with proper speed of course).
5. Make sure that target throttle in full autopilot mode is 10% above throttle settings that were figured out using methods above for mission cruising.
6. If setting a target speed in auto make sure it’s fast enough for the plane and its payload.

Skywalker¶

Skywalker

Available in foam or fibreglass. Current versions have longer wings (1.9m instead of 1.68m).

The skywalker has relatively small ailerons for the size of the aircraft, so you will probably need quite high gains. The following parameters will be a good starting point for most users:

- RLL2SRV_P: 1.8
- PTCH2SRV_P: 1.5
- NAVL1_PERIOD: 20

HobbyKing Skyfun¶

Skyfun

Available here

The SkyFun and FunJet are similar, and are quite “twitchy” planes. A small amount of elevon movement goes a long way! The following values should be a good starting point:

- RLL2SRV_P: 0.4
- PTCH2SRV_P: 0.45
- NAVL1_PERIOD: 17

Multiplex EasyStar¶

Easystar

Available here

The easystar is unusual in not having ailerons. If you are buying a new plane it would be better to get a bixler2, which is a similar size and cost, but flies a lot better.

If you do want to fly an easystar then you will need to set a KFF_RDDRMIX, along with the other key parameters. For example:

- RLL2SRV_P: 0.55
- PTCH2SRV_P: 0.6
- KFF_RDDRMIX: 0.5
- NAVL1_PERIOD: 20

Borojet Maja¶

Borojet Maja

Available here.

The following parameters should be a good starting point:

- RLL2SRV_P: 1.4
- PTCH2SRV_P: 1.3
- NAVL1_PERIOD: 16

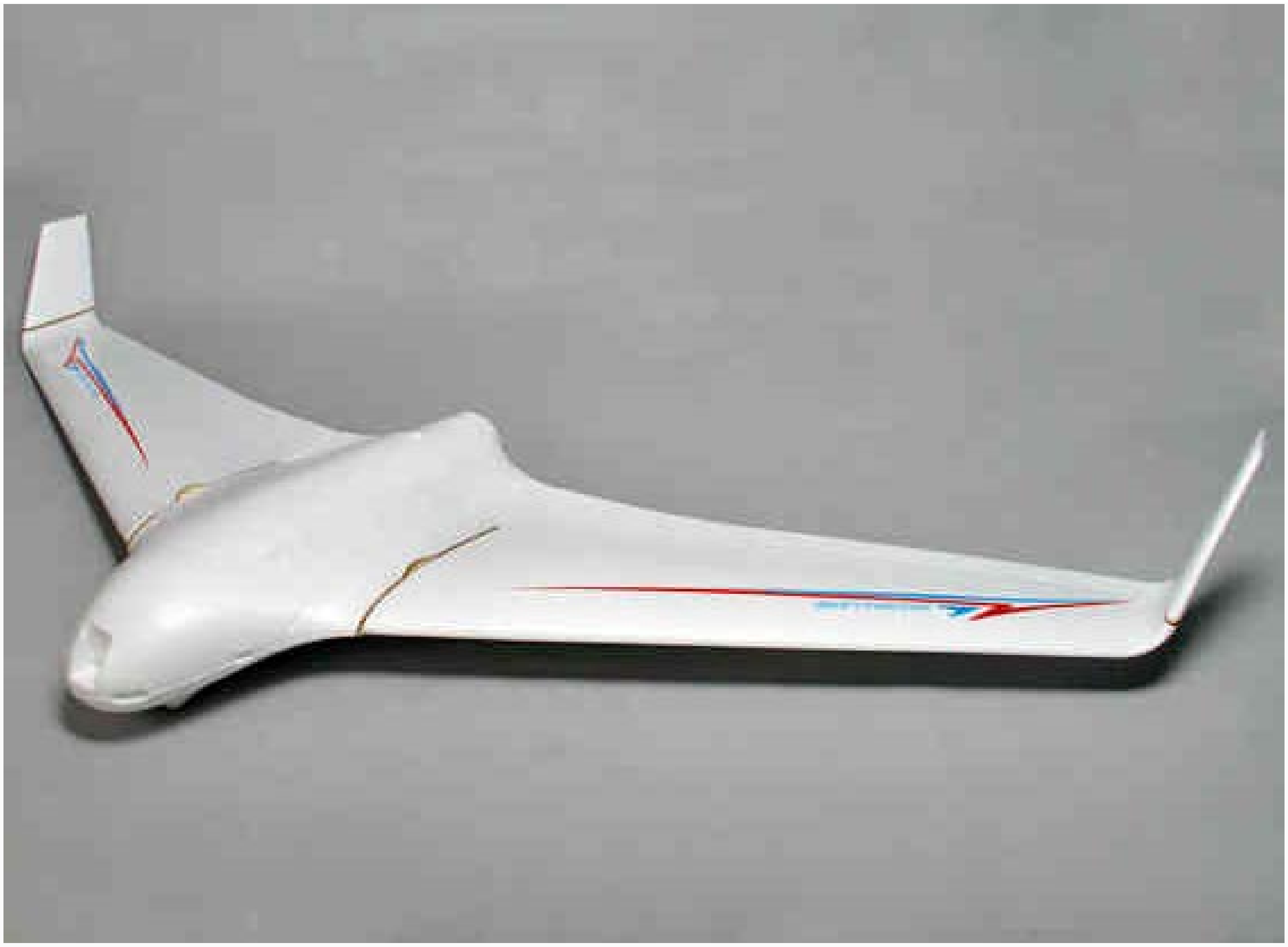
TELINK Toro900 flying wing¶

Available here

The following values should be a good starting point:

- RLL2SRV_P: 1.2
- PTCH2SRV_P: 2
- NAVL1_PERIOD: 18

SkyWalker X8¶



The X8 varies a lot in how people set it up, but you usually need parameters around this range:

- RLL2SRV_P: 0.8
- PTCH2SRV_P: 0.5
- NAVL1_PERIOD: 19

Phoenix Tiger60¶

The Tiger60 is usually flown as a nitro plane, plus it has relatively small ailerons and elevator surfaces. A good starting point will be the following parameters:

- RLL2SRV_P: 1.2
- PTCH2SRV_P: 1.2
- NAVL1_PERIOD: 15
- THR_PASS_STAB: 1
- THR_SUPP_MAN: 1
- THR_SLEWRATE: 30

HotDog¶

The HotDog is a fun sports plane that flies like it is “on rails”. The one in the picture has an OS25 Nitro engine. It flies well with the following parameters:

- RLL2SRV_P: 0.35
- PTCH2SRV_P: 0.6
- NAVL1_PERIOD: 14
- THR_PASS_STAB: 1
- THR_SUPP_MAN: 1
- THR_SLEWRATE: 30

Boomerang 60¶

The Boomerang is a classic high wing trainer aircraft, and like the Tiger60 has relatively small control surfaces. The following parameters will be a good starting point:

- RLL2SRV_P: 1.1
- PTCH2SRV_P: 1.2
- NAVL1_PERIOD: 20
- THR_PASS_STAB: 1
- THR_SUPP_MAN: 1
- THR_SLEWRATE: 30

Mugin¶

The Mugin is a large, fast aircraft. Make sure you have a long enough runway for landing!

The following parameters should be a good starting point:

- RLL2SRV_P: 1.0
- PTCH2SRV_P: 1.3
- NAVL1_PERIOD: 19
- THR_PASS_STAB: 1
- THR_SUPP_MAN: 1
- THR_SLEWRATE: 30

PA Addiction¶

The Precision Aerobatics AddictionX is a fun 3D aircraft. It flies quite slowly, but can do extremely rapid rolls and loops due to its huge control surfaces. The APM flies it fine with the right parameters. The following parameters will be a good start:

- RLL2SRV_P: 0.35
- PTCH2SRV_P: 0.6
- PTCH2SRV_D: 0.04
- NAVL1_PERIOD: 13

RipMax AcroWot¶

The AcroWot is an intermediate nitro sports plane, and a lot of fun to fly! With an OS55AX motor it flies well with the following parameters:

- RLL2SRV_P 1.0
- PTCH2SRV_P: 0.9
- NAVL1_PERIOD: 13
- TRIM_THROTTLE: 35

TechPod¶

The TechPod is a long endurance electric glider, ideal for longer distance photography.

It flies well with the following parameters:

- RLL2SRV_P: 1.5
- PTCH2SRV_P: 1.5
- NAVL1_PERIOD: 17
- ARSPD_FBW_MIN: 9
- ARSPD_FBW_MAX: 20
- TRIM_AIRSPEED_CM: 1200

For a more complete guide see this review.

Normal/Elevon/VTail Mode & Reversing Servos

Every aircraft is different, and as people familiar with RC know, you've got to tell your RC equipment which way the servos go and how that moves the control surfaces, a process that involves "reversing" channels if needed. The same goes for an autopilot. This section will walk you through this process.

Overview¶

For a traditional aileron+elevator+rudder aircraft you need to follow the procedure outlined below to ensure the control surfaces move in the right direction. Note that you must get the channel reversals right both in your transmitter and in the ArduPilot parameters.

First, make sure your control surfaces are going the right way in Manual Mode. In this mode, the RC controls are sent straight through to the servos (possibly with elevon or vtail mixing). If any control surface is going in the wrong direction when you move your RC sticks, use your transmitter's channel reverse function to reverse it.

Second, switch into FBWA mode and check that when the aircraft is rolled or pitched that the control surfaces move the right way to correct the attitude error. This means you need to check that

- When you roll the aircraft to the right that the right aileron goes down and the left aileron goes up.
- When you roll the aircraft to the left that the right aileron goes up and the left aileron goes down.
- When you pitch the aircraft up that the elevator goes down
- When you pitch the aircraft down that the elevator goes up

If any of these are incorrect then you should change the reversal for that channel. See the table below. Also note that if you have changed your RCMAP settings then the reversals will be on different channels. See RCMAP Input Channel Mapping.

If you are using MissionPlanner or a similar modern ground station than you can conveniently see the channel reversal setup and change the settings from one screen

Standard (non-elevon) reversal setup¶

For non-elevon setups (ie. setups with ELEVON_MIXING set to 0 and ELEVON_OUTPUT set to 0), you have 4 parameters that control the servo reversals, one for each channel you can reverse.

The 4 parameters are:

RC1_REV	aileron reversal	set to -1 for reversal	defaults to 1 (meaning no reversal)
RC2_REV	elevator reversal	set to -1 for reversal	defaults to 1 (meaning no reversal)
RC3_REV	throttle reversal	set to -1 for reversal	(for some gas planes)
RC4_REV	rudder reversal	set to -1 for reversal	defaults to 1 (meaning no reversal)

Getting the rudder direction right¶

A very common mistake is to have the rudder reversal set incorrectly. If you have it set incorrectly then your plane can fly very badly, and may not be able to navigate at all.

To check your rudder reversal you need to do the following:

Ensure the KFF_RDDRMIX parameter is set to a non-zero value. For this test you should set it to a high value (such as 0.8) to ensure the rudder movement is large. Remember to reset it back to a lower value afterwards.

You also need to disable ground steering if you have it enabled. Check the GROUND_STEER_ALT parameter. You can re-enable ground steering after getting the rudder direction right.

- put the plane into FBWA mode
- with no stick input (hands off the transmitter) roll the aircraft to the right. The rudder should turn towards the left as it tries to correct the roll.
- Now roll the aircraft to the left. The rudder should turn to the right as it tries to correct the roll.

If the rudder moves in the wrong direction you should change the RC4_REV parameter. A value of 1 means no reversal. A value of -1 means to reverse the rudder.

New style Elevon mixing setup (ELEVON_OUTPUT option)¶

As of Plane 2.73 there is a new ELEVON_OUTPUT option. This option allows you to setup your transmitter for normal aileron/elevator control and for Plane to add elevon mixing on the output of channels 1 and 2. Using ELEVON_OUTPUT has a big advantage over ELEVON_MIXING that your inputs won’t saturate the roll/pitch control in FBWA mode, which means you can get better control of your plane.

Using this setup method it doesn’t matter which channel you have plugged into which aileron.

Note that you cannot use the ELEVON_OUTPUT option on an APM1 board if you have the flight mode channel (FLTMODE_CH) set to 8, as on the APM1 this will lead to hardware pass through of controls when in manual, which means your transmitter must be setup for elevon mixing. If using an APM2 or PX4 this is not a problem, and the ELEVON_OUTPUT option is recommended.

The ELEVON_OUTPUT option is designed to operate exactly as a hardware elevon mixer would operate. To set it up follow these steps:

- Setup your transmitter with no elevon mixing
- Set both RC1_REV and RC2_REV to 1 and ELEVON_MIXING to 0
- Start by setting ELEVON_OUTPUT to 1. In later steps you may adjust this to 2, 3 or 4.
- Put your APM into FBWA mode
- Roll the plane to the right and observe what happens to the elevons
- If the two elevons move in the same direction, then change ELEVON_OUTPUT to 2, and try again.
- If the elevons move in opposite directions, but the APM is correcting the roll in the wrong direction (the left elevon is going down and the right one is going up) then change RC1_REV to -1
- Next try pitching your plane up. If the elevons move in the wrong direction then change RC2_REV to -1
- Now change to MANUAL mode and adjust your transmitter reversals for channels 1 and 2 to produce the right movement in MANUAL mode.

Please make sure that you do careful ground testing after setting these parameters!

Please also see the note about the MIXING_GAIN parameter below.

Old style elevon mixing setup (ELEVON_MIXING option)¶

Note: We STRONGLY suggest you use the new style mixing above rather then this old style.

For old elevon based setups where you have set ELEVON_MIXING to 1, you have 3 different parameters to setup. They are:

ELEVON_REVERSE	reverse the sense of the elevon mixing	set to 1 to reverse, defaults to 0
ELEVON_CH1_REVERSE	reverse channel 1 elevon	set to 1 to reverse, defaults to 0
ELEVON_CH2_REVERSE	reverse channel 2 elevon	set to 1 to reverse, defaults to 0

To select elevon mode or reverse elevon channels, use the elevon checkboxes at the bottom:

Roll to the right illustrated below.

It takes a little trial-and-error to set up elevons on any particular aircraft, but here are the basic steps:

1. First, set it up in manual mode by setting up elevon mixing on your RC transmitter. It matters which elevon is plugged into which channel! **Shown above, the left wing aileron is plugged into Ch1 and the right wing into Ch2.**
2. Still in manual mode, check to see if you have to reverse any channels on your RC transmitter to ensure the control surfaces move the way they should in both pitch and roll.
3. Now that it's working in manual, connect to your APM board with the Mission Planner. Go through the regular setup process. When calibrating your RC input, **don't just move the elevator and aileron sticks to the normal up down, left right positions. Instead, you must move the stick to the CORNERS** or the calibration will be wrong and the servos will try to move too far. This is because now that you've switched your RC transmitter into elevon mode, the elevator and airelon inputs are added when the stick is in the corner (full left and full up as an example).
4. While still in the MP RC setup screen, switch into FBWA Mode. Move move the plane around to test and watch the control surfaces. When you tip the nose of the plane down, the two elevons should go up and vice versa. Likewise with roll; when you roll the plane, the elevons should move to counteract

that and return the plane to level. You'll probably have to reverse something with the check boxes on that screen for correct motion. Just change one thing at a time!

5. If you just can't seem to find the right combination that works, try swapping your servo cables, so that Right is in Output 1 and Left is Output 2. This is something of a last resort, because you'll have to start the setup from the top of this list again.

Please make sure that you do careful ground testing after setting these parameters. Also remember that your RC transmitter must be set up to do elevon mixing, too!

Note

It is possible to configure differential spoilers with old style elevon mixing, although the feature is not widely used and not well tested. Differential spoilers cannot currently be configured with the new type elevon mixing.

Setting up a VTAIL plane¶

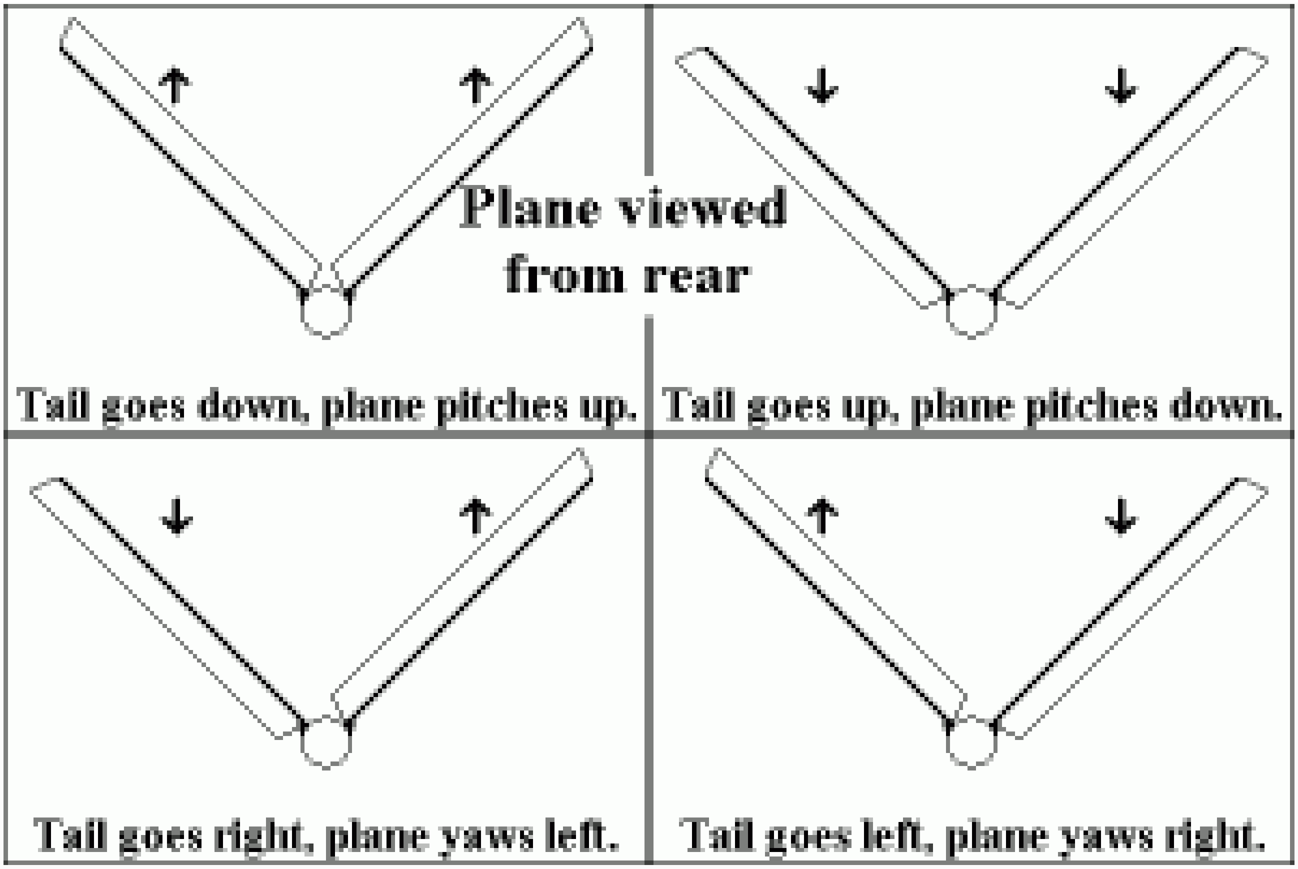
To setup a VTAIL plane, you can enable a software VTAIL mixer using the VTAIL_OUTPUT option. The VTAIL_OUTPUT option works the same way as the ELEVON_OUTPUT option, except that it operates on the elevator and rudder output channels (channels 2 and 4).

Note that you cannot use the VTAIL_OUTPUT option on an APM1 board if you have the flight mode channel (FLTMODE_CH) set to 8, as on the APM1 this will lead to hardware pass through of controls when in manual, which means your transmitter must be setup for vtail mixing. If using an APM2 or PX4 this is not a problem, and the VTAIL_OUTPUT option is recommended for vtail planes. On an APM1 use a hardware vtail mixer instead.

The VTAIL_OUTPUT option is designed to operate exactly as a hardware vtail mixer would operate. To set it up follow these steps:

- Setup your transmitter with no vtail mixing
- Set both RC2_REV and RC4_REV to 1 and KFF_RDDRMIX to 0.5
- Start by setting VTAIL_OUTPUT to 1. In later steps you may adjust this to 2, 3 or 4.
- Put your APM into FBWA mode
- Pitch up the nose of the plane observe what happens to the vtail
- If the two vtail segments move in opposite directions, then change VTAIL_OUTPUT to 2, and try again.
- If the two vtail segments move in the same direction, but the APM is correcting the pitch in the wrong direction (both segments are moving up) then change RC2_REV to -1
- Next try rolling your plane to the right. The two vtail segments should move to try to turn the plane left (to correct for the right roll). If they move in the wrong direction then set RC4_REV to -1
- Now change to MANUAL mode and adjust your transmitter reversals for channels 2 and 4 to produce the right movement in MANUAL mode.
- Finally adjust the KFF_RDDRMIX to a value that gives the right amount of rudder movement for coordinated turns on your plane. This may require some inflight tuning. A initial guess of around 0.5 is likely to work for most planes.

Here's a V-Tail movement diagram courtesy of *Miami Mike*:



Please make sure that you do careful ground testing after setting these parameters!

Please also see the note about the MIXING_GAIN parameter below.

Using MIXING_GAIN to control mixing throws¶

If you use the ELEVON_OUTPUT or VTAIL_OUTPUT options, you may find the MIXING_GAIN parameter useful to control the gain of the mixer.

The default is a gain of 0.5, which ensures that over the full range of the mixer both inputs have authority (it can't saturate). That also means that if you have one input of the mixer (eg. aileron on an elevon plane) at full range, and the other input neutral, then the output is only 1750. That may not be enough roll authority for some planes.

If you change the MIXING_GAIN to 1.0 then you will get the full range of output from a single channel, although if you have full aileron deflection and full elevator at the same time you will saturate the mixer. It will clip output outside of the valid range of 900 to 2100 microseconds.

So if you have found ELEVON_OUTPUT doesn't have enough authority then try raising the MIXING_GAIN.

Important notes¶

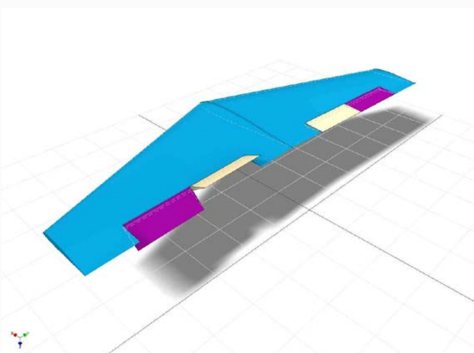
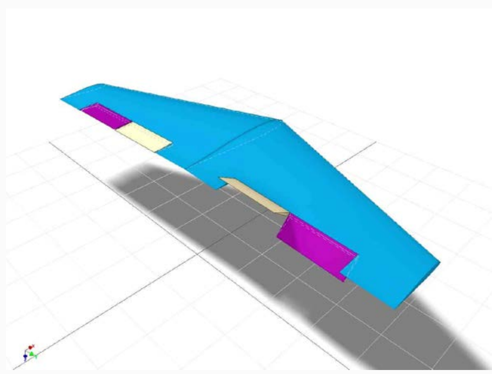
- Whenever you change your firmware your parameter (EEPROM) settings will revert to the defaults if the new firmware has an incompatible parameter (EEPROM) format. The release notes for a release will contain a note if this happens. The developers are careful to try to minimise the number of times this is needed. Please use the APM mission planner or your ground control station to save your

settings, and **carefully check them after any firmware change**.
- make sure you **always do ground tests** before every flight to ensure your channel mixing and reversals are all correct. Be careful to check that not only are your transmitter controls correct, but that the APM responds correctly to attitude changes in the plane when in FBWA mode.

Differential Spoilers

Usage¶

Normal Flying-wing aircraft uses two control surfaces as Elevons to control pitch and roll. In some cases, rudders are added to the winglets to control yaw. Differential spoiler takes advantage of splitting elevons to 4 independent control surfaces: normal elevon functions are reserved for pitch and roll control, but yaw control is done by using two surfaces on one side of the wing to create drag force thus controlling yaw motion. If calibrated correctly, it will ensure pilot has smooth yaw control during take-off and landing as well as compensation during turning (similar to differential aileron).

Left Yaw by inducing drag on left wing:	Right Yaw by inducing drag on right wing:
	

Preparation¶

To use differential spoiler function, the airframe is required to use have 4 control surfaces (2 on each wing) and use elevon configuration. Setup one surface on each side using output channel 1 (roll) and channel 2 (pitch) as elevons:

If you have not done so, please follow the tutorial to setup elevon properly: Archived:APM2.x Wiring QuickStart

Setup¶

Once you have setup two elevon channels successfully, connect the additional 2 surfaces to auxiliary channels (5 to 11 on APM2.X) and set corresponding **RC_FUNCTION**:

Function Name	Function Number	Description
Differential Spoiler 1	16	This should be set to the control surface on the same side of elevon connected to output channel 1 (roll channel)
Differential Spoiler 2	17	This should be set to the control surface on the same side of elevon connected to output channel 2 (pitch channel)

By now, you should have normal elevon function in MANUAL mode. Now switch to FBW mode and use RC controller to test roll, pitch and yaw. Adjust corresponding `RC_REV` until the behavior is correct.

Note

If you have servo direction reversed on either side of the wing, it is most likely that the servo orientations are different or rotation is reversed through programming on some digital servos

Flaperon Configuration

Flaperons use your 2 ailerons (using one channel each) as both flaps and ailerons. This article shows how to set up Flaperons in Plane.

Note

This page was tested on Plane 3.3, but should be relevant to older versions. It was originally created from this blog post.

Input/Output Channels¶

A brief discussion on input and output channels may help. The channels from the transmitter in your hand, to the receiver on the vehicle, and then into the autopilot on the vehicle are your INPUT channels. The channels from the autopilot to your servo's etc are the OUTPUT channels. The input and output channels may not directly map to one another.

Flaperons are the classic example of a setup where input and output channels do not map directly. The autopilot will use the input from the aileron (INPUT channel 1) AND the input from the flap channel (INPUT channel 5 - in the example below) and "mix" them to calculate how the flaperons on the plane should move. The result is sent out to each flap OUTPUT channel (channels 5 and 6 in the example below).

Flaperon setup¶

- Do not do any aileron mixing on your transmitter.
- INPUTS:
  - Leave the standard aileron input on channel 1.
  - You need to add an input channel on your transmitter to control the flaps. You can configure any unused input channel for this however we are going to use RC5. Configure your transmitter to use Channel 5 for flaps (either a switch or a rotary button) and set FLAP_IN_CHANNEL to 5.
  - Move your ailerons to 2 spare output channels on the autopilot that you aren't using. In this example we are using channels 5 and 6.
- OUTPUTS:
  - Set RC5_FUNCTION and RC6_FUNCTION to 24 and 25 (Flaperon 1 and flaperon 2 respectively - which channel is which does not matter).
  - Check that RC5_MIN, RC5_MAX, RC5_TRIM has the correct range set if you haven't used them previously. If your unsure usually 1000, 2000, 1500 will work fine. Do the same for the RC6 equivalents.
- Set FLAPERON_OUTPUT to 1 initially.

- Switch to FBWA or CRUISE. Roll your plane back and forth and make sure the ailerons move in the correct direction (aileron goes down on the wing that you roll down). If they don't, try setting `FLAPERON_OUTPUT` to 4.
- Once this works, try your flaps control on your transmitter and make sure flaps go down and not up. If they go the wrong way, change `RC1_REV` from 1 to -1 (or the other way around) and test again. Once working go back and check `FLAPERON_OUTPUT` in the step above as it may also need to change again.
- Confirm that when in FBWA and your roll the plane the ailerons move in the correct direction, and that your flaps go down.
- Now try the ailerons stick on your transmitter. If they go the wrong way, reverse channel 1 on the transmitter ONLY. Test again. If you put your stick left, the left aileron should go up.

Tuning¶

- Go to failsafe setup in *APM Planner 2* or *Mission Planner*, and make sure the max/min values match `RC5_MIN`/`RC5_MAX` (or adjust them) so that your flaps move all the way (*:ref:`RC1_TRIM`* should also be set to 1500). - Setting the FLAP_SLEWRATE to 100 allows moving flaps from 0 to 100% in one second. Lower this to make your flaps move more slowly.
- Adjust *FLAP_x_PERCNT|SPEED* as desired for auto modes - see *Automatic Flaps <automatic-flaps>*. Note you can ignore the comment on that page saying "parameter for the channel function for the channel you are using for flaps to a value of 3". FLAP_IN_CHANNEL is already set for this. - Have a look at TKOFF_FLAP_PCNT and LAND_FLAP_PERCNT if they are relevant to you.
- When you are flying in manual mode, it can be helpful to setup an elevator down mix on your TX when you set flaps i.e. the more flaps you send, the more elevator down you should send to correct pitch up from flaps. If possible set up the mix value on a rotary switch so that you can control the elevator down correction during a test flight. If you set too much elevator down as a fixed value in your mix, you'll be stuck not being able to use flaps for landing if you put too much elevator down.

Tip

Don't fly until you've rechecked that FBWA/CRUISE moves the ailerons in the right direction and that ailerons also go in the right direction in manual mode.

Crow flaperons¶

If you need Crow flaps (i.e if your ailerons must go up, not down), you can use these instructions and reverse `RC1_REV` so that when you send flaps input, ailerons go up instead of down. Then you should be able to set your flaps channels as FLAP or FLAP_AUTO.

See How would I setup crow flaps? (Fixed Wing FAQ) for more information.

Notes¶

- Manual flaps input is mixed into auto modes. That means if you're landing in manual mode with flaps set to full on your transmitter, and you flip the mode to RTL or some other mode to abort the landing and go back to an auto mode, flaps will stay full. You need to retract them on your transimitter.
- `RCx_MIN` and `RCx_MAX` for Flaperon output channels limit deflection of Flaperons and you can use the TRIM value to move the neutral position in case you want more down travel than up travel.
- `RC1_TRIM` acts as normal aileron trim. `RC1_MIN` and `RC1_MAX` should match the transmitter setting

Automatic Flaps

Plane can control flaps in autonomous modes based on an airspeed schedule. Simply put you can specify two speeds and two flap settings. If your target speed (not your actual speed) is above the specified speeds then your flaps are set to the default (trim) position. If your target speed is lowered below the first flap speed, then flaps are deployed to the first position, and if your target speed is lowered below the second flap speed, then flaps are deployed to the second position.

The target speed can be commanded by changing the value of cruise_speed in the parameter interface, by using the Do_Set_Speed command in a mission, or by the throttle stick position in FBW-B.

Flaps can be configured on any channel (in older versions flaps could only be configured on channel 5, 6, 7 or 8 with 8 not available for manual control).

Software configuration¶

The first step in setting up flaps is to set the parameter for the channel function for the channel you are using for flaps to a value of 3. For example, if you have flaps on channel 5 then set RC5_FUNCTION to 3.

Next set the parameters for your two flap speeds and flap values. These parameters are FLAP_1_PERCNT, FLAP_1_SPEED, FLAP_2_PERCNT, FLAP_2_SPEED.

Redo your radio calibration. Be sure to set the flap switch to the zero flap position before finishing the radio calibration so that zero flaps is stored as the trim (default) position.

That is basically all the setup required for flaps. If you find that your flaps are moving backwards from what you expect, change the reversing parameter for your flap channel - The reversing parameter should have a value of 1 or -1 (Not 0).

Using flaps¶

In Manual, Stabilize, or FBW-A modes your flaps will operate manually and you can set them with your transmitter.

In FBW-B the target airspeed is set by the position of the throttle stick. If set up properly you should see flaps deploy as you lower the throttle stick through the two flap airspeed ranges.

In RTL, Guided, and Auto modes flaps settings are determined by the schedule in your parameters and the current value of cruise_speed. You can change cruise_speed to a value in one of the flap ranges from the GCS and should see the flaps deploy. Also, if you use the Do_Change_Speed command in mission flaps will be deployed when the target speed is changed into one of the flap speed ranges.

Throttle Arming in Plane

A software safety feature that requires the throttle to be explicitly armed by a pilot is available for Plane. It was first introduced in version 2.77 and since 3.3.0 it is enabled by default. This is especially applicable to electric planes. The idea is to prevent an unexpected throttle up when someone’s hand is near a propeller.

Warning

This feature in no way removes the need to respect the prop! When the plane is powered, ALWAYS avoid placing hands in the vicinity of the propellor, even when the throttle is disarmed. If all is not well with the autopilot electronics or software there is always a slight possibility that signal could unintentionally reach the motor. Even though this is unlikely (and made even less likely by safety features such as this) it only takes one time to chew up a finger or hand!
Parameters governing throttle arming are introduced in the ARMING section of the parameters wiki page. This page discusses throttle arming in greater detail and walks through the typical procedures for using this safety feature.

A Safety Note About Arming¶

A safety feature of PX4 / Pixhawk is that all servo output is kept at at minimum values until the safety button is pushed. This has the nice effect of (usually) disabling electric motors and it is quite possible to use only this safety feature for propeller safety. However:

1. This feature is not available for the APM.
2. Generally during pre-flight checks it is desirable to verify servo operation (e.g., to check ailerons are deflecting in the expected direction). This can more safely be done with the throttle disabled. When the button is pushed **all** servos are enabled, including the motor – unless this additional software throttle safety is used.

A note about APM1 and channel 8¶

The APM1 has a hardware multiplexor on channel 8 which forces pass-thru of the first 4 channels if channel 8 is above 1750. This means the arming code is bypassed if you use channel 8 for your flight mode switch. So if you want to use arming on an APM1 then you should change FLTMODE_CH to another channel (say channel 5) and setup your transmitter/receiver to put the mode switch on that other channel.

IMPORTANT: RC Transmitter Calibration¶

It is essential that your RC radio transmitter be calibrated correctly before continuing. Please see the Calibrate your RC input wiki page if you don't know how to calibrate your radio.

Warning

During RC calibration throttle ups are possible, even with safeties enabled and you should therefore remove the propeller or move the plane to a safe area and secure it.
Depending on the ground control software you use, the RC3_MIN value may be continuously changing during calibration. Also, if the RC3_MIN value is inadvertently set incorrectly the plane may throttle up after a bad calibration (e.g., if the low stick setting was not captured).

Note that if you have RCMAP_THROTTLE set to something other than 3, then the RCn_MIN value used will be the one for the channel you have selected as the throttle channel.

When calibrating your RC input you should also be careful to set the minimum value of the throttle (usually RC3_MIN) to the minimum value when in normal flight control. Don't set it to the value used by your transmitter when in throttle failsafe or you won't be able to arm using the rudder as the APM will think you are at a non-zero throttle level.

Simplest Solution: Use Only ARMING_REQUIRE¶

The simplest way to use the throttle arming feature is to require the user to request the throttle to arm. ARMING_REQUIRE has three possible values, 0, 1 and 2 (the default value is 1, enabled). They have the following effect:

- ARMING_REQUIRE=0: No effect. Throttle arming safety is not employed.
- ARMING_REQUIRE=1: Before the user arms throttle, send minimum PWM to the throttle channel (which is usually channel 3). Therefore the RC3_MIN PWM value is sent on the servo output rail.
- ARMING_REQUIRE=2: Before the user arms throttle, send no PWM signal to the throttle channel (usually channel 3). Some ESCs are not happy with this and will continuously beep, some will not. Some users may prefer this setting as it ensures no signal is sent to the ESC when disarmed.

When the ARMING_REQUIRE parameter is set to 1 or 2 (enabled) and ARMING_RUDDER is not to 0 (disabled), all that is required to arm the throttle is to either:

1. Arm the throttle via the ground control software.

   OR

2. Arm the throttle by applying full right rudder input for several seconds.

To arm throttle in Mission Planner, use the Flight Data screen, then select the Actions tab. This provides an “Arm/Disarm” button that can be used to arm and disarm the throttle.

Location of the Arm/Disarm buttonin Mission Planner (button circled in red near the bottom of theimage).

Alternatively, apply full right rudder to your RC transmitter for several seconds to arm the throttle. Whichever method you choose should result in a message from your ground control software stating that throttle arming was successful. Mission planner displays a message in the artificial horizon on the Flight

Data screen:

Message on the artificial horizon inMission Planner stating that arming was successful.

Increased Safety: Perform System Checks Before Arming Throttle¶

By default the ARMING_CHECK parameter is set to 1 so the flight controller performs system health checks before arming throttle when a user attempts to arm. You can enable/disable any check with a bitmask. See wiki documentation on the ARMING_CHECK parameter for more information.

One thing to be aware of if you typically do not fly with a ground control station: **it will be difficult to determine why your autopilot is not arming if you are not connected to a ground control station when arming.** The ARMING_CHECK parameter should probably be left at 0 when at the an airfield without a ground control station.

The following are possible system health messages that may return if ARMING_CHECK is enabled and the autopilot rejects a request to arm the throttle:

- Message: “Hardware Safety Switch.” Solution: push the hardware safety switch on the PX4 or Pixhawk (does not apply to APM).
- Message: “Battery failsafe on.” Solution: Ensure your battery is charged. If it is, ensure your battery failsafe values are set correctly. For more information on failsafes, see the Failsafe Functions wiki page.
- Message: “Radio failsafe on.” Solution: Ensure that the RC transmitter is able to communicate with the RC receiver. For more information on this failsafe, see the documentation on the Throttle Failsafe.
- Message: “Bad GPS Pos.” Solution: Need to get a 3D fix with the GPS receiver. After ensuring your GPS receiver is functioning properly, ensure nothing aboard the plane or in the immediate environment is interfering with GPS satellite signals.
- Message: “No GPS detected. Solution: ensure your GPS receiver is functioning.
- Message: “No compass detected.” Solution: If you do not intend to use a compass, then change the COMPASS_USE parameter to 0. If you do, then you’ll need to ensure your compass is installed and healthy.
- Message: “Compass not calibrated.” Solution: Calibrate compass. In Mission Planner this is accomplished in the Initial Setup screen, menu item Manadtory Hardware > Compass.
- Message: “Compass not healthy.” Solution: Ensure you do not have the compass installed near something that can induce a magnetic field, such as the motor. You also may try re-calibrating the compass.
- Message: “Baro not healthy.” Solution: Try rebooting the autopilot. If this fails you will have to diagnose what the problem is with the barometer.

Disabling Rudder Arming¶

Some pilots will not want to use the rudder arming capability. To only allow throttle arming via the ground control software set the ARMING_RUDDER parameter to 0. The autopilot will no longer arm throttle when full right rudder is applied on the RC transmitter.

Advanced Failsafe Configuration

The core failsafe functionality of Plane is based on RTL (Return to Launch). It is able to initiate a RTL if the aircraft loses contact with the ground station or loses RC control. That is fine for most users, but in some situations more advanced failsafe capabilities are needed. That is what the Advanced Failsafe options are for.

Background¶

The advanced failsafe options for Plane were added for the Outback Challenge competition, and the features are designed to fit the rules of that competition. Whereas normal failsafe is oriented around saving the aircraft in case control is lost by the pilot, the OBC rules are oriented around ensuring safe operation in a defined region of airspace. This means that the advanced failsafe options are designed to deliberately crash the aircraft if there is any risk that it may fly outside of the region of airspace defined by a geographic boundary and a maximum altitude.

While the advanced failsafe features of Plane were designed for the OBC competition, they can be useful in other situations, and are very flexible. They are also harder to configure, so please read this page carefully several times before deciding whether to enable these features on your aircraft.

Mission based¶

The key difference between normal failsafe and the advanced failsafe (AFS) options is that the AFS options are mission based. When a failsafe option (such as loss of GPS lock or loss of ground station communications link) happens, the AFS options specify a waypoint number in the mission that the aircraft will switch to. This allows the pilot to configure a complex series of actions to take when a failsafe event occurs - anything that can be scripted as an Plane mission can be made to happen on failsafe events.

Typically the pilot will setup the AFS options for failsafe conditions so that the aircraft will loiter at its current location for some period of time (say one minute) and then proceed back towards home via a pre-determined flight path. It may also include changes in airspeed, changes in altitude, automatic landing or anything else that can be programmed in a mission. If the failsafe event stops (for example GPS lock is regained) then the aircraft will switch back to the mission item it was previously flying towards and continue the mission.

This makes the AFS options only really appropriate for AUTO missions. If you are primarily flying Plane in CRUISE mode or other modes then you should use the standard failsafe options.

Enabling the AFS failsafe system¶

To enable the AFS failsafe system you need to set the AFS_ENABLE parameter to 1. The default is zero, which means all the other options are disabled.

Note that the AFS system is only built into Plane by default on higher end autopilot boards like the PX4 and Pixhawk. On the APM2 you must recompile the firmware yourself to use the AFS system.

AFS Termination¶

The concept of “flight termination” is key to understanding the AFS failsafe system. Termination is where the aircraft deliberately dives into the ground by setting all control surfaces to maximum and throttle to zero so as to enter a spin.

The AFS system will only start a termination if the AFS_TERM_ACTION is set to the magic value 42. For any other value the AFS system will print a message on the GCS console saying that it wants to terminate, but won’t actually change the control surfaces at all. Using a value other than 42 is useful for test flights where you don’t want the aircraft to terminate on a failsafe event.

Note that if AFS_TERM_ACTION is not set to 42 then other normal failsafe code is still active, for example if you have a geofence enabled then the aircraft will fly back to the geofence return point.

When enabled, the AFS termination system also sets up the secondary IO microcontroller on the Pixhawk autopilot to terminate the aircraft if communication is interrupted between the main FMU microcontroller and the IO microcontroller, for example if the flight firmware crashes.

An AFS flight termination is not recoverable. Once your aircraft starts a termination then is no way to recover.

Types of Failsafe Events¶

The AFS failsafe system supports five of types of failsafe events:

- geofence breach
- maximum pressure altitude breach
- GPS loss
- Ground station communications loss
- barometer failure

Each of these types of failures has its own specific handling, which is described below.

Geofence Breach¶

If a geofence is enabled then the AFS failsafe module will monitor the aircraft for a breach of the boundaries of the geofence (and lower and upper geofence altitudes if set). If a breach happens then the AFS system will immediately terminate the flight (see termination above).

Maximum pressure altitude breach¶

When sharing airspace with other aircraft it is usual practice to define the available flight altitudes in terms of a common reference pressure, typically QNH (a reference to “nautical height”). The QNH reference pressure, measured in millibar, is distributed to all aircraft either via a radio message or through aviation internet and weather sites.

Aircraft then use their barometer to measure the pressure relative to that QNH pressure, which gives them an altitude reference which all aircraft in the area should be using.

The AFS failsystem system is able to enforce a pressure altitude limit by setting the QNH pressure in the AFS_QNH_PRESSURE parameter, as a value in millibars. The pilot should then also set a pressure altitude limit using the AFS_AMSL_LIMIT parameter (in meters). Note that this pressure altitude limit is relative to sea level (AMSL stands for “above mean sea level”).

If both of these parameters are set then the AFS system fill monitor pressure altitude and will initiate a termination if the pressure altitude rises above the AFS_AMSL_LIMIT.

You need to be very careful to set the right AFS_QNH_PRESSURE for your local conditions on the day of your flight, as the QNH pressure can be very different on different days.

In addition to the QNH pressure limit, the AFS system also monitors the health of your barometer. If the barometer is unhealthy for 5 seconds then the AFS system will check the AFS_AMSL_ERR_GPS parameter. If it is -1 (the default) then the aircraft will terminate immediately. If it is not -1 then the AFS system will use the AFS_AMSL_ERR_GPS value as a margin to add to the GPS height, and will allow the flight to continue if the GPS altitude plus the AFS_AMSL_ERR_GPS value (in meters) is below the AFS_AMSL_LIMIT value. The purpose of this margin is to account for the inaccuracy of GPS altitudes. A value of 200 is reasonable for safety to ensure the AFS_AMSL_LIMIT pressure altitude is not breached.

GPS Loss¶

The AFS system monitors the health of your GPS receivers throughout the flight. If all of your available GPS receivers lose position lock then this initiates a GPS failure failsafe.

When a GPS failure occurs (which is defines as loss of GPS lock for 3 seconds) the AFS system will look at the AFS_WP_GPS_LOSS parameter. This parameter species a waypoint number in your mission to use when a GPS failure occurs. If AFS_WP_GPS_LOSS is non-zero the aircraft will change current waypoint to the waypoint number specified in AFS_WP_GPS_LOSS. You should setup your mission so that the aircraft will perform whatever actions you want on GPS loss. For example, you could have a set of waypoints starting at number 10 which first loiter on the spot for 30 seconds, and then proceed back to the airfield. You would then set AFS_WP_GPS_LOSS to 10 to enable that part of the mission on loss of GPS lock.

When setting up mission items for GPS lock it is sometimes useful to include “loiter at the current location” waypoints. That is achieved by setting both the latitude and longitude of LOITER mission commands to zero.

If the GPS recovers after a GPS failsafe has started then the aircraft will automatically resume its mission where it left off.

If during a period of GPS loss the aircraft also loses communications with the ground station then this is termed a “dual loss”, and the aircraft will terminate.

If AFS_MAX_GPS_LOSS is set to a non-zero number, then it is used as a maximum count of the number of GPS failures that will be allowed while returning to the mission after GPS lock is re-established. This counter is only incremented if the 2nd GPS failure happens at least 30 seconds after the previous one (to account for a short period of GPS failure).

Ground station communications loss¶

The AFS system monitors the health of the link between your ground station and your aircraft. It does this by looking for HEARTBEAT MAVLink messages coming from the ground station.

If the aircraft does not receive a HEARTBEAT message for a period of 10 seconds then it enters a GCS failsafe state. It then looks for a AFS_WP_COMMS parameter, and if that is non-zero it will change the current target waypoint to the one given in AFS_WP_COMMS. You should set up a section of your mission with whatever actions you want to take on loss of communications.

If GPS lock is lost at the same time as GCS communications is lost then that is considered a “dual loss”, and the aircraft will immediately terminate.

Note that the monitoring of HEARTBEAT messages only tells the autopilot that it can see messages from the ground station. It does not mean the ground station can see messages from the aircraft. So it is quite possible for your ground station to be reporting loss of communication while the aircraft is still receiving HEARTBEAT messages.

If AFS_MAX_COM_LOSS is set to a non-zero number, then it is used as a maximum count of the number of communication failures that will be allowed while returning to the mission after communications is re-established. This counter is only incremented if the 2nd comms failure happens at least 30 seconds after the previous one (to account for a short period of communications failure).

RC Loss¶

If RC control is lost a manual control mode for more than AFS_RC_FAIL_MS milliseconds then flight termination is activated. This termination mode is only enabled if AFS_RC_FAIL_MS is non-zero. For the OBC rules it should be set to 1500 (giving 1.5 seconds).

Monitoring the AFS system¶

The AFS system provides some additional parameters to make it easier to monitor the health of the failsafe system using external electronics (such as an external failsafe board).

The key parameters are:

- **AFS_TERM_PIN**: This is a digital pin which is set to a high voltage if termination is started. Note that this pin will go high on termination even if the AFS_TERM_ACTION parameter is not set to 42.
- **AFS_HB_PIN**: This is a digital pin number for a pin which is toggled at a rate of 10Hz by the failsafe system. If termination occurs and a AFS_TERM_PIN value is not set then the heartbeat pin will stop toggling.
- **AFS_MAN_PIN**: This is a digital pin number for a pin which goes high when the aircraft is in MANUAL mode. It may be useful with some external failsafe boards to detect manual mode and behave differently.

Manual Termination¶

Apart from automatic termination it is also important for the aircrafts operator to be able to terminate the aircraft immediately if the think the aircraft is a danger to people or other aircraft. To force an immediate termination you should use the AFS_TERMINATION parameter. By setting that parameter to 1 the aircraft will immediately terminate.

Example AFS failsafe mission¶

Setting up a AFS failsafe mission takes time, and needs to be done very carefully. To help you understand what is possible you may find the following example files useful

- A waypoint mission for the 2014 Outback Challenge with waypoints for different AFS failures commented in the file
- A geofence file for the 2014 Outback Challenge

Testing the AFS system in SITL¶

It is highly recommended that you extensively test the AFS system using the SITL simulation system before using it on a real aircraft. You can simulate all types of in-flight failures using the SIM_ parameters. To start SITL in Kingaroy ready for OBC testing you would use:

```
sim_vehicle.py -L Kingaroy --console --map
```

The key parameters for failsafe testing in SITL are:

- Test GPS failure: param set SIM_GPS_DISABLE 1
- Test RC failure: param set SIM_RC_FAIL 1
- Test comms failure: set heartbeat 0
- Test fence failure: switch to CRUISE mode and fly across boundary
- Test QNH failure: param set AFS_AMSL_LIMIT 100

Additional tips for AFS failsafe users¶

You need to ensure that your geofence is enabled before takeoff. This can either be done as part if your preflight checklist, or you could set it a FENCE_CHANNEL and enable it from within your transmitter, ensuring that if your transmitter is out of range that the fence remains enabled.

Settings for Outback Challenge 2014¶

To be compliant with the OBC 2014 rules you should have the following settings:

- AFS_ENABLE: 1
- AFS_WP_COMMS: waypoint number for OBC comms hold followed by two minute loiter, then return to airfield home
- AFS_WP_GPS_LOSS: waypoint number to loiter in place for 30 seconds, followed by return to airfield home
- AFS_TERM_ACTION: 42
- AFS_AMSL_LIMIT: 914
- AFS_QNH_PRESSURE: correct QNH pressure for the day
- AFS_RC_FAIL_MS: 1500
- AFS_MAX_GPS_LOSS: 2
- AFS_MAX_COM_LOSS: 2

Airframe specific guides

Some airframes have quite specific challenges or requirements that make a airframe specific setup guide worthwhile.

Setup Guide for Parrot Disco

The Parrot Disco is a lightweight delta-wing with built-in camera and good flight characteristics. This page will help you get setup with ArduPilot running on your Disco.

Prebuilt Firmware¶

A build of APM:Plane for Disco is automatically generated by the ArduPilot autobuild system. You can see the builds here:

> http://firmware.ardupilot.org/Plane/

the file you need to download is ‘arduplane’.

Building Firmware¶

To build the ArduPilot firmware yourself you should use the waf build system, which is included as part of ArduPilot. The command to build APM:Plane for Disco is:

- ./waf configure –board disco
- ./waf plane

this will give you a file build/disco/bin/arduplane that needs to be installed on your Disco.

Networking¶

The Disco has a builtin WiFi access point, which will show up on your network as “DISCO-xxxxxx” where xxxxxx is a device specific serial number. The WiFi has no password by default.

You can also use USB networking if your operating system supports it. Just plug a USB cable into the left most USB port on the back of the CHUCK module and connect with your operating systems network manager.

Installation¶

ArduPilot is usually installed in /data/ftp/internal_000/APM. You should use adb to create that directory and put a copy of arduplane from the build step in the directory.

You will need some additional setup and configuration files as well. A copy of all of the required files is available here:

> https://github.com/ArduPilot/ardupilot/tree/master/Tools/Frame_params/Parrot_Disco

You will need to modify the /etc/init.d/rcS_mode_default startup script to start ArduPilot. An example of a modified startup script to start ArduPilot if it is installed is available here:

> https://github.com/ArduPilot/ardupilot/blob/master/Tools/Frame_params/Parrot_Disco/rcS_mode_default

you may find the 'Raw' button useful for downloading that file.

That script will start ArduPilot is there is a start_ardupilot.sh script in the APM directory.

You will also need to install the start_ardupilot.sh script. A copy of a suitable script is here:

> https://github.com/ArduPilot/ardupilot/blob/master/Tools/Frame_params/Parrot_Disco/start_ardupilot.sh

that starts the Disco FAN and then loops running ArduPilot. The loop is so you can restart ArduPilot with a MAVLink preflight-reboot request from your GCS without rebooting the Disco.

The startup script assumes that your Disco appears on the 192.168.42.0/24 network for WiFi and 192.168.43.0/24 network for USB networking. It tells ArduPilot to announce itself as a MAVLink device on those two networks. It will pick up the first MAVLink capable ground station that connects on UDP port 14550 on those networks.

Ground Station¶

For configuring your Disco you will need a ground station software package (GCS). There are several choices for use with Disco:

- MissionPlanner (windows only)
- APM Planner (cross platform)
- QGroundControl (cross platform)
- MAVProxy (cross platform, experts only)

They are all capable of controlling all aspects of the Disco setup and flight, including autonomous missions.

Transmitter Setup¶

The usual way to fly a Disco with ArduPilot is with a SBUS receiver and matching transmitter. You can also fly without a transmitter, but that is only recommanded for advanced users who are very familiar with automated flight control with ArduPilot. We hope to support the Disco WiFi based transmitter in ArduPilot in the future.

For controlling Disco using a SBUS receiver and R/C transmitter you need to setup how the elevon mixing is done. The combination of channel directions on your transmitter must match the parameters setup in the Disco.

The most common issue with R/C setup on the Disco is what reversal you have on your R/C transmitter for elevator (pitch) input. Some transmitters default to pulling back on the pitch stick giving a higher PWM value whereas other transmitter manufacturers use a convention where pulling back on the pitch stick gives a lower PWM value.

With the 3.7.0 and earlier release of APM:Plane you need to change your transmitter to match the expected direction for the Disco parameters. That expected direction is that pulling back on the pitch stick produces a lower PWM output from the transmitter. For that setup you need:

- ELEVON_OUTPUT 2
- RC2_REV -1

For the 3.7.1 release and later there are additional options for ELEVON_OUTPUT which allow for your transmitter to have either reversal. If you have 3.7.1 or later installed and your transmitter produces a higher value on the pitch channel when pulling back on the stick then you need:

- ELEVON_OUTPUT 6
- RC2_REV 1

Warning

The telemetry output option of many R/C receivers can interfere badly with the Disco WiFi. Make sure you do a R/C range check before flying and also check that your WiFi telemetry has sufficient range for your needs. Many R/C receivers with a telemetry option can disable the telemetry when binding. Please refer to your receiver documentation for details. It is recommended that you disable receiver telemetry if you find your WiFi telemetry link from the Disco is affected by your receiver.

Loading Parameters¶

You will need to load a set of parameters suitable for the Disco using your GCS. A recommended set of parameters to start with is here:

> https://github.com/ArduPilot/ardupilot/blob/master/Tools/Frame_params/Parrot_Disco/Parrot_Disco.param

you can download that file directly for loading into a GCS with this link:

> https://raw.githubusercontent.com/ArduPilot/ardupilot/master/Tools/Frame_params/Parrot_Disco/Parrot_Disco.param

Compass Calibration¶

You need to calibrate the compass in your Disco before you fly. This must be done with the hatch in place due to the magnetic catch on the hatch.

Each GCS choice has an option to start a compass calibration. Please choose on-board compass calibration for your GCS and follow the prompts.

Note that you may find you need to raise COMPASS_CAL_FIT to allow successful calibration of the Disco, as the magnetic setup of the Disco hardware is not ideal and won’t produce a perfect fit. We recommend setting COMPASS_CAL_FIT to 20.

Accelerometer Calibration¶

You also need to perform an accelerometer calibration. Please follow the prompts in your GCS for the accelerometer calibration procedure. This will only need to be performed once.

Airspeed Calibration¶

Before each flight you should perform an airspeed offset calibration as the airspeed sensor will vary in its zero value between power cycles.

You should loosely cover the pitot tube that is built into the power switch and choose the pre-flight airspeed calibration option in your GCS.

Stabilisation Check¶

Before each flight you should check the stabilization of the Disco by changing to FBWA mode and checking the following:

- roll the Disco to the right. The right elevon should go down, the left elevon should go up
- roll the Disco to the left. The left elevon should go down, the right elevon should go up
- pitch the nose up. Both elevons should go down
- pitch the nose down. Both elevons should go up

Next you should check for correct transmitter control with the Disco held level.

- input right roll on the transmitter. The left elevon should go down and the right elevon should go up
- input left roll on the transmitter. The right elevon should go down and the left elevon should go up
- pull back on the pitch (elevator) stick on the transmitter. Both elevons should go up.
- push forward on the pitch (elevator) stick on the transmitter. Both elevons should go down.

Takeoff¶

The Disco has a very low stall speed which makes it easy to launch in a wide variety of ways. Some recommended ways are:

- a side launch where you hold a wing close to the fuselage, and launch the aircraft forward. An example is shown here:

- a forward throw launch, as shown here:

Always launch into the wind, and be careful to keep your hand clear of the propeller.

Also note that you can configure Disco for “shake to start”, to start the motor when the airframe senses a shaking motion. That is set by the TKOFF_THR_MINACC=4 parameter in the parameter file linked above.

You can see a “shake to start” example here:

Flight Modes¶

For general stabilised flight FBWA mode is recommended. This is also good for takeoffs and landings.

For longer distance FPV flying CRUISE mode is recommended as it will hold airspeed, height above ground and ground course.

You should also setup your transmitter for easy access to RTL mode to bring the plane home if you need to.

Manual Landing¶

To land manually FBWA mode is recommended. Just point the plane in the direction you want to land and drop the throttle. To come in more steeply push the pitch stick away. To perform a nice flare pull back a small amount on the pitch stick just before touchdown.

APM:Plane will automatically put the nose down a couple of degrees when at zero throttle to keep the airspeed up. You can set how much nose down it uses at zero throttle with the STAB_PITCH_DOWN parameter.

Make sure you land with zero throttle or you risk breaking the propeller.

AUTO Landing¶

When using an AUTO mission you can place a NAV_LAND waypoint where you want to land, with a target altitude of zero meters. You also need to place an approach waypoint about 200 meters before the NAV_LAND point, about 30 meters above the ground. The Disco will automatically flare and cut the motor as it approaches the landing point.

Note that the Sonar used on the Disco for landing flare detection does have a tendency to sometimes produce false positives. That can cause the Disco to flare early in the landing as it thinks it is close to the ground. Because the Disco has such good glide characteristics this doesn’t result in a crash, but it does cause it to land well short of the target position.

Log Files¶

There are two types of log files for the Disco with ArduPilot. The first is a “tlog” which is stored by your GCS software on your ground station. The second is a “DF” log, which is stored on board the Disco.

The log directory is /data/ftp/internal_000/APM/logs and can be accessed by ftp. Just enter a URL like ftp://192.168.42.1/ in Windows Explorer if using Windows to view the storage on the Disco and access log files. These files have a ”.bin” extension (for binary log file).

The C.H.U.C.K Autopilot¶

The heart of the Disco is the C.H.U.C.K autopilot, an orange box which is a general purpose autopilot. It is perfectly possible to use the C.H.U.C.K in a different airframe.

For more information on using C.H.U.C.K with ArduPilot please see the C.H.U.C.K AutoPilot

First Flight

The following topics explain how to set up and configure your Plane for its first flight, and includes an explanation of the main Flight Modes that you will use.

Starting up and calibrating Plane

This article describes the basic setup and calibration that should be performed before launching Plane.

Ground calibration

Set your transmitter mode switch to “Manual”. This is a safe mode in which to start up the system.

When you power on your board at the field, you should leave the plane motionless on the ground until the three colored LEDs stop flashing (about 30 seconds). That means that the gyros have been calibrated and the plane is ready to fly (assuming you also already have GPS lock).

After the ground start completes you should wait for GPS lock before flying. If you do not wait for GPS lock the home location will not be set correctly, and the barometric altimeter calibration will be incorrect. It should take less than two minutes to get lock. If you’re using the MediaTek module, the blue LED on the module will flash while it’s waiting for lock, then turn solid once it has it. Once that happens, the red LED on APM should stop flashing and turn solid. If the blue MediaTek LED turns solid but the red APM LED is still flashing, press the reset button on APM and once it reboots, the red LED should go solid.

Note

On the UBLOX GPS module itself the LED is off while acquiring satellites and on blinking when satellites have been acquired.
The PX4 must have its “Safety” mechanism disengaged before it can be armed.

- **PX4 Safety Button LED Indications:**
  - Fast Blinking indicates: Error Condition, Safety cannot be disengaged. Possibly not calibrated or sensor error.
  - Slow Blinking indicates: Safe condition. Safety can be disengaged by depressing Safety Button for 5 seconds.
  - LED Continuously on indicates: Safety has been disengaged. PX4 flight controller may be armed with Throttle down and to the right.
  - When the LED is continuously on indicating Safety Disengaged it may be toggled back to a Safety engaged condition by depressing the Safety button for 5 seconds.

Note

Both the Safety engaged and Safety disengaged conditions require the button to be held down for 5 seconds to toggle them. This is a safety mechanism to prevent accidental disarming during flight and accidental arming during transportation.

Setting the Home Position

Tip

It is very important to acquire GPS lock in order for RTL, Loiter, Auto or any GPS dependent mode to work properly.
For Plane the home position is initially established at the time the plane acquires its GPS lock. It is then continuously updated as long as the autopilot is disarmed.

This means that if you execute an RTL, your plane will return to the location at which it armed. If the plane you arm is not a good return point then please setup a rally point instead. A rally point will be used in preference to the home location for RTL.

BEFORE EVERY FLIGHT: Before you take off, hold your aircraft in your hands and switch to FBWA mode, then pitch and tilt the plane it to confirm that the control surfaces move the correct way to return it to level flight. (The ailerons and elevators will move; the rudder only coordinates turns with the ailerons in flight, so it won't move much on the ground). This will ensure that you haven't accidentally reversed a channel.

You should do this before every flight, just as you move your control surfaces with your RC transmitter to ensure that nothing's reversed. **Failing to do this is the #1 cause of crashes.**

Note

As a safety measure, your throttle will only arm on the ground in Manual mode, Stabilize or for an autotakeoff in Auto mode. It will not come on in any other Auto mode until you are in motion in the air.

First flight

It is highly recommended that you switch into either Stabilize or Fly By Wire mode and observe the behavior of the control surfaces. They should move to return the plane to level when you pitch or roll it. If it isn't rock solid, you can tune the gains by following the instructions here.

If you have not tuned your PID gains then you may like to consider doing the first takeoff in AUTOTUNE mode. That will start the tuning process as soon as you takeoff.

Second flight

For your second flight, change the third mode (position 3 of your RC mode switch) to RTL in the Mission Planner's mode setup page.

This will test navigation. The aircraft should return to the location at which it armed (or the nearest Rally point) and orbit at a fixed altitude (which can be set with the Mission Planner).

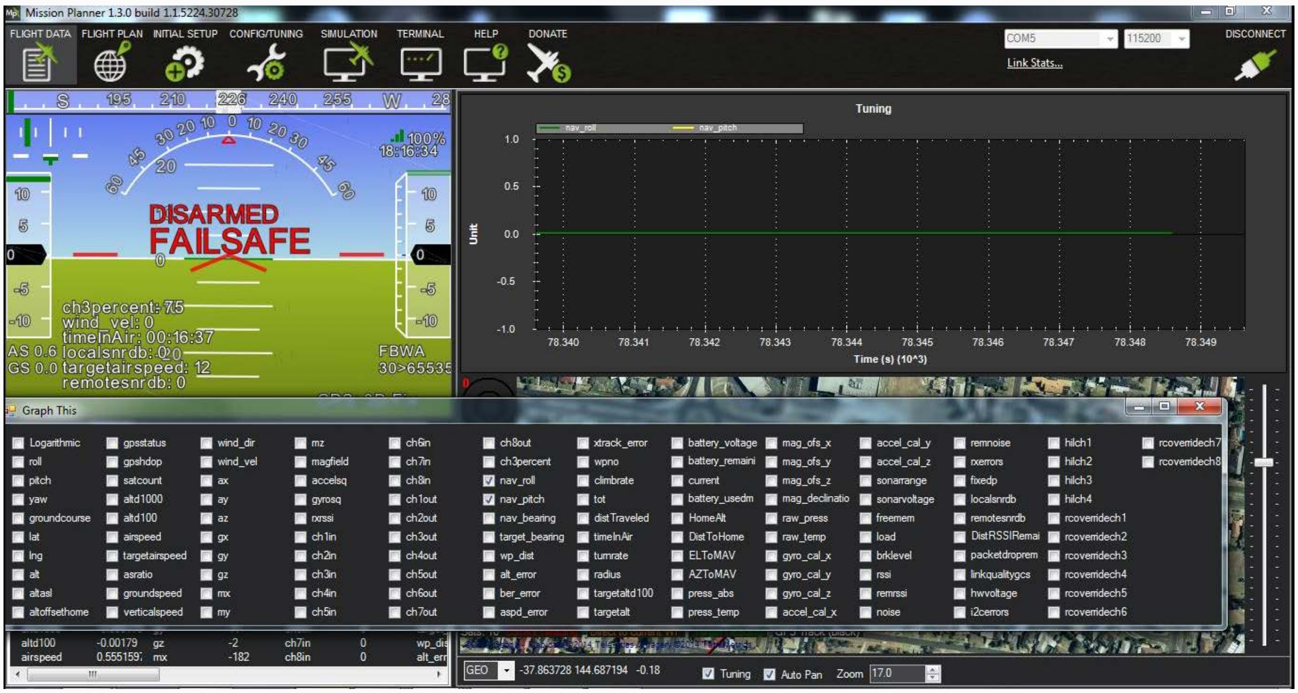
If it does not return crisply and circle overhead in a near-perfect circle, you need to tune the autopilot a bit for your particular airframe. This can usually be done by adjusting the Roll parameters, as described here.

Once all this has checked out, you can program waypoint missions and test then in Auto mode.

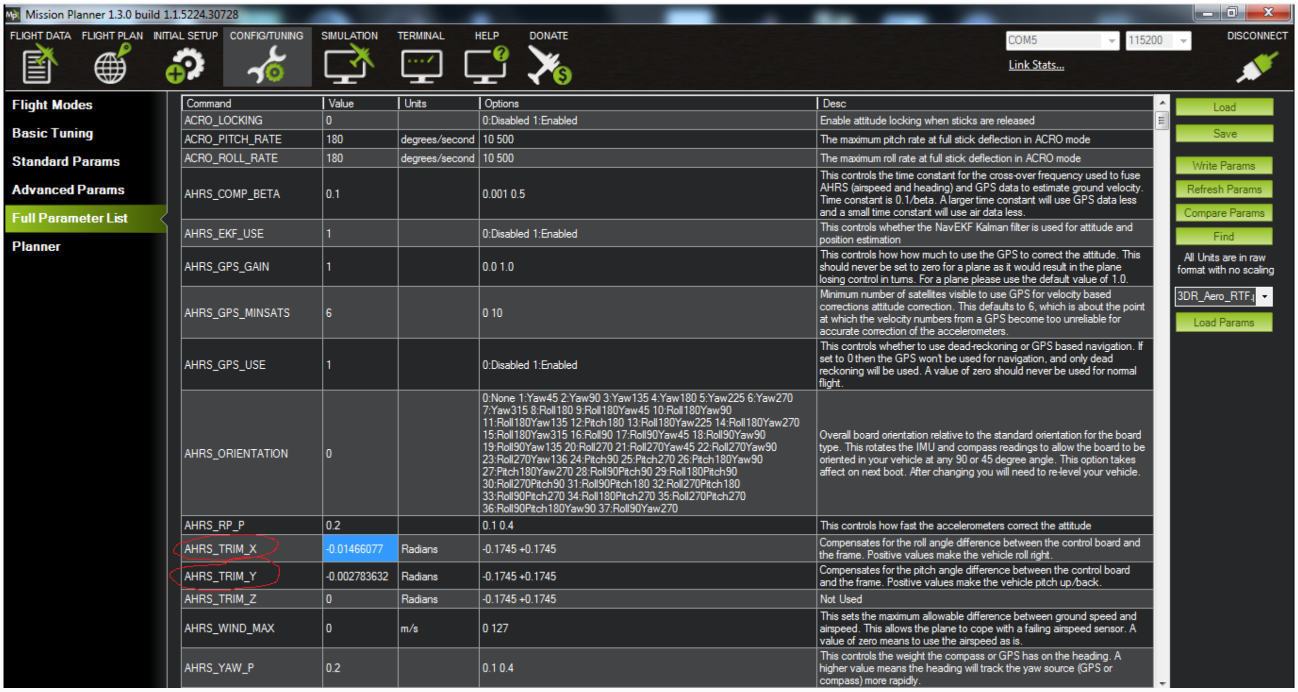
Level Adjustment

You may find after flying your plane in FBWA that it has a tendency to turn in one direction and/or gains or loses height on a mid throttle setting with the transmitter sticks centred. If this happens, perform the following:

1) With your autopilot powered on the ground and connected to your mission planner, select FBWA on your transmitter, select the FLIGHT DATA tuning window and plot the nav_roll and nav_pitch data. With your transmitter sticks centred, these should both be zero as shown in this screenshot. If they are not, you need to repeat your RC calibration or adjust your transmitter trims and repeat the FBWA flight test



If they are zero, then you need to adjust the `AHRS_TRIM_X` (roll) and `AHRS_TRIM_Y` (pitch) for the difference in angle between the autopilot board and your planes attitude when flying straight and level. You can change these by going to **CONFIG/TUNING | Full Parameter List** and adjusting the parameters as shown in the screenshot below.



Warning

These parameters are in radians (every 0.01 is about 0.6 of a degree) so adjust in increments of 0.01 initially. If the plane turns to the left, AHRS_TRIM_X should be increased. If the plane loses height with mid throttle, AHRS_TRIM_Y should be increased.

Arming Plane

Arming your plane

Before you can fly your plane you need to arm it. Arming the aircraft before flight has two purposes:

- prevent the motor from turning when the pilot is not ready to fly (a safety feature)
- prevent takeoff before the autopilot is fully configured and ready to fly

In past releases of APM:Plane arming was optional, and the requirement to arm (controlled by the ARMING_REQUIRED parameter) was disabled by default. This was changed for the 3.3.0 release to require arming by default.

The key thing that arming does is to enable the motor. You will not be able to start the motor (ie. control the throttle) until the aircraft is armed.

Note: If you have AHRS_EKF_USE enabled (you are using the EKF) then it is particularly important that you have arming enabled with arming checks enabled. Flying EKF without arming checks may cause a crash.

Configuring Arming

There are three parameters which control how arming works:

- **ARMING_REQUIRE**: this controls whether an arming step is required. The default is 1, meaning that arming is required before takeoff. If set to 0 then arming is not required (the plane starts off armed).
- **ARMING_CHECK**: this controls what checks the autopilot does before arming is allowed. The default is 1, meaning all checks are done. Most users should leave it at 1, as the arming checks are important to ensure the autopilot is ready.
- **ARMING_RUDDER**: This parameter allows you to configure rudder based arming/disarming. The default is 1, meaning you are able to arm with right rudder. If you set this to 2 you can also disarm with left rudder. If you set this to 0 then you will only be able to arm/disarm via a ground station.

How to Arm

When you are ready to fly you can ask Plane to arm. This can be done in two ways:

- **Rudder Arming**. Hold the rudder stick fully to the right and the throttle stick fully down for 2 seconds.
- **GCS Arming**. Press the arming button on your ground station

How to Disarm

Since APM:Plane 3.4.0 it is possible to disarm using the transmitter. This is done holding throttle at minimum and rudder to the left for 2 seconds. In ArduPlane this condition could be accidentally triggered by pilots while flying so there are additional requirements prior to disarm:

- You need to allow rudder disarming by changing **ARMING_RUDDER** parameter to 2 (ArmOrDisarm).
- The flight controller needs to make sure that you are not actually flying. There is an algorithm for this that uses the **airspeed sensor** readings. So you need this source available and giving values lower enough (in a windy day you might not be able to disarm even landed if the plane thinks you are still flying)

You can also disarm without using the transmitter with one of the following methods:

- use a ground station to issue a disarm command
- use the safety switch on your aircraft (on Pixhawk)
- after an auto-landing the plane will automatically disarm after 20 seconds if still on the ground (controlled by LAND_DISARMDELAY parameter)

Visual and Audible signals

APM:Plane will provide visual and audio clues to the arming state if your autopilot has LEDs and a buzzer. The clues are:

- if the autopilot is disarmed, but is ready to arm then the large 3-colour LED will be flashing green
- if the autopilot is armed and ready to fly the large 3-colour LED is solid green
- when the autopilot is ready to arm it will play a “ready to arm” sound on the buzzer
- when the autopilot is armed or disarmed it will play the corresponding sound

See the sounds page to listen to what the buzzer sounds like for each state.

Arming Checks

Before allowing arming the autopilot checks a set of conditions. All conditions must pass for arming to be allowed. If any condition fails then a message explaining what failed is set to the GCS.

The checks performed are:

- Safety switch. The PX4 safety switch must be set to the safety-off state before arming is allowed. This is either done by pressing the safety switch for 2 seconds until it stops flashing, or you can disable the use of the safety switch by setting BRD_SAFETY_ENABLE=0
- Barometer check. The barometer must be healthy (getting good data)
- Inertial Sensor Checks. The accelerometers and gyroscopes must all be healthy and all be calibrated. If you have more than one accel or gyro then they need to be consistent with each other.
- AHRS checks. The AHRS (attitude heading reference system) needs to be initialized and ready. Note that if you have the EKF enabled this may take up to 30 seconds after boot.
- Compass checks. All compasses must be configured and calibrated, and need to be consistent with each other (if you have more than one compass)
- GPS Checks. You need to have a 3D GPS fix.
- Battery checks. The battery voltage must be above the failsafe voltage (if configured)
- Airspeed checks. If you have configured an airspeed sensor then the sensor needs to be working.
- Logging checks. The logging subsystem needs to be working (ie. a microSD must be fitted and working on PX4)
- RC Control checks. You need to not be in RC failsafe

Throttle output when disarmed

When the plane is disarmed the throttle channel will not respond to pilot input. There are two possible behaviours you can configure:

- ARMING_REQUIRE=1. When disarmed the minimum value for the throttle channel (normally RC3_MIN) will be sent to the throttle channel
- ARMING_REQUIRE=2. When disarmed no pulses are sent to the throttle channel. Note that some ESCs will beep to complain that they are powered on without a control signal

Diagnosing failure to arm

It can be frustrating if your plane refuses to arm. To diagnose arming issues follow this guide

Check it is ready to arm¶

If your board has a “ready to arm” LED (the large LED in the middle of the board on a Pixhawk) then that LED should be flashing green when the board is ready to arm. If it is flashing yellow then that indicates that one of the arming checks is not passing.

Try arming¶

Try sending an arm command with your GCS. If arming is refused then a message will be sent from the autopilot to the GCS indicating why it is refusing to arm.

Rudder arming¶

If you are using right-rudder + zero-throttle to arm and you don't get a message on your GCS giving a arming failure reason then it may be that your RC calibration is a bit off and the autopilot is not quite seeing zero throttle or isn't quite seeing full right rudder.

Reasons for refusing to arm¶

When the autopilot refuses to arm it sends a STATUSTEXT MAVLink message to the GCS explaining why it is refusing. The possible reasons why the autopilot can refuse to arm are:

- **barometer not healthy**. This is very rare. If it happens repeatedly then you may have a barometer hardware fault.
- **airspeed not healthy**. If you have a airspeed sensor fitted and the autopilot is not getting an airspeed reading it will refuse to arm.
- **logging not available**. If your microSD card has failed or is corrupt then logging won't be available and you cannot arm.
- **gyros not healthy**. If the gyros have failed the autopilot will refuse to arm. This is rare, and if it happens repeatedly then you may have a hardware failure.
- **gyros not calibrated**. This happens when the automatic gyro calibration at startup didn't converge. Try rebooting the autopilot with the plane held still.
- **accels not healthy**. If the accelerometers have failed the autopilot will refuse to arm. Try recalibrating your accelerometers.
- **GPS accuracy errors**. There are 4 types of GPS arming errors that can be reported. They are "GPS vert vel error", "GPS speed error", "GPS horiz error", "GPS numsats". Try moving your plane for better GPS reception or switching off any RF sources (such as a FPV transmitter) that may be interfering with your GPS.
- **Mag yaw error**. This happens when your compass is badly out of alignment. Check your compass orientation and re-do your compass calibration or move your plane further away from any magnetic materials.
- **EKF warmup**. This happens when the EKF is still warming up. Wait another 10 seconds and try again.
- **AHRS not healthy**. This means the EKF is not healthy. If the error persists then try rebooting your board.
- **3D accel cal needed**. This happens when you have not done a 3D accelerometer calibration.
- **Inconsistent accelerometers**. This happens when you have multiple IMUs (such as the Pixhawk which has two) and they are not consistent. This can be caused by temperature changes. If the error doesn't clear itself after a minute you will need to redo your accelerometer calibration.
- **Inconsistent gyros**. This happens when you have multiple gyros and they are not reporting consistent values. If the error does not clear itself after 30 seconds then you will need to reboot.
- **Limit errors**. The arming checks some of your parameter settings to make sure they are in a reasonable range. The checks are "LIM_ROLL_CD too small", "LIM_PITCH_MAX too small", "LIM_PITCH_MIN too large", "invalid THR_FS_VALUE".
- **GPS n has not been fully configured**. This happens with a uBlox GPS where the GPS driver is unable to fully configure the GPS for the settings that are being requested. This can be caused by a

bad wire between the autopilot and GPS, or by a bad response from the GPS. If the message is about "GPS 0" then it is the first GPS. If it is "GPS 1" then it is the 2nd GPS. If you get a failure for the 2nd GPS and don't have two GPS modules installed then set GPS_TYPE2 to zero to disable the 2nd GPS

Flight Modes

This article provides links to Plane's flight modes.

Overview

Plane has a wide range of built in flight modes. Plane can act as a simple flight stabilization system, a sophisticated autopilot, a training system or a flight safety system depending on what flight mode and options you choose.

Flight modes are controlled through a radio transmitter switch, via mission commands, or using commands from a ground station (GCS) or companion computer.

Major Flight Modes

All of the flight modes below have optional additional controls that may be used to change the behaviour to suit particular flying needs. After you have read the introductory material below it is highly recommended that you look through the complete set of Plane parameters so you can explore the full range of functionality available.

MANUAL

STABILIZE

FLY BY WIRE_A (FBWA)

FLY BY WIRE_B (FBWB)

AUTOTUNE

TRAINING

ACRO

CRUISE

AUTO

Return To Launch (RTL)

LOITER

CIRCLE

GUIDED

Mission Specific Modes

When flying an AUTO mission Plane has some sub-modes that are set using mission items. The two main sub-modes are TAKEOFF and LAND.

TAKEOFF

LAND

MANUAL Mode¶

Regular RC control, no stabilization. All RC inputs are passed through to the outputs. The only ways in which the RC output may be different from inputs are as follows:

- if a configured failsafe or geofence triggers, and Plane takes control
- if the VTAIL_OUTPUT option is enabled, then a software VTAIL mixer is applied on the output
- if the ELEVON_OUTPUT option is enabled, then a software Elevon mixer is applied on the output

STABILIZE Mode¶

RC control with simple stabilization. If you let go of the sticks then Plane will level the plane. This is a bit like flying a plane with a lot of dihedral. You will find that while it is possible to do maneuvers like rolls and loops in stabilize mode, the tendency of the plane to right itself will make these maneuvers difficult. For people wanting the plane to mostly fly itself, with the pilot just telling it where to fly, you are better off using the FLY BY WIRE_A (FBWA) mode.

In stabilize mode the throttle is limited by the `THR_MIN` and `THR_MAX` settings.

Note

STABILIZE mode is not a good mode to use for tuning the control loops. You are far better off using FLY BY WIRE_A (FBWA) for that.

FBWA Mode (FLY BY WIRE_A)¶

This is the most popular mode for assisted flying in Plane, and is the best mode for inexperienced flyers. In this mode Plane will hold the roll and pitch specified by the control sticks. So if you hold the aileron stick hard right then the plane will hold its pitch level and will bank right by the angle specified in the `LIM_ROLL_CD` option (in centidegrees). It is not possible to roll the plane past the roll limit specified in `LIM_ROLL_CD`, and it is not possible to pitch the plane beyond the `LIM_PITCH_MAX`/`LIM_PITCH_MIN`settings.

Note that holding level pitch does not mean the plane will hold altitude. How much altitude a plane gains or loses at a particular pitch depends on its airspeed, which is primarily controlled by throttle. So to gain altitude you should raise the throttle, and to lose altitude you should lower the throttle. If you want Plane to take care of holding altitude then you should look at the FlyByWireB mode.

In FBWA mode throttle is manually controlled, but is constrained by the `THR_MIN` and `THR_MAX` settings.

In FBWA mode the rudder is under both manual control, plus whatever rudder mixing for roll you have configured. Thus you can use the rudder for ground steering, and still have it used for automatically coordinating turns.

FBWB Mode (FLY BY WIRE_B)¶

The FBWB mode is similar to FLY BY WIRE_A (FBWA), but Plane will try to hold altitude as well. Roll control is the same as FBWA, and altitude is controlled using the elevator. The target airspeed is controlled using the throttle.

To control your altitude in FBWB mode you use the elevator to ask for a change in altitude. If you leave the elevator centred then Plane will try to hold the current altitude. As you move the elevator Plane will try to gain or lose altitude in proportion to how far you move the elevator. How much altitude it tries to gain for full elevator deflection depends on the `FBWB_CLIMB_RATE` parameter, which defaults to 2 meters/second. Note that 2 m/s is quite a slow change, so many users will want to raise `FBWB_CLIMB_RATE` to a higher value to make the altitude change more responsive.

Whether you need to pull back on the elevator stick or push forward to climb depends on the setting of the `FBWB_ELEV_REV` parameter. The default is for pulling back on the elevator to cause the plane to climb. This corresponds to the normal response direction for a RC model. If you are more comfortable with the reverse you can set FBWB_ELEV_REV to 1 and the elevator will be reversed in FBWB mode.

Note that the elevator stick does not control pitch, it controls target altitude. The amount of pitch that will be used to achieve the requested climb or descent rate depends on your TECS tuning settings, but in general the autopilot will try to hold the plane fairly level in pitch, and will primarily climb or descend by raising or lowering the throttle. This can be disconcerting for people used to flying in FBWA mode, where you have much more direct control over pitch.

If you have an airspeed sensor then the throttle will control the target airspeed in the range `ARSPD_FBW_MIN` to `ARSPD_FBW_MAX`. If throttle is minimum then the plane will try to fly at `ARSPD_FBW_MIN`. If it is maximum it will try to fly at `ARSPD_FBW_MAX`.

If you don’t have an airspeed sensor then the throttle will set the target throttle of the plane, and Plane will adjust the throttle around that setting to achieve the desired altitude hold. The throttle stick can be used to push the target throttle up beyond what it calculates is needed, to fly faster.

As with FBWA, the rudder is under a combination of manual control and auto control for turn coordination.

You should also have a look at CRUISE mode, as it is generally better than FBWB, especially if there is significant wind. In CRUISE mode the aircraft will hold a ground track as opposed to just levelling the wings when you don’t input any roll with the aileron stick.

AUTOTUNE Mode¶

The AUTOTUNE mode flies in the same way as FLY BY WIRE_A (FBWA), but it does automatic tuning of roll and pitch control gains. Please read the full documentation on AUTOTUNE for more details.

TRAINING Mode¶

TRAINING mode is ideal for teaching students manual R/C control. It gives users complete control over the rudder and throttle, but clips the maximum roll and maximum/minimum pitch to certain limits which cannot be exceeded. Starting from Plane 3.4, TRAINING mode also restricts plane roll to the Stall Prevention roll limits.

More specifically:

- If the roll is less the LIM_ROLL_CD parameter than the pilot has manual roll control. If the plane tries to roll past that limit then the roll will be held at that limit. The plane will not automatically roll back to level flight, but it will prevent the pilot from rolling past the limit. The same applies to pitch - the pilot has manual pitch control until the LIM_PITCH_MIN or LIM_PITCH_MAX limits are reached, at which point the plane won't allow the pitch to go past those limits.
- When turning, the autopilot will monitor the demanded bank angle and airspeed and work out if there is a sufficient margin above the stall speed to turn at the demanded bank angle. If not the bank angle is limited to the safe value. The stall prevention system will always allow a bank of at least 25 degrees (to ensure you can still manoeuvre if your airspeed estimate is badly off).
- The rudder and throttle are both completely under manual control.

ACRO Mode¶

ACRO (for acrobatic) is a mode for advanced users that provides rate based stabilization with attitude lock. It is a good choice for people who want to push their plane harder than you can in FLY BY WIRE_A (FBWA) or STABILIZE mode without flying in MANUAL. This is the mode to use for rolls, loops and other basic aerobatic maneuvers, or if you just want an "on rails" manual flying mode.

To setup this mode you need to set ACRO_ROLL_RATE and ACRO_PITCH_RATE. These default to 180 degrees/second, and control how responsive your plane will be about each axis.

When flying in ACRO the aircraft will try to hold it's existing attitude if you have no stick input. So if you roll the plane to a 30 degree bank angle with 10 degrees pitch and then let go of the sticks, the plane should hold that attitude. This applies upside down as well, so if you roll the plane upside down and let go of the sticks the plane will try to hold the inverted attitude until you move the sticks again.

When you apply aileron or elevator stick the plane will rotate about that axis (in body frame) at a rate proportional to the amount of stick movement. So if you apply half deflection on the aileron stick then the plane will start rolling at half of `ACRO_ROLL_RATE`.

So to perform a simple horizontal roll, just start in level flight then hold the aileron stick hard over while leaving the elevator stick alone. The plane will apply elevator correction to try to hold your pitch while rolling, including applying inverse elevator while inverted.

In the current implementation the controller won't use rudder while the plane is on it's side to hold pitch, which means horizontal rolls won't be as smooth as a good manual pilot, but that should be fixed in a future release. This also means that it won't hold knife-edge flight.

Performing a loop is just as simple - just start with wings level then pull back on the elevator stick while leaving the aileron alone. The controller will try to hold your roll attitude through the loop. You can stop the loop upside down if you like as part of maneuvers such as Immelman turns or cuban eights.

Note that if you are using ACRO mode to try and teach yourself aerobatic flying then it is highly recommended that you setup a geo-fence in case you get disoriented.

Warning

It is very easy to stall your plane in ACRO mode, and if you stall you should change to MANUAL mode to recover.

- make sure you know the limitations of your airframe, and what the correct stall recovery procedure is. This varies a lot between airframes. Search for stall recovery tutorials for R/C aircraft and read them
- don't overload your airframe, only fly ACRO mode with a lightly loaded plane
- make sure you have enough airspeed for whatever maneuver you are attempting. Throttle and speed control is completely under manual pilot control in ACRO mode
- practice stall recovery before trying anything too fancy. Make sure you practice when you have plenty of altitude to give you time to try different recovery strategies

It can be a lot of fun flying ACRO mode, but you can also easily stall and crash hard. Automatic stall detection and recovery in autopilots is an area of research, and is not yet implemented in Plane, so if you do stall then recovery is up to you. The best mode for recovery is MANUAL.

CRUISE Mode¶

Cruise mode is a bit like FLY BY WIRE_B (FBWB), but it has "heading lock". It is the ideal mode for longer distance FPV flight, as you can point the plane at a distant object and it will accurately track to that object, automatically controlling altitude, airspeed and heading.

The way it works is this:

- if you have any aileron or rudder input then it flies just like FBWB. So it holds altitude until you use the elevator to change the target altitude (at the `FBWB_CLIMB_RATE` rate) and it adjusts airspeed based on throttle
- when you let go of the aileron and rudder sticks for more than 0.5 seconds it sets an internal waypoint at your current location, and projects a target waypoint one kilometre ahead (note that heading lock will only activate if you have GPS lock, and have a ground speed of at least 3 m/s)
- as it flies along it heads for the target waypoint, and constantly updates that target to always be one kilometre ahead, leaving the previous waypoint as the position that you centred the aileron and rudder sticks
- as long as you don't touch the aileron or rudder, it will run the same navigation system it uses for waypoints, including crabbing, cross-track etc, so it will very accurately hold that ground course even in the face of changing wind conditions

One of the nicer aspects of CRUISE mode is how it handles rudder. If you give it some rudder then the roll controller will keep the wings level, but the plane will yaw with the rudder. So you get a “wings level” turn, allowing you to rotate your flight to point at whatever geographic feature you want to head towards. Then when you let go of the rudder it will head straight for that point.

Note that you can configure CRUISE mode to do terrain following on PX4/Pixhawk. See the terrain following documentation.

Warning

Make sure you only fly FPV if it is allowed by your country’s flight and airspace control rules. Many countries do not allow non-line-of-sight flight without a special operating license.

AUTO Mode¶

In AUTO mode Plane will follow a mission (a set of GPS waypoints and other commands) set by your ground station. When entering AUTO mode Plane will continue from whatever mission item it was last doing, unless you have reset the mission.

When in AUTO Plane will by default allow the pilot to influence the flight of the plane by using “stick mixing”, which allows for aileron, elevator and rudder input to steer the plane in a way that can override the autopilot control. Whether this is enabled is determined by the STICK_MIXING option. By default stick mixing behaves the same as FLY BY WIRE_A (FBWA) mode.

Warning

“Home” position is always supposed to be your Plane’s actual GPS takeoff location:

1. It is very important to acquire GPS lock before arming in order for RTL, Loiter, Auto or any GPS dependent mode to work properly.
2. For Plane the home position is initially established at the time the plane acquires its GPS lock. It is then continuously updated as long as the autopilot is disarmed.
   - This means if you execute an RTL in Plane, it will return to the location where it was when it was armed - assuming it had acquired GPS lock.
   - Consider the use of Rally Points to avoid returning directly to your arming point on RTL

RTL Mode (Return To Launch)¶

In RTL mode the plane will return to its home location (the point where the plane armed - assuming it had GPS lock) and loiter there until given alternate intructions (or it runs out of fuel!). As with AUTO mode you can also “nudge” the aircraft manually in this mode using stick mixing. The target altitude for RTL mode is set using the ALT_HOLD_RTL parameter in centimeters.

Alternatively, you may configure the plane to return to a Rally Point, rather than the home location.

Warning

“Home” position is always supposed to be your Plane’s actual GPS takeoff location:

1. It is very important to acquire GPS lock before arming in order for RTL, Loiter, Auto or any GPS dependent mode to work properly.
2. For Plane the home position is initially established at the time the plane acquires its GPS lock. It is then continuously updated as long as the autopilot is disarmed.
   - This means if you execute an RTL in Plane, it will return to the location where it was when it was armed - assuming it had acquired GPS lock.
   - Consider the use of Rally Points to avoid returning directly to your arming point on RTL

LOITER Mode¶

In LOITER mode the plane will circle around the point where you started the loiter, holding altitude at the altitude that you entered loiter in. The radius of the circle is controlled by the WP_LOITER_RAD parameter, but is also limited by your `NAV_ROLL_CD` limit, and your *:ref:`NAVL1_PERIOD`* navigation tuning. As with *Return To Launch (RTL) <rtl-mode>* and AUTO mode you can “nudge” the plane while in LOITER using stick mixing, if enabled.

Warning

“Home” position is always supposed to be your Plane’s actual GPS takeoff location:

1. It is very important to acquire GPS lock before arming in order for RTL, Loiter, Auto or any GPS dependent mode to work properly.
2. For Plane the home position is initially established at the time the plane acquires its GPS lock. It is then continuously updated as long as the autopilot is disarmed.
   - This means if you execute an RTL in Plane, it will return to the location where it was when it was armed - assuming it had acquired GPS lock.
   - Consider the use of Rally Points to avoid returning directly to your arming point on RTL

CIRCLE Mode¶

Circle mode is similar to LOITER, but doesn’t attempt to hold position. This is primarily meant as a failsafe mode and is the mode that the aircraft will enter by default for 20 seconds when a failsafe event occurs, before switching to RTL.

Circle mode is deliberately a very conservative mode, and doesn’t rely on GPS positioning as it is used when GPS fails. It will do a large circle, The bank angle is set to the `LIM_ROLL_CD` divided by 3, to try to ensure the plane remains stable even without GPS velocity data for accelerometer correction. That is why the circle radius is so large.

Circle mode uses throttle and pitch control to maintain altitude at the altitude where it started circling.

GUIDED Mode¶

The GUIDED mode is used when you want the aircraft to fly to a specific point on the map without setting up a mission. Most ground control stations support a “click to fly to” feature where you can click a point on the map and the aircraft will fly to that location then loiter.

The other major use for GUIDED mode is in geo-fencing. When the geo-fence is breached the aircraft will enter GUIDED mode, and head to the predefined geo-fence return point, where it will loiter until the operator takes over.

TAKEOFF Mode¶

Auto takeoff is set by the mission control scripting only. The takeoff mission specifies a takeoff pitch and a target altitude. During takeoff Plane will use the maximum throttle set by the `THR_MAX` parameter. The takeoff mission item is considered complete when the plane has reached the target altitude specified in the mission.

Before takeoff it is important that the plane be pointing into the wind, and be aligned with the runway (if a wheeled takeoff is used). The plane will try to hold its heading during takeoff, with the initial heading set by the direction the plane is facing when the takeoff starts. It is highly recommended that a compass be enabled and properly configured for auto takeoff, as takeoff with a GPS heading can lead to poor heading control.

If you are using a wheeled aircraft then you should look at the `WHEELSTEER_*` PID settings for controlling ground steering. If you are hand launching or using a catapult you should look at the `TKOFF_THR_MINACC` and `TKOFF_THR_MINSPD` parameters.

LAND Mode¶

Auto Land is set by the mission control scripting only. Throttle and altitude is controlled by the autopilot. After getting closer `LAND_FLARE_ALT` meters from the target altitude or `LAND_FLARE_SEC` seconds from the target landing point the plane will “flare” to the `LAND_PITCH_CD` pitch (in centidegrees) and will hold heading for the final approach.

Setting up Plane for reliable auto-takeoff and landing is very airframe dependent, and it is recommended that you first get some experience flying your aircraft in FBWA mode, and be ready to take over control in manual or FBWA mode the first few times you use an automatic takeoff or landing.

You should also look through the complete list of parameters, as there are a lot of parameters that help control takeoff and landing for different situations.

Flight Features

Plane has a lot of advanced features which can be accessed either via configuration parameters or via mission commands. The list below will give you some information on some of the more widely used features. For a full set of features see the full parameter list and the MAVLink protocol definition.

Automatic Takeoff

Plane can automatically launch a wide range of aircraft types. The instructions below will teach you how to setup your mission to support automatic takeoff.

Basic Instructions

The basic idea of automatic takeoff is for the autopilot to set the throttle to maximum and climb until a designated altitude is reached. To cause the plane to execute a takeoff, add a NAV_TAKEOFF command to your mission, usually as the first command. There are two parameters to this command - the minimum pitch, and the takeoff altitude. The minimum pitch controls how steeply the aircraft will climb during the takeoff. A value of between 10 and 15 degrees is recommended for most aircraft. The takeoff altitude controls the altitude above home at which the takeoff is considered complete. Make sure that this is high enough that the aircraft can safely turn after takeoff. An altitude of 40 meters is good for a wide range of aircraft.

During takeoff the wings will be held level to within LEVEL_ROLL_LIMIT degrees. This prevents a sharp roll from causing the wings to hit the runway for ground takeoffs.

Note that the takeoff direction is set from the direction the plane is pointing when the automatic takeoff command is started. So you need to point the plane in the right direction, then switch to AUTO mode. During the first stage of the takeoff the autopilot will use the gyroscope as the principal mechanism for keeping the aircraft flying straight. After sufficient speed for a good GPS heading is reached the aircraft will switch to using the GPS ground track which allows it to account for a cross-wind.

You should try to launch into the wind whenever possible.

Hand Launching

A hand launch is a common method to launch smaller aircraft, such as foam gliders. Plane has a number of parameters that control how a hand launch is done. If you are planning to hand launch your aircraft in AUTO mode then please look through these options carefully.

The keys to a good hand launch are:

- if the propeller is behind your hand during launch then ensuring that the motor does not start until it is past your hand
- ensuring that the aircraft does not try to climb out too steeply

The main parameters that control the hand launch are:

- TKOFF_THR_MINACC
- TKOFF_THR_DELAY
- TKOFF_THR_MINSPD
- TECS_PITCH_MAX

When the auto takeoff mission command starts (usually by switching to AUTO mode) the autopilot starts in “throttle suppressed” mode. The throttle will not start until the conditions set by the TKOFF_THR_ parameters met.

The TKOFF_THR_MINACC parameter controls the minimum forward acceleration of the aircraft before the throttle will engage. The forward acceleration comes from the throwing action of your arm as you launch the aircraft. You need to set this value high enough that the motor won’t start automatically when you are carrying the aircraft normally, but low enough that you can reliably trigger the acceleration with a normal throwing action. A value of around 15 m/s/s is good for most aircraft.

The TKOFF_THR_DELAY parameter is a delay in 1/10 of a second units to hold off starting the motor after the minimum acceleration is reached. This is meant to ensure that the propeller is past your hand before the motor starts. A value of at least 2 (which is 0.2 seconds) is recommended for a hand launch.

The TKOFF_THR_MINSPD parameter is a minimum ground speed (as measured by the GPS) before the motor starts. This is an additional safety measure to ensure the aircraft is out of your hand before the motor starts. A value of 4m/s is recommended for a hand launch.

Note that if your aircraft is a “tractor” type with the motor at the front then you may want to set TKOFF_THR_DELAY and TKOFF_THR_MINSPD to zero, or use lower values.

The final parameter you should think about is the `TECS_PITCH_MAX` parameter. That controls the maximum pitch which the autopilot will demand in auto flight. When set to a non-zero value this replaces the LIM_PITCH_CD parameter for all auto-throttle flight modes. Setting this parameter to a value which is small enough to ensure the aircraft can climb reliably at full throttle will make takeoff much more reliable. A value of 20 is good for most aircraft.

Catapult Launching

The main differences between catapult launching and hand launching is that a catapult will usually give the aircraft a greater level of acceleration, and the risk involved is primarily that the propeller will strike the catapult frame instead of hitting your hand.

In most other ways a catapult launch is like a hand launch, and the same 4 key parameters apply. If your catapult is setup so that the motor cannot run until the aircraft is out of the frame of the catapult then you will need to choose the parameters to ensure there is sufficient delay. Often this means a higher value for TKOFF_THR_MINACC (say 20m/s/s) and a longer delay before the GPS ground speed is measured. Some experimentation may be needed, but a value of `TKOFF_THR_DELAY` of 5 is likely to be good for many catapults.

Bungee Launching

A bungee launch uses a long piece of stretched elastic to launch the aircraft. This can be a cheaper alternative to a catapult and gives good results for a lot of small to medium sized models.

The same 4 parameters that apply to hand launch and catapult launch also apply to a bungee launch, but the values you will need are different. The main risk with a bungee launch (especially with a pusher propeller) is that the propeller will strike the bungee cord, damaging either the propeller or the bungee or both. To prevent this from happening you should have a much higher value of `TKOFF_THR_DELAY`, making it high enough that the aircraft will have released the bungee before the motor starts. A value of around 50 (giving a 5 second delay) may be a good starting point.

Runway Takeoffs (CTOL)

The final class of takeoff is runway takeoff, also known as wheeled takeoff or CTOL (Conventional Takeoff and Landing). Setting up for a good automatic takeoff from a runway is a bit more complex than the other types of launches with more parameters to set and more tuning required.

One key consideration with runway takeoffs is whether you have a tail dragger (tail wheel steering) or tricycle undercarriage (nose wheel steering). Automatic takeoff is easier with a tricycle undercarriage aircraft, with a tail dragger needing additional parameters.

The key parameters for runway takeoff are:

- TKOFF_TDRAG_ELEV
- TKOFF_TDRAG_SPD1
- TKOFF_THR_SLEW
- TKOFF_ROTATE_SPD
- TECS_PITCH_MAX
- GROUND_STEER_ALT

In addition to those parameters you also need to tune ground steering, so that the ground steering controller is able to reliably steer the aircraft. See the separate page on setting up ground steering. As part of this tuning you will need to setup the GROUND_STEER_ALT parameter.

The first two parameters are primarily for tail dragger aircraft, although they can also be used to hold the nose of a tricycle aircraft down on takeoff.

The TKOFF_TDRAG_ELEV parameter is used to hold the tail of a tail dragger hard on the runway during the initial stages of takeoff, to give it enough grip on the runway to steer. For a tail dragger this is normally set to 100, meaning that 100% up elevator is applied during the initial stages of takeoff. For a tricycle undercarriage plane that needs a bit of extra weight on the nose for good steering you may find that a value of -20 (meaning 20% down elevator) may help.

When the takeoff starts, the autopilot will apply `TKOFF_TDRAG_ELEV` elevator (as a percentage) until the aircraft reaches a speed of `TKOFF_TDRAG_SPD1` meters per second. You need to set TKOFF_TDRAG_SPD1 to a speed below the takeoff speed, but above the speed where the aircraft is able to steer using its rudder. When the aircraft reaches TKOFF_TDRAG_SPD1 it will release the elevator and instead use the normal flight pitch controller to try to hold the pitch level. That will have the effect of raising the tail on a tail dragger aircraft.

The TKOFF_ROTATE_SPD parameter controls when the autopilot will try to raise the nose (pitch up) to leave the ground. This needs to be a speed at which the aircraft can sustain a climb, so it should be at least 2 meters per second above the stall speed of the aircraft, preferably more. A higher value will mean a longer takeoff (and thus need more runway).

The TKOFF_THR_SLEW parameter controls the throttle slew rate (as a percentage per second) during takeoff. This is used to allow the throttle to ramp up at a rate appropriate for your aircraft. How high this should be depends on the type of aircraft. It is usually a good idea for a ground takeoff to limit how fast the throttle ramps up to prevent torque from the motor causing large steering changes. A value of 20 (meaning 20% throttle change per second) is good for many tail draggers. A tricycle undercarriage aircraft may be able to handle a larger throttle slew rate.

As with other types of takeoff the `TECS_PITCH_MAX` parameter controls the maximum pitch used when climbing on takeoff. Make sure that this is limited to a value that the aircraft can use to climb quickly at full throttle. A value of around 20 degrees is good for a wide range of aircraft.

Testing Ground Takeoff in FBWA mode

It is sometimes useful to test the takeoff code using the FBWA flight mode. The way you do this is to set the `FBWA_TDRAG_CHAN` parameter to an RC input channel on your transmitter for a switch (usually a momentary switch, such as the trainer switch). When this RC channel goes high while you are on the runway waiting for takeoff in FBWA mode the autpilot will check if you have configured the `TKOFF_TDRAG_ELEV` and `TKOFF_TDRAG_SPD1` parameters. If they have been set to non-zero values then the elevator will be controlled in FBWA in an identical manner to how it is controller for an AUTO takeoff. The elevator will go to the `TKOFF_TDRAG_ELEV` value (usually 100% for a tail dragger) as soon as that RC channel goes high, and will stay there until the aircraft reaches a ground speed of TKOFF_TDRAG_SPD1 meters per second.

This provides a convenient way to test auto takeoff in FBWA mode, and also is a nice way to get better ground steering in FBWA mode in general.

Automatic Landing

This article explains how to land Plane as part of a mission plan and includes information about how a landing can be safely aborted.

Configuring for Automatic Landing

Plane can automatically land an aircraft, as part of a mission plan.

To land the plane you need to add a NAV_LAND command to the end of your mission indicating the latitude, longitude and altitude of your desired touchdown point. In most cases, the altitude should be set to 0. During landing, the autopilot will shut down the throttle and hold the current heading when the plane reaches the flare point, controlled by the parameters described below.

Key Parameters¶
The key parameters that control automatic landing are:

- LAND_FLARE_ALT
- LAND_FLARE_SEC
- LAND_PITCH_CD
- TECS_LAND_ARSPD
- TECS_LAND_SPDWGT

The meaning and recommended value of each of these parameters is described below.

Setting the Flare Point¶

The “flare” is the final stage of the landing when the autopilot cuts the throttle and raises the pitch, increasing drag and slowing the aircraft to sink onto the ground. The appropriate time to flare depends on the type of aircraft, and is controlled by the LAND_FLARE_ALT and LAND_FLARE_SEC parameters.

The primary control of the flare is the `LAND_FLARE_SEC` parameter. This is the time in seconds before the aircraft would hit the ground if it continued with its current descent rate. So if the plane is descending at 2 meters/second and you set the `LAND_FLARE_SEC` to 3 then the aircraft would flare at an altitude of 6 meters above the ground. By using a time to impact to control the flare the aircraft is able to flare at a higher altitude if it is descending quickly, and at a lower altitude if it is descending slowly. That helps ensure the flare is able to produce a smooth touchdown.

The second control is `LAND_FLARE_ALT`. That is an altitude above the ground in meters at which the aircraft will flare, regardless of its descent rate.

The appropriate values for these two parameters depends on how the autopilot is estimating its altitude above the ground. If the autopilot has a good rangefinder (such as LIDAR) then you can safely choose quite small numbers, and flare close to the ground. That will generally produce a better landing. A value for `LAND_FLARE_SEC` of 1.5 and `LAND_FLARE_ALT` of 2 is a good place to start with a LiDAR. If you are relying solely on a barometer for landing altitude then you will probably need higher values, to account for barometric error.

Controlling the glide slope¶

Another important factor in setting up the flare point is the glide slope. The glide slope is the ratio of the distance from the last waypoint to the landing point, and the height difference between the last waypoint and the landing point. For example, if the landing point is 300 meters from the last waypoint, and the last waypoint is 30 meters above the ground then the glide slope is 10%.

If the glide slope is too steep then the aircraft will not be able to flare in time to avoid crashing, plus the autopilot may not be able to keep the plane on the approach slope accurately. It is recommended that you start with a glide slope of at most 10%. What glide slope your plane can handle will depend on how well your pitch controller tuning is, how good your TECS tuning is, and the landing speed you ask for.

If you find your aircraft is not following the desired glide slope accurately then you should first check your pitch tuning in your logs, and ensure that the demanded and achieved pitch match within a couple of degrees during landing. If they don’t then look at the documentation on pitch tuning (or possibly re-run AUTOTUNE). If the demanded and achieved pitch do match then you should check your TECS logs to ensure that the demanded and achieved airspeed are matching during landing. Have a look at the TECS tuning patch for more information.

You should also be aware that many model aircraft can glide for long distances, and it may be that your requested glide slope and airspeed combination just isn’t achievable.

Landing Airspeed¶

Automatic landing is greatly assisted by the use of an airspeed sensor. When using an airspeed sensor the landing approach speed (the speed coming down the glide slope) is controlled by the TECS_LAND_ARSPD parameter, in meters/second.

You need to choose a value for `TECS_LAND_ARSPD` that is above the stall speed of your aircraft, but low enough that the aircraft is able to lose altitude and land in a reasonable distance. Note that as the stall speed is dependent on the weight of your aircraft you will need to adjust the landing speed if you change the aircraft's weight significantly (such as by adding batteries or a camera).

To further improve landing you can use a Pre-Flare to reduce airspeed just before the flare. This is enabled by setting either LAND_PF_ALT or LAND_PF_SEC to either enter a pre-flare state at a fixed altitude or at an estimated seconds to ground (given your current decent rate). Once the Pre-Flare is triggered the desired airspeed becomes LAND_PF_ARSPD. This value should be lower than `TECS_LAND_ARSPD` but greater than the stall speed. This is particularly useful where reverse thrust is available. However, some aircraft can handle a stall landing so setting this to a very low number (1) will tell the aircraft to bleed off as much airspeed as possible before the flare.

Controlling the approach¶

During the landing approach the autopilot needs to balance the requested airspeed (set by `TECS_LAND_ARSPD`) and the requested glide slope and landing position (set by the previous waypoint and final landing point). The default configuration tries to balance these two demands equally, but for some aircraft you may want to prioritize one over the other.

The priority of airspeed control versus height control is set using the TECS_LAND_SPDWGT parameter. A value of 1 (the default) means a balance between the two. A value closer to two gives a higher priority to airspeed and a value closer to zero gives a higher priority to height control. For example, if you are landing at a speed close to the stall speed you may wish to place a high priority on the airspeed control. To do that you should set `TECS_LAND_SPDWGT` to a value close to 2, such as 1.9.

If what you want in a landing is precision in the position where it lands then you should set `TECS_LAND_SPDWGT` to a low number, such as 0.2 or even 0.0. In that case the plane will still try to achieve the target landing airspeed by using the throttle, but it will not try to control airspeed with pitch.

If you are landing a glider (or any aircraft without a motor) then you should set `TECS_LAND_SPDWGT` to 2.0, so that airspeed is the priority and pitch will be used to control airspeed.

In most cases a value of -1 gives the best result. This special value will auto-adjust the value during the landing, scaling it from your normal TECS_SPDWEIGHT value down to zero at the point of landing. So up in the sky during approach you maintain good airspeed but by the time you land the emphasis is on a more accurate landing.

Controlling the flare¶

The final stage of the landing is called the "flare". During the flare the aircraft tries to retain a course along the line between the last waypoint and the landing waypoint, and it controls it's height solely using a target descent rate. Once the flare is started the throttle is "disabled" - set to some value between THR_MIN and zero.

The main job of the flight controller in the flare is to try to achieve the descent rate specified in the

TECS_LAND_SINK parameter. That defaults to 0.25 meters/second, which is a reasonable touchdown vertical speed for most models. To achieve that speed the TECS controller uses pitch control only as the motor has been forced to zero.

The primary parameters which affect the ability of the aircraft to achieve the desired descent rate are LAND_PITCH_CD, TECS_LAND_DAMP and the main pitch tuning parameters.

The `LAND_PITCH_CD` parameter sets the minimum pitch target in the flare (in centi-degrees). This parameter is very airframe specific and is designed to prevent the nose of the aircraft being too far down on touchdown causing issues with damaging the landing gear or breaking a propeller. For most aircraft this should be a small positive number (such as 300, meaning 3 degrees), but for some belly landing aircraft a small negative number can be good, to allow the nose to be kept down a small amount to reduce the chance of stall if the flare happens too far off the ground.

Note that the actual pitch of the aircraft can be quite a bit above `LAND_PITCH_CD` as the TECS controller tries to control the descent rate. The maximum pitch is controlled by the TECS_PITCH_MAX parameter if it is non-zero, otherwise by the LIM_PITCH_MAX parameter.

The `TECS_LAND_DAMP` parameter is a damping constant for the pitch control during. A larger number will cause the pitch demand to change more slowly. This parameter can be used to reduce issues with sudden pitch changes when the flare happens.

After the Flare¶

After the plane flares it continues to navigate, but with zero throttle. The navigation direction is a line extrapolated forward through the landing point from the last waypoint. Note that the navigation roll will be limited to LEVEL_ROLL_LIMIT (which defaults to 5 degrees) to prevent wing strike, so if there is a significant cross-wind then it is likely that the aircraft will not be able to maintain the exact path.

If your aircraft is consistently landing long (which can happen for a variety of reasons) then you can adjust TECS_LAND_SRC to either force a stall (negative) or bring it down (positive). This value will adjust your `TECS_LAND_SINK` proportional to the distance from the LAND point. This helps ensure you land in a reasonable distance from the LAND point.

Note

Possible causes of landing long include ground effect giving the aircraft more lift as it is close to the ground or simply the aircraft traveling very fast.

When the plane has stopped moving for LAND_DISARMDELAY seconds (default 20 seconds) it will disarm the motor. Optionally, you can disable servo movement once LAND_DISARMDELAY has triggered by setting LAND_THEN_NEUTRL.

Using a rangefinder¶

If you have fitted a rangefinder to your aircraft then you can use it for much more accurate landing control. To allow the rangefinder to be used for landing you need to set the RNGFND_LANDING parameter to 1.

When using a rangefinder for landing the altitude given by the rangefinder is used only in the landing approach and to determine the flare point, and is designed to allow the aircraft to more accurately follow the glide slope and to flare at the right time.

Note

The effectiveness of a rangefinder can depend on the surface you are flying over, so it is a good idea to do some low passes in a flight mode such as FBWA first, then examine the logs to check that the rangefinder is working correctly.
Also note that if you have a longer range rangefinder then it is a very good idea to set the minimum range of the rangerfinder well above zero. For example, the PulsedLight Lidar has a typical range of over 40 meters, and when it gets false readings it tends to read ranges of less than 1 meter. Setting RNGFND_MIN_CM to 150 will discard any rangerfinder readings below 1.5 meters, and will greatly improve the robustness of the Lidar for landing.

Improving the landing¶

The key to a good landing is the autopilot knowing how far off the ground it is. With the default setup the only sensor available to detect altitude is the barometer. Unfortunately barometers suffer from three main types of error:

- barometric drift due to changes in atmospheric pressure
- barometric drift due to changes in the temperature of the autopilot electronics
- barometric error due to local pressure changes from airflow around the barometer

The ideal setup for good automatic landing is to have a Lidar. A Lidar can measure the distance to the ground very accurately, and doesn't suffer from drift. If you have a Lidar installed you can enable its use for landing with `RNGFND_LANDING=1`.

If a Lidar isn't fitted then there are a few things you can do to minimise barometric error problems with auto-land

- perform a barometer calibration after the electronics have warmed up. The easiest way to do this with a Pixhawk is to disarm the plane with the safety switch. When the plane is disarmed it assumes it is on the ground and will zero the barometer to the current pressure.
- try to prevent direct airflow over the autopilot that could cause speed related pressure changes
- fly shorter flights, allowing for less time for airpressure changes. Check your logs and see if the landing is happening at zero altitude consistently

With planes that belly land it can also work well to setup the landing with a shallow pitch (in `LAND_PITCH_CD`) and set a slightly higher altitude to flare at. That will only work if your stall speed is low enough that gliding for a while will work reliably.

Using DO_LAND_START

Sometimes it is useful to trigger an automatic landing as part of an RTL (return to launch). To do this you need to do two things:

- add a DO_LAND_START mission item to your mission, just before the start of your landing sequence
- set the RTL_AUTOLAND parameter to 1 or 2

The way it works is that when the plane enters an RTL it checks to see if the parameter RTL_AUTOLAND is set to 1 or 2. If it is then the current mission is searched for a mission item of type DO_LAND_START. If one is found then the plane will automatically enter AUTO mode and land, starting at the part of the mission just after the `DO_LAND_START` marker.

The exact behaviour depends on the `RTL_AUTOLAND` value:

- If `RTL_AUTOLAND=1` then the plane will first RTL as normal, then when it starts circling the return point (home or a rally point) it will then switch to the AUTO mission after the `DO_LAND_START` and land
- If `RTL_AUTOLAND=2` then the plane will bypass the RTL completely and go straight to the landing sequence.

You can optionally include more than one `DO_LAND_START` mission item in your mission. If that is done then the latitude/longitude of the `DO_LAND_START` mission items is used to choose which landing sequence to use. The `DO_LAND_START` closest to the current location is used. This can be useful if you have multiple landing sequences for different wind conditions or different areas.

How to abort an auto-landing

A landing-abort mechanism is provided to allow you to abort a landing sequence in a safe, controlled, and expected way. Custom abort behaviour can be pre-programmed as part of the mission or you can use the default abort mechanism. To enable this feature set param LAND_ABORT_THR=1.

There are three steps to this feature: #. Trigger an abort #. The behavior during the abort #. The mission state after the abort completes.

Note

This section describes the abort behavior introduced in Plane 3.4.
Step 1) Abort land triggers¶
The are three ways to trigger an auto-landing abort. All of them will only work while in AUTO mode and currently executing a `LAND` waypoint mission item:

- *Send the ``MAV_CMD_DO_GO_AROUND`` command using a GCS.* Mission Planner has a button labeled “Abort Landing” on the FlightData Actions tab.
- *RC input Throttle > 90%*. This will trigger an abort while staying in AUTO mode. The throttle only needs to be high briefly to trigger it. Don’t forget to lower it!
- *Mode change*. For human piloted landing abort you can switch out of AUTO mode into, for example MANUAL/STABILIZE/FBWA, and navigate the aircraft safely however you’d like. Using this method will skip abort behavior step 2 because it is being done manually. When switching back to AUTO the mission will resume as described in step 3 below.

Step 2) Abort land flight behavior¶
The abort behaviour has a default configuration and does not require a pre-planned mission. The default abort behavior is to simulate an auto-takeoff: pitch up at least 10 degrees and set throttle to TKOFF_THR_MAX and hold the heading until it reaches a target altitude of 30m. It is possible to override

the pitch and altitude to allow for a customized behavior.

- Pitch minimum. If there was a NAV_TAKEOFF ever executed on this mission then the same pitch will be re-used here.
- Target altitude. If NAV_LAND param1 is >0 then it is used as a target altitude in meters. Else If a NAV_TAKEOFF was ever executed on this mission then the same altitude will be re-used here.

This step is skipped if the abort trigger is via mode change because it is assumed the pilot manually took over and flew the aircraft to a safe altitude at the pitch and throttle of their choosing.

Step 3) Mission state after an aborted landing completes¶

Once an abort land has completed, by either reaching the target altitude or switching back to AUTO, the mission index will have changed and you will no longer be executing a NAV_LAND command. The mission index will change to be one of these three options and checked for in this order:

- If the NAV_LAND mission item is followed by mission item CONTINUE_AND_CHANGE_ALT with param1 = 0 or 1 then the mission index will increment once to that command and execute it like normal. This can be followed by further post-abort mission planning for any custom planned mission behavior.
- Else If there is a DO_LAND_START in the mission then it jumps to that index.
- Else the mission index decrements once to be the index before the NAV_LAND. This will ensure the same landing approach is repeated.

Reverse-Thrust Landing

Some ESC's allow for reverse direction. When using reverse on the propeller it will generate a negative thrust which can be used to slow you down. During a steep landing approach this method can be used to maintain a stable airspeed allowing you to land much more precisely even with a LiDAR Baro bump on the approach. To use this feature it is highly recommend to use an airspeed sensor and a rangefinder (see above) for an accurate altitude.

Note

Reverse-thrust landings are available starting from Plane v3.5.1.

Key Parameters¶

The key parameters that control reverse thrust landing in addition to the ones listed in section 1.1 are:

- LAND_PF_ALT
- LAND_PF_SEC
- LAND_PF_ARSPD
- USE_REV_THRUST
- TECS_APPR_SMAX
- RC3_TRIM
- THR_MIN

ESC (Electronic Speed Controller)¶

Hardware selection and programming¶

Most ESCs can operate in forwards and reverse, however that is usually not a stock feature and may need to be reprogrammed to do it. Any SimonK and BLHeli compatible ESC can be flashed to support reverse thrust.

Here's info about BLHeli compatible ones.

Hardware configuration¶

Note

Remove propeller while configuring ESCs.

Configure your ESC for reverse thrust by changing it's neutral point. Many ESC require custom firmware to accomplish this. Search google or your ESC's mfgr for instructions on how to configure your particular ESC.

Set these:

1. Minimum PWM to 1000, mid to 1500, and maximum to 2000.
2. `THR_MIN` to a negative value such -100. Next set `RC3_TRIM` (or whatever `RCx` is mapped to throttle via `RCMAP_THROTTLE`) to your ESC's mid value.

Determining your max glide slope angle¶

For a steep landing approach, the limitation is how well you can maintain your desired airspeed. This is determined by your aircraft's ability to create reverse thrust (motor+prop combo) and its resistance to slowing down (aircraft mass). In most cases extreme steepness is unnecessary, but possible. With an over-sized motor and lightweight aircraft you can come in as steep as 60 degrees.

To determine your steepest approach angle, set TECS_APPR_SMAX very high as to not limit you (e.g. 99). Next, plan a mission with a steeper than normal approach (try 15 degrees and go up from there). Watch your airspeed on the approach - the plane should be able to maintain TECS_LAND_ARSPD with only 75% of the available reverse throttle range. If not, you're coming in too steep for the negative-thrust-to-mass ratio of your aircraft.

Tip

Keep in mind that whatever value you determine as your maximum may not be acceptable in all wind conditions. It is best to be a little conservative.

Setting up the Pre-Flare¶

With a rangefinder and airspeed sensors installed, at the pre-flare point we will have an accurate airspeed and altitude reading. This gives us a good idea of our momentum and stable "initial conditions" to the final flare. Set `LAND_PF_ALT` (or `LAND_PF_SEC`) to a fairly high point (for example 10m) and adjust from there. Next set `LAND_PF_ARSPD` to a value just above your stall speed.

When LAND_PF_ALT is reached the airspeed demand will instantly go from TECS_LAND_ARSPD to LAND_PF_ARSPD. This will cause it to slam on the brakes via increased reverse thrust to reduce speed instead of just maintaining a given speed.

The trick is to set `LAND_PF_ALT` to an altitude where it achieves `LAND_PF_ARSPD` before killing the throttle at `LAND_FLARE_ALT` (which occurs at a somewhat low altitude - around 1 or 2m).

Example, `TECS_LAND_ARSPD = 15`, `LAND_PF_ARSPD = 12`, `LAND_PF_ALT=12`, `LAND_FLARE_ALT=2`. Depending on your slope, mass of aircraft and motor+propellor thrust ability, you're expecting the aircraft to decelerate from 15 to 12m/s airspeed while dropping 10m. These are the critical params to adjust to ensure a smooth and slow flare.

Flare¶

Now that you are starting the flare with a stable and predictable airspeed, it's much easier to control the flare. If you've already tuned your flare for an auto-land without reverse thrust you'll want to retune it. You'll notice you're coming in much slower.

Other benefits of reverse-thrust landings¶

LiDAR baro bump is handled better¶

On a long duration flight the baro drift will cause an altitude offset that is not detectable until the LiDAR detects the ground (at which point the aircraft "snaps" to the glide-slope). This causes an increased airspeed moments before your flare, causing a touch-down beyond the intended land point. With reverse-thrust the "snap" still happens, but the TECS controller automatically changes the throttle demand to maintain the desired airspeed.

Determining actual stall speed of your aircraft¶

Unless you really know what you're doing, stall speed can be hard to estimate. To be sure of the value you normally need to slowly decrease your airspeed until you stall - with the consequent problem that now you have a stalled plane falling out of the sky.

With LAND_PF_ALT and LAND_PF_ARSPD you can check your stall speed much lower to the ground. To know the exact moment it stalls, check your logs for when roll and roll_desired diverge.

Inverted Flight

Plane now supports inverted flight. You can tell if your version supports inverted flight by looking for the INVERTEDFLT_CH EEPROM configuration parameter.

Details

The INVERTEDFLT_CH parameter allows you to configure a transmitter switch to enable inverted flight with APM. The default is zero, which means no inverted flight. To enable it, choose a RC input channel which is not being used for anything else, and setup your transmitter to give a PWM value larger than 1750 when a previously unused 2 position switch is enabled, and a value well below 1750 when it is not enabled.

For example, if channel 7 is currently unused, and you set INVERTEDFLT_CH to 7, then if your transmitter sends 1900 on channel 7 the APM will flip the plane upside down and continue flying in whatever mode it is in, but with the plane inverted.

Transmitter Setup

Enabling inverted flight only changes how APM stabilises the plane. It will stabilise it with a roll of 180 degrees from normal whenever inverted flight is enabled.

To get the most out of inverted flight it can be useful to also invert your transmitter elevator and rudder controls while your inverted flight switch is enabled. This should be possible with any good transmitter. For example, with the ER9X firmware on a Turnigy 9X transmitter you could use mixes like this:

```
CH01   +100% AIL
CH02   +100% ELE
     * -100% FULL Switch(GEA)
CH03   +100% THR
CH04   +100% RUD
     * -100% FULL Switch(GEA)
CH07   +100% MAX  Switch(GEA)
```

that would setup the elevator and rudder to invert when the GEAR switch is active, along with setting CH7 to full.

Interaction with flight modes

Inverted flight can be used with any of the stabilized flight modes (which means anything but manual mode). That means you can switch on inverted flight during a mission and the plane will continue the mission inverted. Make sure you disable it before trying to land!

Tips

Here are some general tips about inverted flight

- Try it out in HIL simulation first to get used to it, and ensure your transmitter is setup right.
- make sure you have plenty of height before engaging or disengaging inverted flight. Depending on your airframe, you could lose a lot of height while flipping over, and it may be that your airframe struggles to maintain height while inverted

Stall Prevention

Plane has logic to try to prevent stalls when in certain modes. This page describes how it works and how to configure it.

Basic Idea

One of the most common ways that users crash is through a stall. A stall happens when the airflow over the wing is not sufficient to hold the aircraft in the air. Stalls can happen at any speed, but the most common type of stall is a low speed stall, where the airflow is too slow to provide enough lift.

The amount of airflow you need over the wing to hold the aircraft in the air depends (among other things) on the bank angle you are flying at. If you are banked over hard then you need to fly faster to get enough lift to stay in the air. This is because the lift is produced perpendicular to the wing, so when the wing is rolled over it provides only part of the lift to holding the aircraft up, and the rest of the lift goes into making the aircraft turn.

When the STALL_PREVENTION parameter is set to 1 then two things are done:

- when in roll controlled modes the autopilot will monitor your demanded bank angle and airspeed and work out if you have sufficient margin above the stall speed to turn at the demanded bank angle. If you don't then the turn will be limited to the safe limit, but it will always allow a bank of at least 25 degrees (to ensure you can still manoeuvre if your airspeed estimate is badly off).
- when in auto-throttle modes the autopilot will also raise the minimum airspeed in the TECS system to the level at which the current demanded bank angle can be safely achieved. So it will add more engine power or lower the nose to raise the airspeed so that the bank angle that the navigation controller is demanding can be achieved without a stall

Configuring stall prevention

There are two key parameters that control stall prevention:

- The STALL_PREVENTION parameter. If this is set to zero then no stall prevention is done. This may be useful if you have no airspeed sensor and the synthetic airspeed estimate is not good enough
- The ARSPD_FBW_MIN parameter, which is the configured minimum airspeed for level flight. It is this value that is scaled with the bank angle to calculate the safe airspeed for any demanded bank angle

Affected modes

The following modes are affected by the bank angle limiting of the stall prevention code:

- FBWA
- FBWB
- AUTOTUNE
- CIRCLE
- RTL
- AUTO
- CRUISE
- LOITER
- GUIDED
- TRAINING (from Plane 3.4)

In each of these modes the aircraft is either calculating a desired bank angle for the navigation code, or the user is inputting a desired bank angle via the aileron stick. In both cases the stall prevention code will limit the bank angle based on the amount of margin between `ARSPD_FBW_MIN` and the current airspeed.

The following modes are affected by the automatic speed increase in TECS when the bank angle is limited by stall prevention:

- CIRCLE
- RTL
- AUTO
- CRUISE
- LOITER
- GUIDED

These are all of the “auto throttle” modes, where the TECS controller controls pitch and throttle.

Example Usage

An example of how stall prevention works is in AUTO mode when making a turn for an automated landing. The landing approach will aim for an airspeed of TECS_LAND_ARSPD, which is often quite low, and may be just above the stall speed. If the final turn is sharp (it is often 90 degrees) the plane may be trying to perform a sharp turn while dropping speed for landing. That could induce a stall. The stall prevention code means two things happen:

- the initial turn is kept shallow enough to prevent the stall
- the target airspeed is kept high enough during the turn to allow the turn to complete, after which it is lowered

Nose Down in FBWA and AUTOTUNE

The FBWA mode has an additional special feature to help prevent straight line stalls. It is often difficult for a pilot to judge the airspeed of a plane when it is coming head on, and so many pilots find themselves flying too slow when they are trying to fly gently in FBWA mode. This is especially true if the ArduPilot pitch controller is well tuned and manages to keep the nose up even when flying slowly.

To make a level flight low speed stall less likely some additional down pitch is added in FBWA and AUTOTUNE modes based on the throttle position. The amount of down pitch added is based on the STAB_PITCH_DOWN parameter, which defaults to 2 degrees. At zero throttle this full down pitch is added. If throttle is above `TRIM_THROTTLE` then no down pitch is added. Between those two values the down pitch is added in proportion to the throttle.

This has the effect of slightly lowering the nose when you lower throttle in FBWA and AUTOTUNE modes, which makes the plane gain a bit of speed, and thus makes it less likely to stall. The value you need for `STAB_PITCH_DOWN` depends on how much drag your plane has. A very sleek model will need a smaller value. A high drag model will need a larger value.

Geo-Fencing in Plane

The Geo-Fencing support in Plane allows you to set a virtual ‘fence’ around the area you want to fly in, specified as an enclosed polygon of GPS positions plus a minimum and maximum altitude.

Note

In Plane you MUST have the fence polygon drawn for the geofence parameters to work. In Copter you can specify a maximum fence altitude without the fence polygon and Copter will adhere to it but Plane requires you to draw the polygon for any of the fence parameters including the max/min ALT to work.
When fencing is enabled, if your plane goes outside the fenced area then it will switch to GUIDED mode, and will fly back to a pre-defined return point, and loiter there ready for you to take over again. You then use a switch on your transmitter or use commands in your Ground Control Station (GCS) to take back control.

Use for R/C training

One of the main uses of geo-fencing is to teach yourself (or someone else) to fly radio controlled planes. When you have a properly configured geo-fence it is very hard to crash, and you can try manoeuvres that would normally be too likely to end in a crash, trusting the APM to ‘bounce’ the plane off the geo-fence before the flight ends in disaster.

Geo-fencing can be combined with any APM flight mode. So for a raw beginner, you would combine it with one of the stabilised flight modes (such as STABILIZE or FBWA). Once the pilot has gained some confidence you could combine it with MANUAL mode, which gives direct control of the plane and allows for the most interesting aerobatic manoeuvres. When used in this way the APM stays out of your way completely, just passing the controls to the servos directly, and only takes control if you go outside the fenced area or outside the defined altitude range.

Use for containment

During fully autonomous operation the fence can be used as a failsafe measure to ensure the aircraft stays within the intended flight area. During fully autonomous operation use the FENCE_AUTOENABLE parameter and the plane will automatically engage the fence after takeoff is complete and automatically disable the fence when it arrives at a landing waypoint. Set the FENCE_AUTOENABLE parameter to 1 to use this feature. For details on setting up landing and takeoff waypoints see the Planning a Mission with Waypoints and Events page.

Of course, the fence can be used for containment in semi-autonomous missions as well (missions where, e.g., takeoff and/or landing are manual) – the fence would still be enabled/disabled via the R/C transmitter or the GCS in that case – FENCE_AUTOENABLE is optional.

Setting up geo-fencing

To setup geo-fencing in Plane you need to configure several things:

1. the boundary of the fence, as a set of GPS points
2. the action to take on fence breach
3. the location of the return point; note that optionally you may use the FENCE_RET_RALLY parameter to have the plane return to the closest Rally Point instead of the fence return point.
4. the minimum and maximum altitude of the fenced area
5. what RC channel on your transmitter you will use to enable geo-fencing (if any)
6. an optional setting (FENCE_AUTOENABLE) when you want to configure the fence to automatically enable after an autonomous takeoff and automatically disable after an autonomous landing
7. how you want to take back control after a fence breach

These can all be setup using the APM Mission Planner.

There are a few rules that you must follow when setting up your fence boundary:

1. the return point must be inside the fence boundary
2. the fence boundary must be fully enclosed. This means it must have at least 4 points, and the last point must be the same as the first point
3. the boundary can have at most 18 points

If you setup your fence with the APM planner it should ensure you follow these rules.

Please remember when making your fence boundary that your plane will have some momentum when it hits the fence, and will take time to turn back to the return point. For a plane like the SkyWalker we recommend an additional safety margin of around 30 meters inside the true boundary of where you want to fly. The same goes for the minimum altitude - you need to make it high enough that the APM has time to recover from a fast dive. How much margin you need depends on the flight characteristics of your plane.

Apart from the fence boundary, the following MAVLink parameters control geo-fencing behaviour:

1. FENCE_ACTION - the action to take on fence breach. This defaults to zero which disables geo-fencing. Set it to 1 to enable geo-fencing and fly to the return point on fence breach. Set to 2 to report a breach to the GCS but take no other action. Set to 3 to have the plane head to the return point on breach, but the pilot will maintain manual throttle control in this case.
2. FENCE_MINALT - the minimum altitude in meters. If this is zero then you will not have a minimum altitude.
3. FENCE_MAXALT - the maximum altitude in meters. If this is zero then you will not have a maximum altitude.
4. FENCE_CHANNEL - the RC input channel to watch for enabling the geo-fence. This defaults to zero, which disables geo-fencing. You should set it to a spare RC input channel that is connected to a two position switch on your transmitter. Fencing will be enabled when this channel goes above a PWM value of 1750. If your transmitter supports it it is also a good idea to enable audible feedback when this channel is enabled (a beep every few seconds), so you can tell if the fencing is enabled without looking down.
5. FENCE_TOTAL - the number of points in your fence (the return point plus the enclosed boundary). This should be set for you by the planner when you create the fence.
6. FENCE_RETALT - the altitude the aircraft will fly at when flying to the return point and when loitering at the return point (in meters). Note that when FENCE_RET_RALLY is set to 1 this parameter is ignored and the loiter altitude of the closest Rally Point is used instead. If this parameter is zero and FENCE_RET_RALLY is also zero, the midpoint of the FENCE_MAXALT and FENCE_MINALT parameters is used as the return altitude.
7. FENCE_AUTOENABLE - if set to 1, the aircraft will boot with the fence disabled. After an autonomous takeoff completes the fences will automatically enable. When the autonomous mission arrives at a landing waypoint the fence automatically disables.
8. FENCE_RET_RALLY - if set to 1 the aircraft will head to the nearest Rally Point rather than the fence return point when the fence is breached. Note that the loiter altitude of the Rally Point is used as the return altitude.

Note

A Rally Point can be outside of the geofence but this is NOT recommended. If you have a rally point outside the geofence you will need to disable the geofence using FENCE_CHANNEL before you can control the plane again otherwise the plane will stay in GUIDED mode FOREVER circling the rally point. Once the geofence is disabled you should fly the plane back inside the geofence and then re-enable it. One additional parameter may be useful to get the most out of geo-fencing. When you breach the fence, the plane will switch to GUIDED mode and fly back to the return point (or the nearest Rally Point, if FENCE_RET_RALLY has been set to 1). Once you are back inside the fence boundary you are able to take control again, and you need to tell the APM that you want to take control. You can do that in one of 3 ways:

1. changing modes using the APM mode switch on your transmitter, or changing modes via the Mission Planner GCS (e.g., change from GUIDED mode to AUTO mode).
2. disabling and re-enabling geo-fencing using the FENCE_CHANNEL channel
3. set the RST_SWITCH_CH MAVLink parameter to another two-position channel that is attached to a spring loaded switch. The RST_SWITCH_CH parameter defaults to zero which disables it. If you set it to a channel then you can use this channel switch to take back control after a fence breach.

If not flying completely autonomously, I find that using RST_SWITCH_CH is the best option for geo-fencing as it means that the APM has fencing enabled throughout the flight, and you don’t get any behaviour change by switching modes. It does take up another channel though, so some people may not have enough channels to use it.

Setting up the fence boundary

APMPlanner¶

To setup a fence boundary you should use the ‘Flight Planner’ screen in the APM Planner.

Start by right-clicking the location you want for the return point and choosing ‘Set return location’. The return point should be somewhere in the middle of your flight area, and in easy visual range of where you will be standing when you fly.

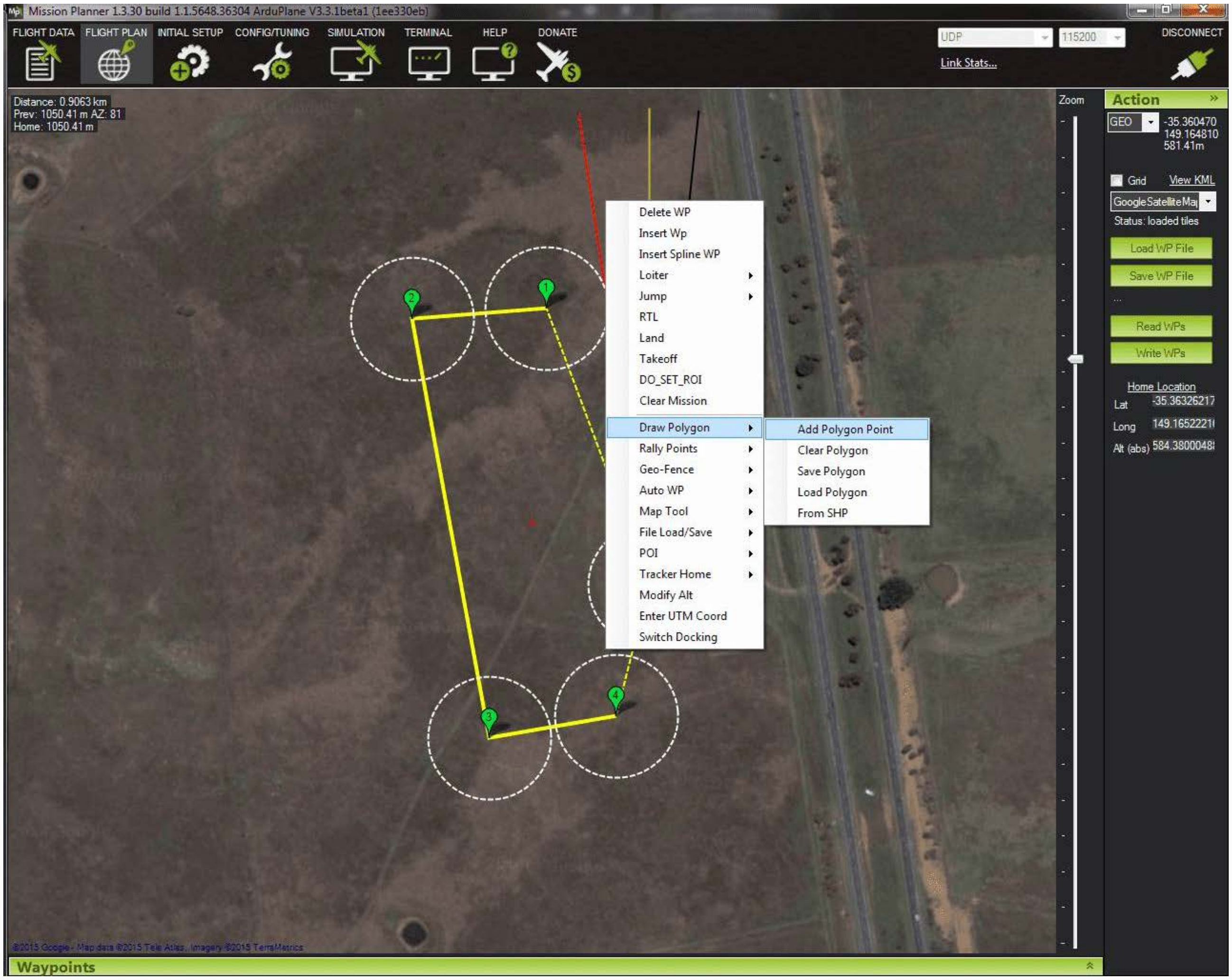
After you’ve set the return point you should right click on the first point on the boundary of the fence you want. Choose ‘Draw Polygon ->

Add polygon point’. You are then in polygon mode, and you should left-click to add each point in the boundary of your fence. The planner will automatically complete the polygon by connecting the last point to the first one.

You can then right-click and choose geo-fencing upload to send your fence boundary to the APM. The planner will ask you for the minimum and maximum altitude (in meters) of your fence before uploading. You can also save your fence to a file for later loading.

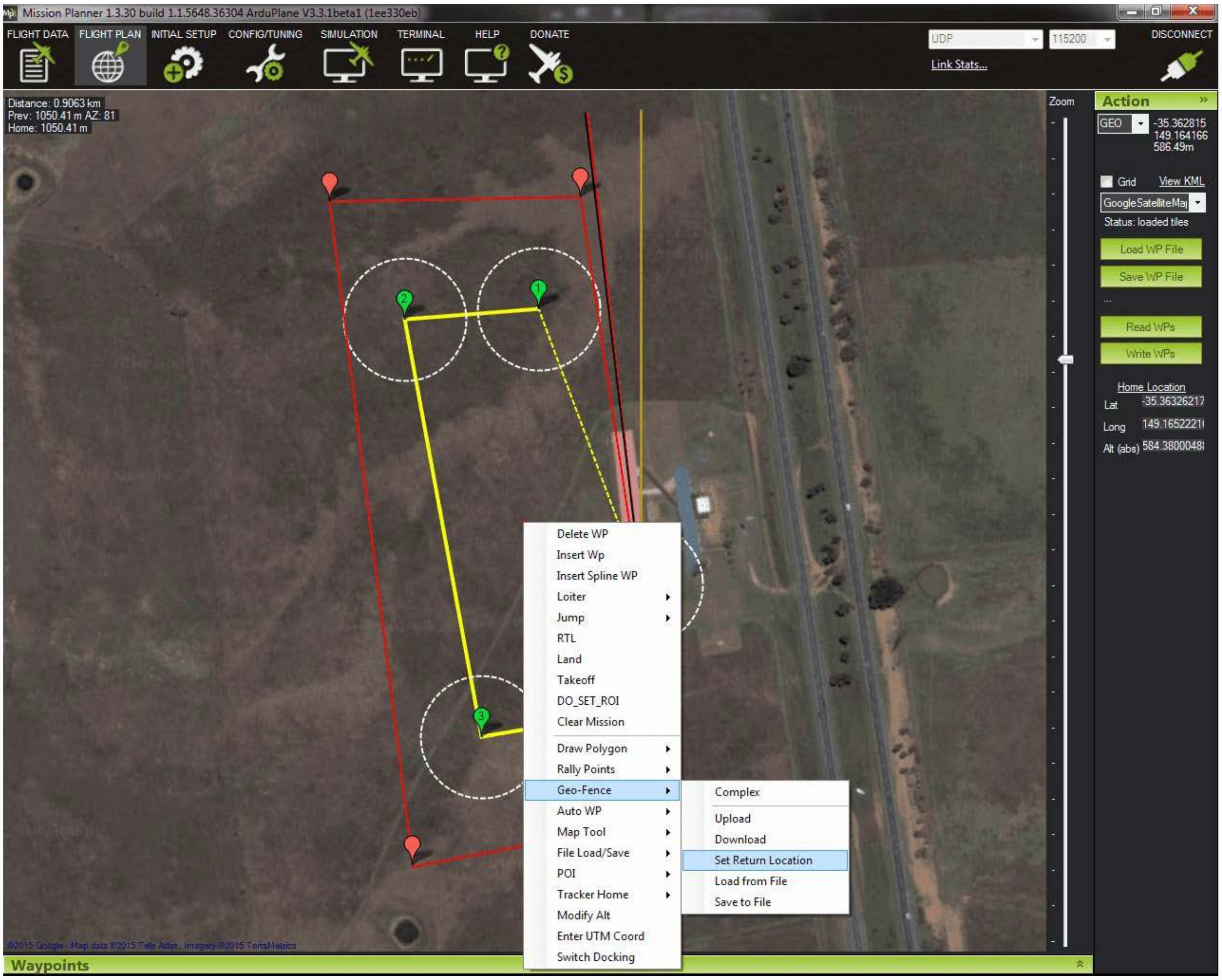
Mission Planner¶

Mission Planner follows a very similar process. Start by right-clicking where you want to begin the geo-fence boundary.



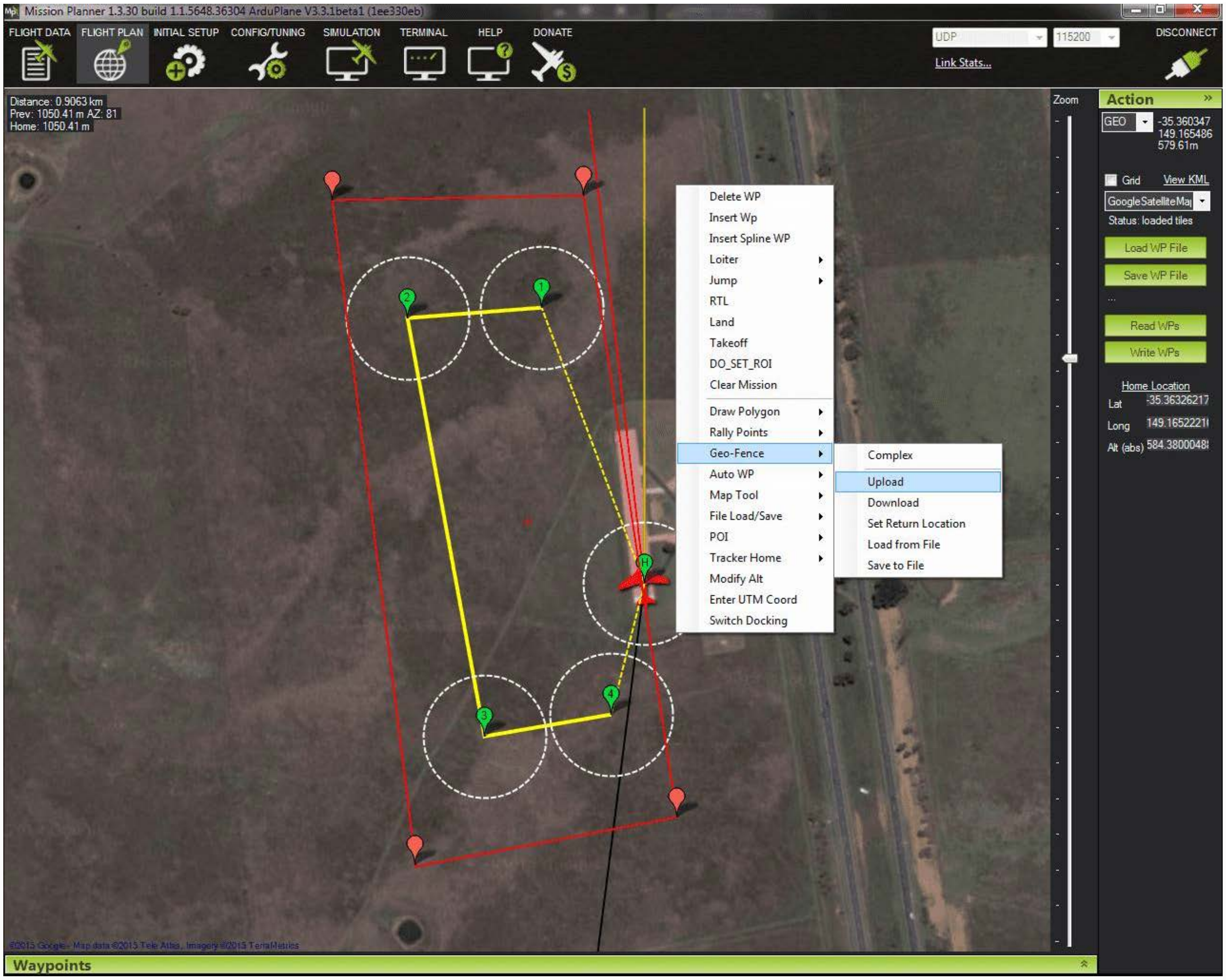
Add Polygon Point

Continue to click on the map where you want the geo-fence boundary and the polygon will appear. You can drag any points you want to adjust. Then right click on the map where you want the plane to return to when a geo-fence breach occurs.



Geo-Fence Set ReturnLocation

Finally upload the geo-fence.



Geo-Fence Upload

Altitude of the return point

If you set the FENCE_RET_RALLY parameter to 1, then the return altitude will be **the same as the loiter altitude of the nearest Rally Point**. If the FENCE_RET_RALLY is set to 0, then you may set the return altitude in meters above the Home Point with the FENCE_RETALT parameter. **Otherwise**:

If you set FENCE_MINALT and FENCE_MAXALT to other than zero (and have FENCE_MAXALT greater than FENCE_MINALT) then the return point altitude will be half way between FENCE_MINALT and FENCE_MAXALT.

If you don't setup FENCE_MINALT and FENCE_MAXALT (ie. leave them at zero) then the return point altitude will be given by the ALT_HOLD_RTL parameter, which is also used for RTL mode. Note that ALT_HOLD_RTL is in centimetres, whereas FENCE_MINALT and FENCE_MAXALT are in meters.

If your flying club and local flying rules don't set a maximum altitude then we recommend you use a maximum altitude of at most 122 meters (which is around 400 feet). Beyond that altitude it becomes quite difficult to keep good eye contact with your model.

With FENCE_MINALT set at 30 meters (to allow for some dive momentum) and FENCE_MAXALT set to 122 meters, the return point will be 76 meters, which is quite a good altitude to leave the plane loitering while you are getting ready to have another go.

Stick-mixing on fence breach

The APM enables ‘stick mixing’ by default when in auto modes. This means that you can change the path of a loiter, for example, by using your transmitter sticks.

When you are using geo-fencing, stick mixing will be disabled on fence breach until your plane is back inside the fenced region. This is to ensure that the bad control inputs that caused you to breach the fence don’t prevent it from recovering to the return point.

As soon as you are back inside the fence stick mixing will be re-enabled, allowing you to control the GUIDED mode that the plane will be in. If by using stick mixing you manage to take the plane outside the fence again then stick mixing will again be disabled until you are back inside the fence.

Tips for flying with geo-fencing

You should have geo-fencing disabled when on the ground and for takeoff. Be careful not to enable it on the ground, as it may declare a fence breach and try to fly to the return point. If flying fully autonomously you may use the FENCE_AUTOENABLE parameter to assist with this complication.

Also remember to disable it for landing, as the altitude breach when you are coming in will make it very hard to land!

If you are using an APM1 and want to combine geo-fencing with MANUAL mode, then remember that on the APM1 the APM software is bypassed when using channel 8 for mode switching and a switch PWM channel value above 1750 (this is called ‘hardware manual’ on the APM1). So you either need to set a different switch position as MANUAL, or use a different mode switch control channel (and set FLTMODE_CH to the channel you are using).

Before you takeoff and fly with geo-fencing make sure all the parameters are setup as described above, and also make sure you have a good GPS lock. If you lose GPS lock then geo-fencing will disable itself until GPS lock is regained, so don’t use it if your GPS signal is marginal.

I’d also recommend you test it gently at first. Try slowly approaching a fence boundary and ensure it correctly ‘bounces’ off the virtual wall and returns to the return point OK. Then after taking control again, try slowly approaching the minimum altitude and ensure it bounces off the FENCE_MINALT you have set.

While developing geo-fencing I found that combining it with MANUAL mode is the most fun. It gives you all of the excitement of manual flight with sharp turns and fancy stunts while saving your plane when you make a mistake.

Example flight

This is the track from a flight with geo-fencing enabled at my local flying club while flying my !SkyWalker. The white lines show the geo-fence boundary, plus you can see the return point in the middle. You can also see the points where the plane breached the geo-fence to the north, west and south. There were also numerous altitude breaches, as I was using this flight to try to improve my inverted flight skills in MANUAL mode. The plane would not have survived without the geo-fence!

Notice that the geo-fence in this example runs along the middle of the runway. This is to conform to my local club rules. The takeoff and landing were done with the fence disabled. I had FENCE_CHANNEL set to 7, and RST_SWITCH_CH set to 6. That allowed me to enable the fence after takeoff using one switch, then to take back control after a breach using the spring loaded trainer switch.

MAVLink support

The APM will report the fence status via the MAVLink GCS protocol. The key status packet is called FENCE_STATUS, and is defined in “ardpilotmega.xml”. A typical FENCE_STATUS packet looks like this:

```
2011-12-20 16:36:35.60: FENCE_STATUS breach_status : 1, breach_count : 15, breach_type : 1, breach_time :
1706506
```

The breach_status field is 0 if inside the fence, and 1 if outside. The breach_count is how many fence breaches you have had on this flight. The breach_type is the type of the last breach (see the FENCE_BREACH enum in ardupilotmega.xml). The breach_time is the time in milliseconds of the breach since APM was booted.

The MAV_SYS_STATUS_GEOFENCE bit of the MAV_SYS_STATUS_SENSOR portion of the SYS_STATUS message indicates whether or not the geo-fence is breached. As of this writing only the MAVProxy GCS recognizes this status bit and reports the status of the geo-fence. In the future the Mission Planner, APM Planner, and other GCS applications should get support for announcing geo-fence status during the flight.

The MAV_CMD_DO_FENCE_ENABLE MAVLink command message allows a GCS to enable or disable a fence interactively. As of this writing only MAVProxy supports this message using the “fence enable” or “fence disable” commands. In the future Mission Planner, APM Planner, and other GCS applications should get support for interactively enabling and disabling the geo-fence without needing to use a manual transmitter.

Advanced Features

Geo-fencing in Plane can also be used as part of a failsafe system, for competitions like the Outback Challenge. For those type of events you should define your fence boundary as usual, but additionally build APM with the FENCE_TRIGGERED_PIN option set in **APM_Config.h**. This option allows you to set a digital pin on your APM to go high when the fence is breached. You can connect this pin to your planes failsafe device to trigger the planes failsafe mode (which for the OBC competition involves setting extreme servo values to dive the plane into the ground).

Channel Output Functions

The first 4 output channels in Plane have specific meanings - typically Aileron, Elevator, Throttle and Rudder. Beyond that you may map the remaining output channels to any of a large number of output functions. On a board like the Pixhawk which has 14 output channels that means you have 10 channels that you can freely assign to any function.

This page describes how to configure these output channels and what each of the available functions is.

The RCn_FUNCTION parameters

In the advanced parameter view of your GCS you will find that each RC output channel beyond channel 4 has a RCn_FUNCTION parameter. For example, RC5_FUNCTION controls the output function of channel 5, RC6_FUNCTION controls the output function of channel 6 and so on.

The values you can set these parameters to are shared with copters and rovers, and not all of them have been implemented on fixed wing aircraft. The ones that are implemented on fixed wing are listed below:

- Disabled=0
- RCPassThru=1
- Flap=2
- Flap_auto=3
- Aileron=4
- mount_pan=6
- mount_tilt=7
- mount_roll=8
- camera_trigger=10
- mount2_pan=12
- mount2_tilt=13
- mount2_roll=14
- DifferentialSpoiler1=16
- DifferentialSpoiler2=17
- AileronWithInput=18
- Elevator=19
- ElevatorWithInput=20
- Rudder=21
- Flaperon1=24
- Flaperon2=25
- GroundSteering=26

The default value for all channels is 0, meaning disabled. That means the channel will output the trim value for that channel (for example, if RC5_FUNCTION is 0 then channel 5 will output RC5_TRIM) unless it is overridden by a mission command.

All of these functions may be used on multiple channels. So if you want 3 more elevator channels for some reason you can set RCn_FUNCTION to 19 on 3 of your output channels. The normal elevator (channel 2) will still work, but you will also have 3 extra elevons, each of which you can trim and set the range of separately.

Disabled

For normal operation, the Disabled output function sets the output value of the channel to the trim value. The exception to this is when a MAVLink override of the channel or a mission servo set is used. So in some ways “disabled” could be called “mission-controlled”.

When you fly an auto mission you can ask for a servo to be set to a value as part of that mission. In that case you should set the RCn_FUNCTION for that channel to Disabled, so that the value doesn’t get changed by another output function immediately after the mission sets the value.

RCPassThru

Setting a channel to RCPassThru means it will output the value that is coming into the board from the corresponding input channel. For example, if RC5_FUNCTION is 1 (meaning RCPassThru) then channel 5 output will always be equal to channel 5 input.

Flap

When a channel is set as a flap it’s value comes from the flap input channel (controlled by the FLAP_IN_CHANNEL parameter). The reason you may want to use this instead of a RCPassThru is that you can setup multiple flap channels with different trims and ranges, and you may want to take advantage of the FLAP_SLEWRATE to limit the speed of flap movement.

Flap_auto

The flap auto output function behaves like the Flap output, except it can also accept automatic flap output from the TKOFF_FLAP_PCNT and LAND_FLAP_PERCNT parameters, as well as the FLAP_1_SPEED, FLAP_1_PERCNT, FLAP_2_SPEED and FLAP_2_PERCNT parameters.

If you have both a FLAP_IN_CHANNEL set and a Flap_auto output function set then the amount of flap applied is the higher of the two.

Aileron

The aileron output function adds additional aileron outputs, with separate per-channel trim and range. This is useful when you want to trim each aileron separately, or if your main aileron is setup as an elevon mixer (using the ELEVON_OUTPUT option), and you also want some normal ailerons.

AileronWithInput

The AileronWithInput function is used where you have setup your transmitter to input the aileron signal you want on this channel on the corresponding input channel for this output. For example, if you set RC6_FUNCTION=18 and have setup your transmitter to send the right aileron signal for manual mode to channel 6 then you can use AileronWithInput. The main difference is in manual mode. In manual with AileronWithInput the output comes directly from the corresponding input channel. With the Aileron output function the output in manual mode is based on the main aileron input channel (usually channel 1) but trimmed and scaled according to the RC6 trim values.

Mount_pan, Mount_tilt and Mount_roll

These control the output channels for controlling a servo gimbal. Please see the camera gimbal configuration documentation for details.

The Mount2_pan, Mount2_tilt and Mount2_roll options are the same, but control a second camera gimbal

Camera_trigger

The Camera_trigger output function is used to trigger a camera with a servo. See the camera gimbal documentation for details.

Elevator

The elevator output function adds additional elevator outputs, with separate per-channel trim and range. This is useful when you want to trim each elevator separately, or if your main elevator is setup as an elevon mixer (using the ELEVON_OUTPUT option), and you also want some normal elevator.

ElevatorWithInput

The ElevatorWithInput function is used where you have setup your transmitter to input the elevator signal you want on this channel on the corresponding input channel for this output. For example, if you set RC6_FUNCTION=20 and have setup your transmitter to send the right elevator signal for manual mode to channel 6 then you can use ElevatorWithInput. The main difference is in manual mode. In manual with ElevatorWithInput the output comes directly from the corresponding input channel. With the Elevator output function the output in manual mode is based on the main elevator input channel (usually channel 2) but trimmed and scaled according to the RC6 trim values.

Rudder

The rudder output function adds additional rudder outputs, with separate per-channel trim and range. Separate rudder channels is particularly useful for nose wheel steering where the nose wheel may need to be reversed as compared to the normal rudder channel or for multi-wheel planes.

GroundSteering

The GroundSteering output function acts much like the rudder output function except that it only acts when the aircraft is below GROUND_STEER_ALT altitude. At altitudes above GROUND_STEER_ALT the output will be the trim value for the channel.

Flaperon1 and Flaperon2

Using the flaperon1 and flaperon2 output functions you can setup flaperons, which are ailerons that double as flaps. They are very useful for aircraft which have ailerons but no flaps.

To use the flaperon output functions you need to also set the FLAPERON_OUTPUT option to the right value (from 1 to 4) for your servo setup. See the Elevon setup page for a more detailed description of how this parameter works (that page is for elevons, but flaperons work in the same manner).

Note that flaperons act like Flap_auto described above for the flap component of the output.

QuadPlane Support

This article explains how to set up and use a combined fixed wing and multicopter aircraft, also known as a “QuadPlane”.

QuadPlane Overview

A QuadPlane is a combined fixed wing and MultiCopter aircraft. This sort of aircraft brings the benefit of vertical takeoff and landing, significantly greater speed and range of travel, and the ability to hover and perform copter-like tasks at the destination.

QuadPlane is built upon Plane, but adds control over between 4 and 8 horizontal rotors. Additional modes and commands allow a QuadPlane to take off, land and fly like a copter, and to smoothly transition between the Plane and Copter-like modes in both automatic and autopilot-assisted modes. The additional rotors can also provide lift and stability in normal Plane modes.

Installing the Firmware

QuadPlane support is in APM:Plane releases from 3.5.0 onwards. The normal instructions for installing the Plane firmware apply.

When you install the plane firmware and look in the parameter list you will see a Q_ENABLE parameter. That defaults to zero, which disables QuadPlane support. Setting Q_ENABLE to 1 will enable QuadPlane support. You then need to refresh your parameter list to see all the other QuadPlane options. All QuadPlane specific parameters start with Q_.

Building a QuadPlane

Putting together a QuadPlane involves careful planning. This page will give you some general guidance on design principles for QuadPlanes that may help you with your build.

General Rules

A wide range of fixed wing aircraft can be converted to have VTOL capabilities. While ArduPilot uses the name QuadPlane for these aircraft, you are not restricted to just QuadCopter motor layouts. Almost any multicopter motor arrangement can be used with a QuadPlane, including quad, hexa, tricoper, octa and octaquad.

Some of the key factors to success are:

- a fixed wing frame that can carry the weight of the multicopter lifting motors and power system, along with any payload needed
- sufficient power in the lifting motors not just for the total airframe weight, but also for the additional load that may be induced by downforce on the wings
- complete clearance above and below the whole disk area of the lifting motors, to ensure they achieve full aerodynamic thrust
- minimum wing twist and flex so the motors provide thrust vertically at all times
- a sufficiently robust mounting system for the lifting motors
- minimising aerodynamic drag from the lifting motors and frame

When you are designing a QuadPlane it is highly recommended that you make use of eCalc to help choose the motors, ESCs, batteries, propellers and other components of your design. Brushless motors vary a lot in their power to weight ratio, and ensuring you choose motors that keep the weight down while supplying sufficient lifting power is important.

QuadPlane Range

Perhaps surprisingly, it is sometimes possible to increase the potential range of an aircraft using a QuadPlane conversion. This may seem counter intuitive as a QuadPlane conversion will both add weight and increase aerodynamic drag to an airframe.

The reason why range can be increased is the extra carrying capacity of a QuadPlane. Many fixed wing aircraft are limited in the amount of battery they can carry due to the requirements for reliable launch. During launch, and especially when using a flying launch such as a catapult or bungee, the aircraft needs to rapidly accelerate to an airspeed above its stall speed. If it fails to reach that speed suffiently quickly then it will crash. A QuadPlane avoids this problem by taking off vertically, and can spend longer on the acceleration needed to sufficient speed for forward flight.

This means it is often possible to pack a lot more battery into a QuadPlane than is possible in the same airframe without VTOL motors. The extra battery capacity can more than make up for the increased weight and drag of the VTOL motors.

To make the most of this advantage you need to do very rapid VTOL takeoffs and landings to minimise the battery consumption in VTOL flight. The video below demonstrates just how rapid a takeoff can be achieved with a properly setup quadplane.

A second factor that can help with QuadPlane range is the flexibility available in choosing the propeller and power train for the forward motor. As conventional takeoff is not needed the forward motor does not need to be optimised for the high level of thrust needed for takeoff. This can allow larger propellers and geared motors to be used that are highly efficient for forward cruise flight.

Finally, for really long range with a QuadPlane you can use an internal combustion engine for the forward motor. A gas engine can run for a lot longer than an electric motor with the same weight of fuel.

Build Logs

Here are some build logs of a few QuadPlanes that may help you with ideas for your own build.

- Porter OctaQuadPlane build: http://diydrones.com/profiles/blogs/building-flying-and-not-crashing-a-large-octaquadplane
- Porter QuadPlane build: http://diydrones.com/profiles/blogs/building-flying-and-crashing-a-large-quadplane
- QuadRanger build: http://px4.io/docs/quadranger-vtol/

if you would like to add your own build to this list then please contact the ArduPilot dev team.

QuadPlane Frame setup

The QuadPlane code supports several frame arrangements of quadcopter, hexacopter, octacopter and octaquad multicopter frames.

The motor order and output channel is the same as for copter (see Copter motor layout) except that the output channel numbers start at 5 instead of 1.

For example, with the default Quad-X frame the motors are on outputs 5 to 8. The arrangement is:

- **Channel 5:** Front right motor, counter-clockwise
- **Channel 6:** Rear left motor, counter-clockwise
- **Channel 7:** Front left motor, clockwise
- **Channel 8:** Rear right motor, clockwise

You can remember the clockwise/counter-clockwise rule by “motors turn in towards the fuselage”, except for the H configuration, there all directions are inverted!

Another common setup is an octa-quad, which uses the following ordering

- **Channel 5:** Front right top motor, counter-clockwise
- **Channel 6:** Front left top motor, clockwise

- **Channel 7:** Rear left top motor, counter-clockwise
- **Channel 8:** Rear right top motor, clockwise
- **Channel 9:** Front left bottom motor, counter-clockwise
- **Channel 10:** Front right bottom motor, clockwise
- **Channel 11:** Rear right bottom motor, counter-clockwise
- **Channel 12:** Rear left bottom motor, clockwise

You can remember the clockwise/counter-clockwise rule for an octa-quad by “top motors turn in towards the fuselage, bottom motors turn out away from the fuselage”.

The normal plane outputs are assumed to be on 1 to 4 as usual. Only vertical lift outputs (5 to 8 on a quad setup) run at high PWM rate (400Hz). In a quad setup you can also use channels 9 to 14 in any way you like, just as with the normal Plane code.

You can optionally move the quad motors to be on any other channel above 4, using the procedure outlined below.

To use a different frame type you can set Q_FRAME_CLASS and Q_FRAME_TYPE. Q_FRAME_CLASS can be:

- 0 for quad
- 1 for hexa
- 2 for octa
- 3 for octaquad

Within each of these frame classes the Q_FRAME_TYPE chooses the motor layout

- 0 for plus frame
- 1 for X frame
- 2 for V frame
- 3 for H frame

Using different channel mappings

You can remap what output channels the quad motors are on by setting values for RCn_FUNCTION. This follows the same approach as other output functions.

Note

Note that you do not need to set any of the RCn_FUNCTION values unless you have a non-standard motor ordering. It is highly recommended that you use the standard ordering and do not set the RCn_FUNCTION parameters, leaving them at zero. They will be automatically set to the right values for your frame on boot.
The output function numbers are:

- 33: motor1
- 34: motor2

- 35: motor3
- 36: motor4
- 37: motor5
- 38: motor6
- 39: motor7
- 40: motor8

So to put your quad motors on outputs 9 to 12 (the auxillary channels on a Pixhawk) you would use these settings in the advanced parameter list:

- RC9_FUNCTION = 33
- RC10_FUNCTION = 34
- RC11_FUNCTION = 35
- RC12_FUNCTION = 36

ESC calibration

Most models of PWM based ESC need to be calibrated to ensure that all the ESCs respond to the same input with the same speed. To calibrate them they need to receive maximum PWM input when initially powered on, then receive minimum PWM input when they have beeped to indicate that the maximum has registered.

The method for calibrating the ESCs on an ArduPilot QuadPlane depends on what version of the firmware you have loaded. The old method is used on versions prior to the 3.6.0 release. The new method is used on 3.6.0 and later.

Warning

You must remove all propellers from your vehicle before doing any ESC calibration. Calibrating with propellers installed is dangerous.

ESC Calibration Procedure (3.6.0 and later)

This process uses the Q_ESC_CAL parameter to enable ESC calibration in QSTABILIZE mode. There are two modes of operation available:

1. with Q_ESC_CAL=1 the output to the motors will come directly from the throttle stick in QSTABILIZE mode when the vehicle is armed
2. with Q_ESC_CAL=2 the output to the motors will be full throttle when the motors are armed

The process when using Q_ESC_CAL=1 is

1. remove your propellers for safety
2. power up just the flight board and not your motors. If you don’t have the ability to isolate power to the ESCs when on battery power then power up your flight board on USB power
3. set the Q_ESC_CAL parameter to 1

4. change to QSTABILIZE mode
5. set the safety switch off to activate the outputs
6. arm your aircraft. The PWM output on all quad motors will now be controlled by your throttle stick
7. move the throttle stick to maximum
8. add power to your ESCs by connecting the battery
9. wait for the ESCs to beep to indicate they have registered the maximum PWM
10. lower the throttle stick to zero and disarm your aircraft
11. you should hear a beep from your ESCs to indicate they have registered the throttle range

The process when using Q_ESC_CAL=2 is

1. remove your propellers for safety
2. power up just the flight board and not your motors. If you don’t have the ability to isolate power to the ESCs when on battery power then power up your flight board on USB power
3. set the Q_ESC_CAL parameter to 2
4. change to QSTABILIZE mode
5. set the safety switch off to activate the outputs
6. arm your aircraft. The PWM output on all quad motors will now be at maximum
7. add power to your ESCs by connecting the battery
8. wait for the ESCs to beep to indicate they have registered the maximum PWM
9. disarm your aircraft
10. you should hear a beep from your ESCs to indicate they have registered the throttle range

Note that using Q_ESC_CAL=1 can be useful for testing your motors response. This is the only mode when you are able to directly control the throttle level on all your motors at once. While in this mode you can use a laser tachometer to test your motor speeds at different throttle levels if you have one.

Old ESC Calibration Procedure (3.5.3 and earlier)

1. remove your propellers for safety
2. power up just the flight board and not your motors. If you don’t have the ability to isolate power to the ESCs when on battery power then power up your flight board on USB power
3. set both the parameters Q_M_SPIN_ARMED and Q_THR_MID to 1000. This sets the PWM output when armed at zero throttle to full power
4. set the safety switch off to activate the outputs
5. arm your aircraft. The PWM output on all quad motors will now climb to maximum.
6. add power to your ESCs by connecting the battery
7. wait for the ESCs to beep to indicate they have registered the maximum PWM
8. disarm your aircraft. The ESCs should beep again indicating they have registered minimum PWM

Now set the Q_M_SPIN_ARMED and Q_THR_MID parameters back to the correct values. A value of 50 for Q_M_SPIN_ARMED is a reasonable starting point. For Q_THR_MID a value of between 500 and 600 is good depending on the power of your motors

QuadPlane Parameter setup

All QuadPlane specific parameters start with a “Q_” prefix. The parameters are very similar to the equivalent Copter parameters so if you are familiar with those you should find setting up a QuadPlane is easy.

Key parameters are:

- To enable QuadPlane functionality you need to set the Q_ENABLE parameter to 1 and then refresh the parameter list
- The Q_THR_MIN_PWM and Q_THR_MAX_PWM parameters used to set the PWM range of the quad motors (this allows them to be different from the range for the forward motor). These need to be set to the range your ESCs expect.
- The most critical tuning parameters are Q_A_RAT_RLL_P and Q_A_RAT_PIT_P. These default to 0.25 but you may find significantly higher values are needed for a QuadPlane.
- The Q_M_SPIN_ARMED parameter is important for getting the right level of motor output when armed in a quad mode
- It is recommended that you set ARMING_RUDDER to 2 to allow for rudder disarm. Alternatively you could have MANUAL as one of your available flight modes (as that will shut down the quad motors). Please be careful not to use hard left rudder and zero throttle while flying or you risk disarming your motors.
- The Q_THR_MID parameter is important for smooth transitions. It defaults to 500 which means 50% throttle for hover. If your aircraft needs more or less than 50% throttle to hover then please adjust this. That will prevent a throttle surge during transition as the altitude controller learns the right throttle level

Note

The QuadPlane code requires GPS lock for proper operation. This is inherited from the plane code, which disables intertial estimation of attitude and position if GPS lock is not available. Do not try to fly a QuadPlane indoors. It will not fly well

QuadPlane Flight modes

The QuadPlane code is based upon the Plane master firmware but with 5 extra modes:

- mode 17: QSTABILIZE (like Copter STABILIZE)
- mode 18: QHOVER (like Copter ALT_HOLD)
- mode 19: QLOITER (like Copter LOITER)
- mode 20: QLAND (like Copter LAND)
- mode 21: QRTL (like Copter RTL)

Tip

You may probably need to set the `FLTMODE*` parameters for these extra modes as numeric values if your GCS doesn’t understand these values yet.
If you are familiar with the equivalent Copter flight modes then you should be comfortable flying a QuadPlane. The only real difference comes during transition between fixed wing and QuadPlane flight,

which is described below.

Note

The landing detection logic in QLOITER mode is not as sophisticated as the landing detection logic in Copter, so if you get GPS movement while on the ground in QLOITER mode then the aircraft may try to tip over as it tries to hold position while in contact with the ground. It is suggested that you switch to QHOVER or QSTABILIZE once landed as these are unaffected by GPS movement.

Flight modes to avoid

The linked nature of of the vertical lift and fixed wing control in quadplanes means the autopilot always needs to know what the pilot is trying to do in terms of speed and attitude. For this reason you should avoid the following flight modes in a quadplane:

- ACRO
- STABILIZE
- TRAINING

these modes are problematic as the stick input from the pilot is not sufficient to tell the autopilot what attitude the aircraft wants or what climb rate is wanted, so the quadplane logic does not engage the quad motors when in these modes. These modes also make log analysis difficult. Please use FBWA mode instead of STABILIZE for manual flight.

In the future we may adds ways to use the quad motors in these modes, but for now please avoid them.

The other mode where the quad motors are disabled is MANUAL mode. That mode still can be useful for checking aircraft trim in fixed wing flight or for taxiing your aircraft.

Flying a QuadPlane

While flying a QuadPlane can actually be easier than flying a conventional fixed wing aircraft there are some things you need to understand. Please read the following sections carefully.

Transition

QuadPlane transition is where the aircraft is changing between flying primarily as a VTOL (copter-like) aircraft and flying as a conventional fixed wing aircraft. Transition happens in both directions, and can either be commanded by the pilot or happen automatically based on airspeed and flight mode.

The primary way to initiate a transition is to change flight mode, either using the flight mode channel on your transmitter or using a ground station to command a mode change.

- If you transition to MANUAL then the quad motors will immediately stop.
- If you transition to any other fixed wing mode then the quad will continue to supply lift and stability until you have reached the ARSPD_FBW_MIN airspeed (or airspeed estimate if no airspeed sensor).

- Once that airspeed is reached the quad motors will slowly drop in power over Q_TRANSITION_MS milliseconds (default is 5000, so 5 seconds) and will switch off after that

If you transition from a fixed wing mode to a QuadPlane mode then the forward motor will immediately stop, but the control surfaces will continue to provide stability while the plane slows down. This allows for transitions to QuadPlane modes while flying at high speed.

The one exception to the forward motor stopping in QuadPlane VTOL modes is if you have the Q_VFWD_GAIN parameter set to a non-zero value. In that case the forward motor will be used to hold the aircraft level in a wind. See the description of Q_VFWD_GAIN.

Note

If you transition to QLOITER or QLAND while flying at high speed then the loiter code will try to bring the aircraft to a very rapid stop which will cause the plane to pitch up hard and then fly backwards to get back to the point where QLOITER was entered. Unless you are sure of the strength of your airframe it would be a good idea to transition to QHOVER first which will result in a much gentler transition, then move to QLOITER once the aircraft has slowed down.

Assisted fixed-wing flight

The QuadPlane code can also be configured to provide assistance to the fixed wing code in any flight mode except MANUAL. To enable quad assistance you should set Q_ASSIST_SPEED parameter to the airspeed below which you want assistance.

When Q_ASSIST_SPEED is non-zero then the quad motors will assist with both stability and lift whenever the airspeed drops below that threshold. This can be used to allow flying at very low speeds in FBWA mode for example, or for assisted automatic fixed wing takeoffs.

It is suggested that you do initial flights with Q_ASSIST_SPEED set to zero just to test the basic functionality and tune the airframe. Then try with Q_ASSIST_SPEED above plane stall speed if you want that functionality.

From the 3.7.0 release an additional assistance type is available based on attitude error. If Q_ASSIST_ANGLE is non-zero then this parameter gives an attitude error in degrees above which assistance will be enabled even if the airspeed is above Q_ASSIST_SPEED. The attitude assistance will only be used if Q_ASSIST_SPEED greater than zero.

What assistance the quad motors provides depends on the fixed wing flight mode. If you are flying in an autonomous or semi-autonomous mode then the quad motors will try to assist with whatever climb rate and turn rate the autonomous flight mode wants when assistance is enabled (ie. airspeed is below Q_ASSIST_SPEED or attitude error is above Q_ASSIST_ANGLE). In a manually navigated mode the quad will try to provide assistance that fits with the pilot inputs.

The specific handling is:

- In AUTO mode the quad will provide lift to get to the altitude of the next waypoint, and will help turn the aircraft at the rate the navigation controller is demanding.

- In fixed wing LOITER, RTL or GUIDED modes the quad motors will try to assist with whatever climb rate and turn rate the navigation controller is asking for.
- In CRUISE or FBWB mode the quad will provide lift according to the pilots demanded climb rate (controlled with pitch stick). The quad motors will try to turn at the pilot demanded turn rate (combining aileron and rudder input).
- In FBWA mode the quad will assume that pitch stick input is proportional to the climb rate the user wants. So if the user pulls back on the pitch stick the quad motors will try to climb, and if the user pushes forward on the pitch stick the quad motors will try to provide a stable descent.
- In AUTOTUNE mode the quad will provide the same assistance as in FBWA, but it is not a good idea to use AUTOTUNE mode with a high value of Q_ASSIST_SPEED as the quad assistance will interfere with the learning of the fixed wing gains.
- In MANUAL, ACRO and TRAINING modes the quad motors will completely turn off. In those modes the aircraft will fly purely as a fixed wing.
- In STABILIZE mode the quad motors will try to provide lift if assistance is turned on.

Return to Launch (RTL)

When flying a QuadPlane you have a choice of several methods of handling return to launch. The choices are:

- circle about the return point as a fixed wing
- fly as a VTOL aircraft to the return point then land vertically
- fly as a fixed wing aircraft until close to the return point then switch to VTOL and land vertically

In each case a key concept is the return point. This is defined as the closest rally point, or if a rally point is not defined then the home location. See the Rally Points page for more information on rally points.

Fixed Wing RTL

The default behaviour of the RTL mode is the same as for fixed wing. It will fly to the nearest rally point (or home if no rally point is defined) and circle as a fixed wing aircraft about that point. The VTOL motors will not be used unless the aircraft drops below the airspeed defined in Q_ASSIST_SPEED. The altitude the aircraft will circle at will be the altitude in the rally point, or the ALT_HOLD_RTL altitude if a rally point is not being used.

VTOL RTL (QRTL)

If you prefer to do return to launch as a VTOL aircraft (like a multirotor would do) then you can use the QRTL flight mode. That flight mode will transition to VTOL flight and then fly at the Q_WP_SPEED speed towards the return point, at an altitude of Q_RTL_ALT.

Once the return point is reached the aircraft will start a vertical descent towards the ground for landing. The initial descent rate is set by Q_WP_SPEED_DN. Once the aircraft reaches an altitude of Q_LAND_FINAL_ALT then the descent rate will change to Q_LAND_SPEED for the final landing phase.

In the final landing phase the aircraft will detect landing by looking for when the VTOL motor throttle drops below a minimum threshold for 5 seconds. When that happens the aircraft will disarm and the VTOL motors will stop.

Hybrid RTL

The final option for RTL in a QuadPlane is to fly as a fixed wing aircraft until it is close to the return point at which time it switches to a VTOL RTL as described above. To enable this type of hybrid RTL mode you need to set the Q_RTL_MODE parameter to 1.

The initial altitude that will be aimed for in the fixed wing portion of the hybrid RTL is the same as for a fixed wing RTL. You should set your rally point altitude and ALT_HOLD_RTL options appropriately to ensure that the aircraft arrives at a reasonable altitude for a vertical landing. A landing approach altitude of about 15 meters is good for many QuadPlanes. This should be greater than or equal to the Q_RTL_ALT values.

The distance from the return point at which the aircraft switches from fixed wing to VTOL flight is set using the RTL_RADIUS parameter, or if that is not set then the WP_LOITER_RAD parameter is used. The aircraft will then slow down as it approaches the return point, aiming for an altitude set by Q_RTL_ALT.

Once the return point is reached the aircraft begins to descend and land, exactly as described in the VTOL RTL mode above.

What will happen?

Understanding hybrid aircraft can be difficult at first, so below are some scenarios and how the ArduPilot code will handle them.

I am hovering in QHOVER and switch to FBWA mode

The aircraft will continue to hover, waiting for pilot input. If you take your hands off the sticks at zero throttle the aircraft will continue to hold the current height and hold itself level. It will drift with the wind as it is not doing position hold.

If you advance the throttle stick then the forward motor will throttle-up and the aircraft will start to move forward. The quad motors will continue to provide both lift and stability while the aircraft is moving slowly. You can control the attitude of the aircraft with roll and pitch stick input. When you use the pitch stick (elevator) that will affect the climb rate of the quad motors. If you pull back on the elevator the quad motors will assist with the aircraft climb. If you push forward on the pitch stick the power to the quad motors will decrease and the aircraft will descend.

The roll and pitch input also controls the attitude of the aircraft, so a right roll at low speed will cause the aircraft to move to the right. It will also cause the aircraft to yaw to the right (as the QuadPlane code interprets right aileron in fixed wing mode as a commanded turn).

Once the aircraft reaches an airspeed of ARSPD_FBW_MIN (or Q_ASSIST_SPEED if that is set and is greater than ARSPD_FBW_MIN) the amount of assistance the quad motors provide will decrease over 5 seconds. After that time the aircraft will be flying purely as a fixed wing.

I am flying fast in FBWA mode and switch to QHOVER mode

The quad motors will immediately engage and will start by holding the aircraft at the current height. The climb/descent rate is now set by the throttle stick, with a higher throttle stick meaning climb and a lower throttle stick meaning descend. At mid-stick the aircraft will hold altitude.

The forward motor will stop, but the aircraft will continue to move forward due to its momentum. The drag of the air will slowly bring it to a stop. The attitude of the aircraft can be controlled with roll and pitch sticks (aileron and elevator). You can yaw the aircraft with rudder.

I am flying fast in FBWA mode and switch to QLOITER mode

The quad motors will immediately engage and the aircraft will pitch up hard, as it tries to hold position at the position it was in when you switched to QLOITER mode.

The aircraft will stop very quickly, and will back up slightly to the position where QLOITER was entered. The movement of the aircraft can be controlled with roll and pitch sticks (aileron and elevator). You can yaw the aircraft with rudder.

The climb/descent rate is now set by the throttle stick, with a higher throttle stick meaning climb and a lower throttle stick meaning descend. At mid-stick the aircraft will hold altitude.

I switch to RTL mode while hovering

The aircraft will transition to fixed wing flight. The quad motors will provide assistance with lift and attitude while the forward motor starts to pull the aircraft forward.

The normal Plane RTL flight plan will then be run, which defaults to circling at the RTL altitude above the arming position or nearest rally point. If you have RTL_AUTOLAND setup then the aircraft will do a fixed wing landing.

If you set Q_RTL_MODE to 1 then the aircraft will switch to a VTOL landing when it gets close to return point.

Typical flight

A typical test flight would be:

- takeoff in QLOITER or QHOVER
- switch to FBWA mode and advance throttle to start flying fixed wing
- switch to QHOVER mode to go back to quad mode.

Weathervaning and Wind Hold

Flying a QuadPlane in significant levels of wind can present a challenge. The issue is that the large wing surface offers a lot of surface area for the wind to interact with. That can lead to a reduction in attitude and position control and high motor and ESC load.

To reduce the impact of wind when flying in VTOL modes the ArduPilot QuadPlane code supports two features:

- Active weathervaning
- Position hold using forward motor

Together these two features can greatly reduce the impact of wind on VTOL flight by keeping the aircraft pointed into the wind and reducing the area of the wing exposed to the wind.

Active Weathervaning

Active weathervaning acts to turn the nose of the aircraft into the wind when flying in position-controlled VTOL modes. You can enable active weathervaning by setting the Q_WVANE_GAIN parameter to a non-zero value. The default is not to use active weathervaning.

The way it works is the autopilot looks at the roll attitude needed to control the desired position. The basic algorithm is “turn into the roll”. If the aircraft needs to roll to the right in order to hold position then it will turn in that direction on the assumption that the right roll is needed in order to hold against the wind.

How quickly the aircraft turns is given by the Q_WVANE_GAIN parameter. A good value to start with is 0.1. Higher values will make the aircraft turn into the roll more quickly. If the value is too high then you can get instability and oscillation in yaw.

To cope with a small amount of trim in the aircraft there is an additional parameter Q_WVANE_MINROLL which controls the minimum roll level before weathervaning will be used. This defaults to one degree. If you find your aircraft starts yawing even in no wind then you may need to raise this value.

Active weathervaning is only active in VTOL modes, and VTOL sections of AUTO modes (such as VTOL takeoff and VTOL landing). It is not active in QSTABILIZE and QHOVER modes as those are not position controlled modes. It is active is QLOITER, QLAND and QRTL modes.

Using the Forward Motor

In addition to active weathervaning, the QuadPlane code supports using the forward motor to hold the pitch level in VTOL flight modes. To enable use of the forward motor for position hold you need to set the Q_VFWD_GAIN parameter to a non-zero value.

The way it works is to look at two factors:

- the navigation attitude pitch of the aircraft

- the difference between the desired forward velocity and the actual forward velocity

These are combined with the Q_VFWD_GAIN to ramp up and down the throttle on the forward motor in order to minimize the attitude pitch of the aircraft. That keeps the area of wing exposed to the wind minimized which can reduce VTOL motor load.

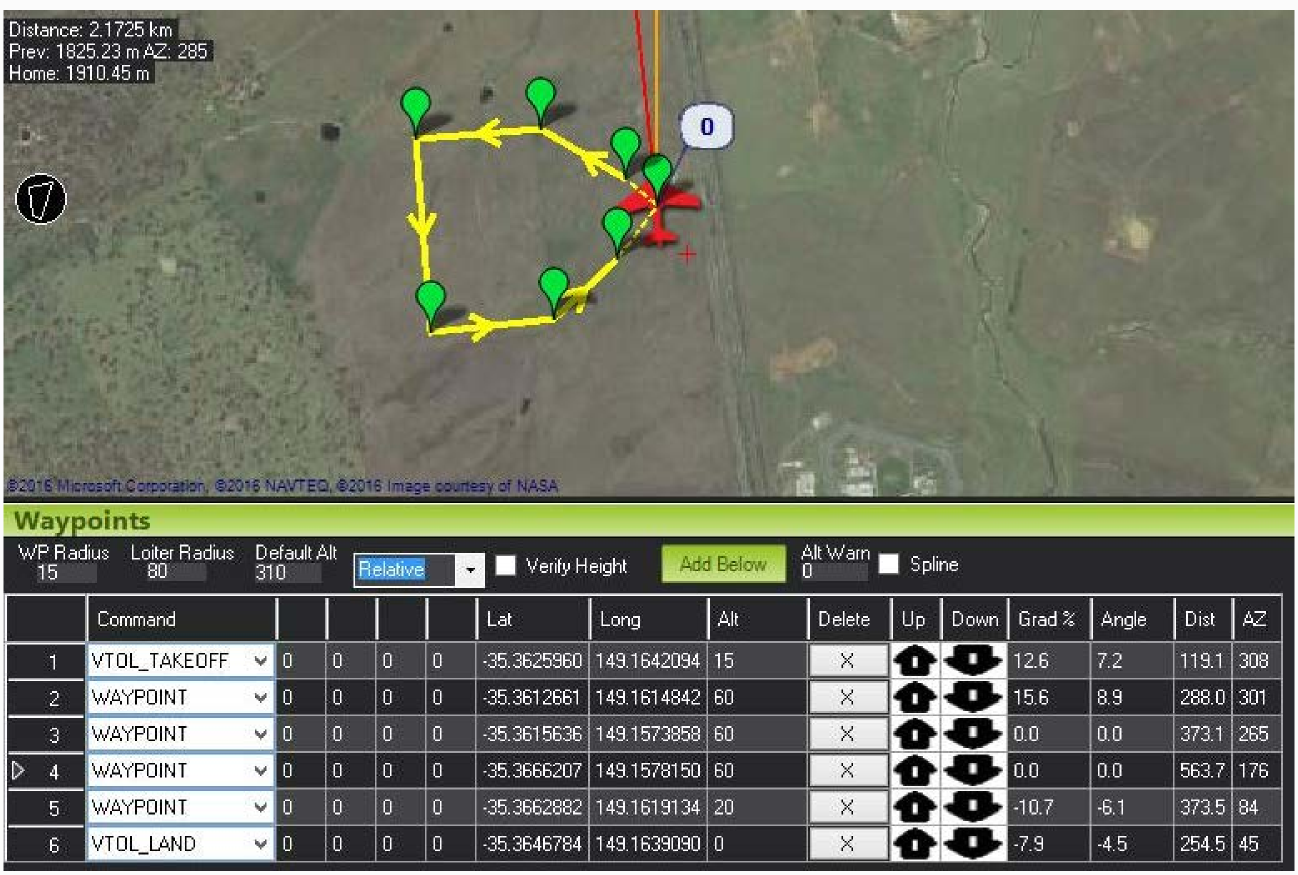
A good value to start with for Q_VFWD_GAIN is 0.05. Higher values will use the forward motor more aggressively. If the value is too high you can get severe pitch oscillations.

Note that you can also use reverse thrust on the forward motor. If your THR_MIN parameter is less than zero then reverse thrust is available and the motor will use reverse thrust to slow down or move backwards as needed. See the reverse thrust section in the automatic landing documentation for more details.

As with active weathervaning, using the forward motor is only enabled in position controlled VTOL modes. This means it is not enabled in QSTABILIZE or QHOVER flight modes. It is available in QLOITER, QRTL, QLAND and in AUTO mode when executing VTOL flight commands.

QuadPlane AUTO Missions

You can ask the QuadPlane code to fly AUTO missions, with everything from automatic vertical takeoff, to mixtures of fixed wing and VTOL waypoints and automated VTOL landings. The result is an extremely versatile aircraft capable of long range missions with vertical takeoff and landing.



	Command					Lat	Long	Alt	Delete	Up	Down	Grad %	Angle	Dist	AZ
1	VTOL_TAKEOFF	0	0	0	0	-35.3625960	149.1642094	15	X			12.6	7.2	119.1	308
2	WAYPOINT	0	0	0	0	-35.3612661	149.1614842	60	X			15.6	8.9	288.0	301
3	WAYPOINT	0	0	0	0	-35.3615636	149.1573858	60	X			0.0	0.0	373.1	265
4	WAYPOINT	0	0	0	0	-35.3666207	149.1578150	60	X			0.0	0.0	563.7	176
5	WAYPOINT	0	0	0	0	-35.3662882	149.1619134	20	X			-10.7	-6.1	373.5	84
6	VTOL_LAND	0	0	0	0	-35.3646784	149.1639090	0	X			-7.9	-4.5	254.5	45

AUTO VTOL Takeoff

The most common use of VTOL mission commands in a QuadPlane is an automatic VTOL takeoff. To use a VTOL takeoff you plan your auto mission as usual with your ground stations mission editor, but instead of a NAV_TAKEOFF command for a fixed wing takeoff you instead use a NAV_VTOL_TAKEOFF command for a VTOL takeoff.

The only parameter to a NAV_VTOL_TAKEOFF is the altitude above the takeoff point where the takeoff is complete. Once that altitude is reached the aircraft will move to the next waypoint, transitioning to fixed wing flight as needed. The latitude and longitude of the NAV_VTOL_TAKEOFF command is ignored.

AUTO VTOL Landing

There are several ways to perform an automatic VTOL landing. The simplest is to include a NAV_VTOL_LAND command in your mission. That command should use an altitude of zero, and have a latitude and longitude of the landing position.

When using NAV_VTOL_LAND it is important to have the right horizontal spacing between that waypoint and the previous one. As soon as the aircraft starts on the NAV_VTOL_LAND waypoint it will transition to VTOL flight, which means it will start flying much more slowly than it does in fixed wing flight. So you need to put the previous waypoint the right distance from the landing point. If it is too far from the landing point then the aircraft will spend a lot of time in VTOL flight which will waste battery. If it is too close to the landing point then it will have to stop very abruptly in order to land.

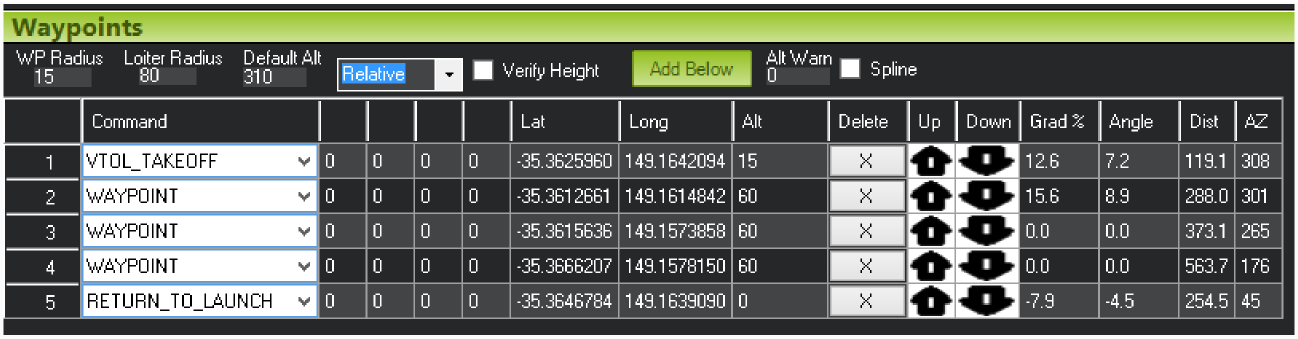
For most small QuadPlanes a distance of between 60 and 80 meters from the last waypoint to the landing point is good. For larger faster flying QuadPlanes you will need a larger distance.

Also make sure the altitude of the last waypoint is chosen to be within a reasonable height of the landing. The VTOL landing approach will be done at whatever height the aircraft is at when it starts on the NAV_VTOL_LAND waypoint. So you would typically want the previous waypoint to have an altitude of about 20 meters above the ground.

Return to Launch

An alternaive to using a NAV_VTOL_LAND command is to use a RETURN_TO_LAUNCH command, and to set the Q_RTL_MODE parameter to 1.

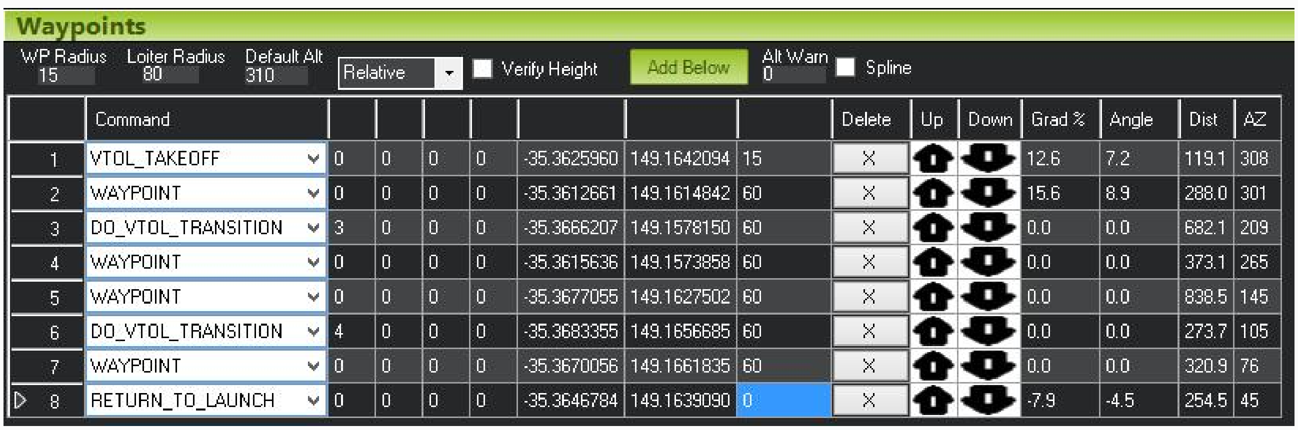
The advantage of using a RETURN_TO_LAUNCH with Q_RTL_MODE set is that the aircraft will automatically use fixed wing flight until it gets within RTL_RADIUS of the return point. That makes it easier to plan missions with a VTOL landing from anywhere in the flying area.



	Command					Lat	Long	Alt	Delete	Up	Down	Grad %	Angle	Dist	AZ
1	VTOL_TAKEOFF	0	0	0	0	-35.3625960	149.1642094	15	X			12.6	7.2	119.1	308
2	WAYPOINT	0	0	0	0	-35.3612661	149.1614842	60	X			15.6	8.9	288.0	301
3	WAYPOINT	0	0	0	0	-35.3615636	149.1573858	60	X			0.0	0.0	373.1	265
4	WAYPOINT	0	0	0	0	-35.3666207	149.1578150	60	X			0.0	0.0	563.7	176
5	RETURN_TO_LAUNCH	0	0	0	0	-35.3646784	149.1639090	0	X			-7.9	-4.5	254.5	45

Mixing VTOL and Fixed Wing Flight

To mix fixed wing and VTOL flight in one mission you can use the DO_VTOL_TRANSITION command in your mission. A DO_VTOL_TRANSITION command takes a single parameter. If the parameter is set to 3 then the aircraft will change to VTOL mode. If the parameter is set to 4 then it will change to fixed wing mode.

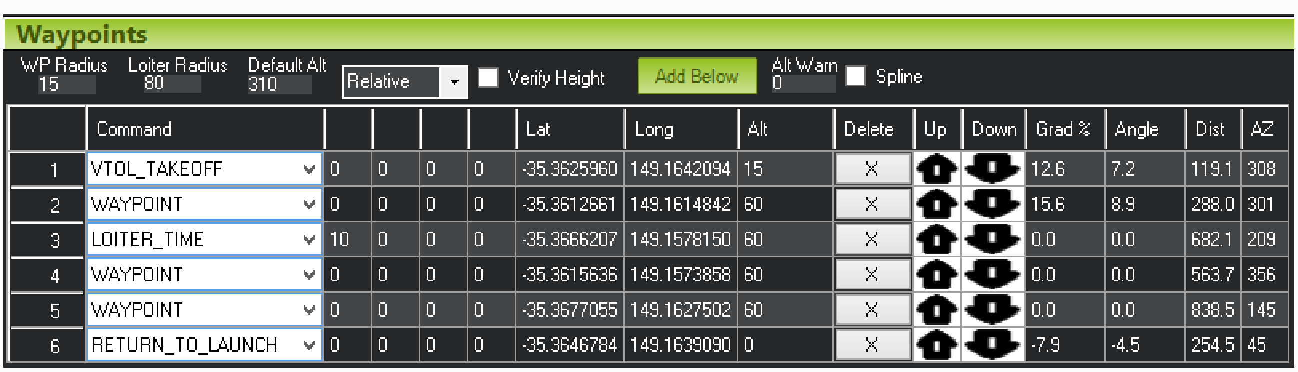


	Command								Delete	Up	Down	Grad %	Angle	Dist	AZ
1	VTOL_TAKEOFF	0	0	0	0	-35.3625960	149.1642094	15	X			12.6	7.2	119.1	308
2	WAYPOINT	0	0	0	0	-35.3612661	149.1614842	60	X			15.6	8.9	288.0	301
3	DO_VTOL_TRANSITION	3	0	0	0	-35.3666207	149.1578150	60	X			0.0	0.0	682.1	209
4	WAYPOINT	0	0	0	0	-35.3615636	149.1573858	60	X			0.0	0.0	373.1	265
5	WAYPOINT	0	0	0	0	-35.3677055	149.1627502	60	X			0.0	0.0	838.5	145
6	DO_VTOL_TRANSITION	4	0	0	0	-35.3683355	149.1656685	60	X			0.0	0.0	273.7	105
7	WAYPOINT	0	0	0	0	-35.3670056	149.1661835	60	X			0.0	0.0	320.9	76
8	RETURN_TO_LAUNCH	0	0	0	0	-35.3646784	149.1639090	0	X			-7.9	-4.5	254.5	45

In the above example the aircraft will do a VTOL takeoff, then it will fly to waypoint 1 as a fixed wing aircraft. It will then switch to VTOL mode and fly as a VTOL aircraft through waypoints 4 and 5, then it will switch back to fixed wing flight to reach waypoint 7, before finally flying home and landing as a VTOL aircraft (assuming Q_RTL_MODE is set to 1).

Hovering in a Mission

By setting the Q_GUIDED_MODE parameter to 1 your quadplane will handle loiter commands in GUIDED mode and in AUTO missions as a VTOL aircraft. For example, the following mission:



	Command					Lat	Long	Alt	Delete	Up	Down	Grad %	Angle	Dist	AZ
1	VTOL_TAKEOFF	0	0	0	0	-35.3625960	149.1642094	15	X			12.6	7.2	119.1	308
2	WAYPOINT	0	0	0	0	-35.3612661	149.1614842	60	X			15.6	8.9	288.0	301
3	LOITER_TIME	10	0	0	0	-35.3666207	149.1578150	60	X			0.0	0.0	682.1	209
4	WAYPOINT	0	0	0	0	-35.3615636	149.1573858	60	X			0.0	0.0	563.7	356
5	WAYPOINT	0	0	0	0	-35.3677055	149.1627502	60	X			0.0	0.0	838.5	145
6	RETURN_TO_LAUNCH	0	0	0	0	-35.3646784	149.1639090	0	X			-7.9	-4.5	254.5	45

the aircraft will pause while hovering for 10 seconds at waypoint 3. It will fly the rest of the mission as a fixed wing aircraft. This can be very useful for getting good photographs of a number of locations in a mission while flying most of the mission as an efficient fixed wing aircraft.

Guided Mode

In addition to AUTO mode, you can also use a QuadPlane in GUIDED mode. To use VTOL support in GUIDED mode you need to set the Q_GUIDED_MODE parameter to 1. When set, GUIDED mode behaviour will change so that the position hold at the destination will be done as a VTOL hover rather than a fixed wing circle.

The approach to the guided waypoint will be done as a fixed wing aircraft. The transition to VTOL flight will begin at the WP_LOITER_RAD radius in meters. This should be set appropriately for your aircraft. A value of 80 meters is good for a wide range of QuadPlanes.

When hovering at the destination in GUIDED mode if a new GUIDED destination is given then the aircraft will transition back to fixed wing flight, fly to the new location and then hover again in VTOL mode.

QuadPlane Simulation

A simple QuadPlane model is available in SITL, allowing you to test the features of the QuadPlane code without risking a real aircraft.

You can start it like this:

```
sim_vehicle.py -j4 -f quadplane --console --map
```

To visualise the aircraft you can use FlightGear in view-only mode. The simulation will output FlightGear compatible state on UDP port 5503. Start FlightGear using the **fg_plane_view.sh** scripts provided in the **Tools/autotest** directory.

Note that to get good scenery for FlightGear it is best to use a major airport. I tend to test at San Francisco airport, like this:

```
sim_vehicle.py -L KSFO -f quadplane --console --map
```

Using the joystick module with a USB adapter for your transmitter gives a convenient way to get used to the QuadPlane controls before flying.

If flying at KSFO there is a sample mission available with VTOL takeoff and landing:

```
wp load ../Tools/autotest/ArduPlane-Missions/KSFO-VTOL.txt
```

As usual you can edit the mission using “module load misseditor”

Fixed Wing FAQ

This is a set of frequently asked questions with answers. It is created when we see questions in the forums or on discuss.ardupilot.org that are not sufficiently answered in the rest of the docs.

When reading this FAQ please refer to the full parameter list for an explanation of each parameter which is mentioned in the answers.

How do you stop a Nitro plane from cutting the engine in flight?

For internal combustion motor planes (which can be prone to cutting the engine at low throttle), you should use the following settings:

- THR_MIN=10
- THR_PASS_STAB=1
- THR_SUPP_MAN=1

That will prevent the throttle dropping below 10%, but will give you manual throttle control for idling while on the ground, and manual throttle control in stabilisation modes (such as FBWA and STABILIZE) for shutting down the motor when you need to.

THR_SLEWRATE can also aid in prevention of a nitro engine stalling in flight by slowing the throttle advance from going wide open to quickly.

How do you prevent the servo demo and have faster startups?

Set the parameter SKIP_GYRO_CAL=1. That will disable the servo demo and will also skip the gyro calibration done on each startup. The last gyro calibration from when you calibrated the accelerometers will be used instead. This also means you don’t need to hold the plane still when booting.

The downside of using SKIP_GYRO_CAL is that your gyros may have drifted since you last calibrated them, perhaps due to temperature changes. If you have ARMING_REQUIRE enabled then the arming checks will catch that problem if it is very large, but in general we recommend not using SKIP_GYRO_CAL unless you have a good reason to use it. This is especially the case if you have the EKF enabled, as it is particularly sensitive to gyro error.

Tuning is too difficult, how do I make it easier?

Please see the documentation for autotune

How do I setup reverse throttle on a IC plane?

Some planes (mostly nitro or petrol planes) have a reversed throttle servo, so lower PWM values on the throttle channel gives more throttle not less. To setup Plane to handle this you need to change 3 settings:

- set RC3_REV to -1
- setup your transmitter for reverse throttle
- change THR_FS_VALUE to a high value instead of a low value (eg. 2100 instead of 900). It needs to be a higher PWM value than you will use in normal flight

After you setup reverse throttle make sure you test correct failsafe by turning off your transmitter while on the ground.

What happens if an airspeed sensor fails in flight?

What happens when airspeed fails depends on the type of failure. The most likely scenarios are:

- the I2C cable fails with a digital airspeed sensor
- the pitot tube gets partially blocked, leading to incorrect scaling of the pressure (either too low or high)
- the pitot tube gets completely blocked, leading to fixed readings (either too low or high)

In the first case where an I2C failure is detected the code will detect the failure and stop using the airspeed sensor. The aircraft will continue to fly with the algorithms it uses for an aircraft with no airspeed sensor. The ground station will be notified of the failure, and should display a warning to the user.

For the other two cases the code does not currently have a reliable way to detect an airspeed failure. If the failure leads to a low airspeed reading then if the plane is in an auto-throttle mode (such as AUTO, GUIDED, LOITER or RTL) then the plane will tend to lose altitude as it tries to gain speed. The amount of altitude it will lose depends on how low the airspeed reading is. If the airspeed reading it low enough then it may trigger a fast enough descent to crash the aircraft.

If the failure leads to a too high airspeed reading then the plane will slow down to try to keep its airspeed at the target airspeed. If the reading is high enough then the plane may slow down enough to cause a stall and crash the aircraft.

We are looking at ways of detecting partial airspeed sensor failures and hope to add some protection into the code in the future.

Why don’t my surfaces move enough with FLAPERON_OUTPUT, VTAIL_OUTPUT or ELEVON_OUTPUT?

You are probably using the default MIXING_GAIN of 0.5. The default is setup to prevent channel saturation. If you instead want to be able to have full deflection then try setting MIXING_GAIN=1.0 or something in between.

How do I get a good flare in automatic landing?

Please see this page

How do I reset all parameters to defaults?

To reset all parameters set the parameter FORMAT_VERSION to 0 and reboot. When ArduPilot starts up it checks if FORMAT_VERSION has the correct value, and if it doesn't it wipes the parameters, which resets them to the default values.

What does "Bad AHRS" mean on a ground station?

It means the "Attitude Heading Reference System" is unhealthy. That is the software that determines the attitude of the aircraft. The most common cause is bad 3D accelerometer calibration when AHRS_EKF_USE is set to 1. Check your accelerometer calibration.

How do I reduce throttle oscillation in auto flight?

There are 3 parameters that affect the amount the throttle changes in automatic flight.

- THR_SLEWRATE is the percentage of throttle change allowed per second. A value of 100 means the throttle cannot change over its full range in less than 1 second.
- TECS_THR_DAMP is a damping factor for throttle control. The default is 0.5. A higher value will dampen throttle changes.
- TECS_TIME_CONST is the overall time constant for both throttle and pitch changes in TECS. It controls how rapidly TECS tries to correct for any error in speed or height. It is in seconds, and defaults to 5. A higher value makes the pitch and throttle corrections happen more slowly.

Why do I get small surface movements in ground tests?

Before takeoff it is common to look at the amount of movement of ailerons and elevator when the plane is rolled and pitched on the ground. Some users have wondered why the amount of movement they see in this test is less in recent releases of the firmware.

The reason is the new stall prevention code. When the plane is on the ground the airspeed is very low, so is always under the minimum airspeed set in ARSPD_FBW_MIN. That means the maximum roll demand is limited to 25 degrees, which means the amount of demanded aileron surface movement is less than it would be without stall prevention.

If you want to see what the movement would be without stall prevention then just set STALL_PREVENTION=0. Remember to turn it back on before you fly.

How would I setup crow flaps?

Crow flaps combined flaperons with normal flaps, but the flaperons move upward when the flaps are engaged. Crow flaps can add a lot of drag to slow an aircraft for landing without inducing a lot of pitching moment.

To setup crow flaps you will need to combine two features. First you will need to setup flaperons on two output channels using the flaperon output channels functions. You will need to choose the FLAPERON_OUTPUT parameter value so that the flaps go up instead of down when flaps are engaged, while being careful that aileron input goes in the right direction for roll.

Then you should separately setup 1 or 2 flap channels (depending on whether your flap servos are setup to use a Y lead) using the flap_auto output channel function.

It is strongly suggested that you also setup a FLAP_INPUT_CHANNEL on an RC input channel to allow easy testing of flaps on the ground, and to give manual flap control for testing in FBWA mode. That will allow you to test what degree of flap movement produces the desired increase in drag without inducing a stall.

Why do my servos jitter when on the ground?

When the aircraft is on the ground in a mode where it is doing attitude stabilization (such as FBWA mode) the servos often move about a small amount, even though the aircraft is not moving.

The reason this happens is the attitude estimation code is doing it's best to estimate the attitude of the aircraft, and it is getting a small amount of false input. The sources of the false input are:

- if you have GPS lock then the GPS may be reporting a small amount of velocity change (GPS noise). This gets used to correct the accelerometers and comes out as a small amount of attitude noise, resulting in small attitude corrections.
- if you don't have GPS lock but you have an airspeed sensor then the DCM code will try to use cross-product of the airspeed with the gyros to estimate inertial force corrections to the accelerometers. The airspeed is quite noisy at low speed, so this effect can be quite large

Both of these effects are smaller if you enable the EKF (with AHRS_EKF_USE=1) as it has smarter logic for handling attitude estimation when on the ground.

How is airspeed used with no airspeed sensor?

When you have an aircraft with no airspeed sensor Plane uses a range of techniques to fly as reliably as possible despite the lack of airspeed sensor data. The techniques are:

- a synthetic airspeed estimate is calculated by the AHRS system by combining a wind estimate, the GPS ground speed and the response of the aircraft when turning. This airspeed estimate is usually quite good, although it is not as accurate as a real airspeed sensor.
- for speed and height control, a different algorithm in TECS is used that does not rely on an airspeed measurement. The algorithm primarily relies on using throttle to maintain the desired height, relying on the fact that an aircraft will start to sink if its airspeed is too low. See the TECS code for full details.
- For surface speed scaling (the change in control surface movement needed with different airspeed) the synthetic airspeed estimate is used.
- For stall prevention (if enabled) the synthetic airspeed is used

When no airspeed sensor is available some parameters are not used for some purposes:

- the TRIM_ARSPD_CM parameter is not used as an airspeed target in auto flight. Instead the TRIM_THROTTLE parameter is used as base throttle, with extra throttle added/removed to retain the target altitude
- the ARSPD_FBW_MIN and ARSPD_FBW_MAX parameters are not used for airspeed limiting in TECS, but they are still used for the stall prevention code, using the synthetic airspeed value

Why does my trim change when I change modes?

Some people experience a problem where their roll or pitch trim changes when they change flight modes. So for example they are trimmed with level aileron in manual with the aircraft level and when they change to FBWA mode the ailerons move significantly off centre trim.

One likely cause of this is that you have a transmitter that has per flight mode trims. The Taranis is a good example of this if you use its builtin flight mode controls. What happens is you setup the plane with correct trims in MANUAL by adjusting using the trip tabs in flight, but those trims don't get used when you change flight modes. You need to change your transmitter settings so that the stick inputs are the same in all flight modes.
