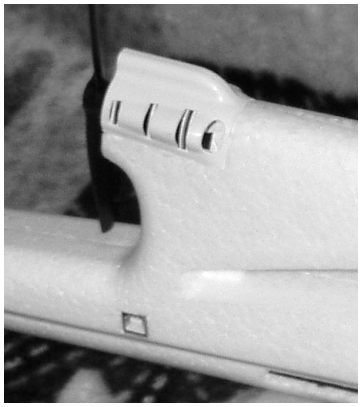
FLEXIPILOT 1.35

Programming navigation and failsafe features

The document describes programming waypoints and defining navigation parameters for the autopilot,
setting up actions to be performed upon RC control loss
and using mission simulator for mission testing.

Connection

The FLEXIPILOT has built-in USB connection capability at high speed. Before connecting the autopilot to your computer you can consider installing FTDI USB drivers supplied by the chip vendor, also found in the autopilot software bundle. The connection is supported with all modern operating systems and all you need to setup navigation is Console Terminal running at 921600 baudrate, 8N1 (eight bits, no parity, one stop bit, no flow control). Consider using HyperTerminal supplied with every Windows system.



Principles of operation

The autopilot uses the knowledge of roll, pitch, airspeed and other parameters for determining the optimal course of flight, which is measured over ground. The optimal course takes into account deviation from the desired flight path what eliminates the influence of crosswinds.

There is dedicated Feedback Control Regulator that finds the optimal turn rate that is required for converging to the optimal course. Normally the parameters of this regulator require laborious tuning that involves determining the airplane's dynamics. In certain cases, when the FLEXIPILOT is sold integrated with flying platform, this work is not necessary but as the characteristics will vary with different payload and wing loading, you are always free to change those parameters.
Another Feedback Control Regulator controls the optimal rudder deflection for obtaining the optimal course.
There are a lot of additional technical details about how the autopilot manages to maintain altitude in turns, how it prevents spin and how it adjusts for different airspeeds, but the general rule is that the whole navigation is based on WAYPOINTS.
Contrary to ground vehicles, the correspondence between what is 'drawn' by the waypoint and what you see on the sky is closely related to flight dynamics. It may surprise you how far the waypoints should be 'by the eye' in order to allow smooth flying. 'Tight' UAV patterns typically occupy area of the size 300x300m, even if the plane can make 50m-radius turns and follow the track within 5m. You have probably never measured it, but the RC modelers rarely fly further than 300m apart because of the 'psychical discomfort' in determining plane's orientation.
The geographical coordinates that define flight pattern are stored on non-volatile memory of the autopilot and once written, can be reused several times.

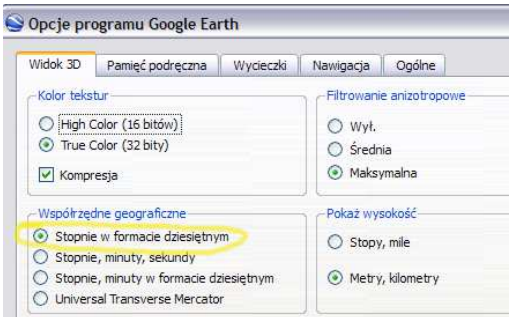
FLEXIPILOT by default is supplied with more than 1200 waypoint memory, but this number can vary slightly depending on firmware version.
Considering that the airplane turn radius is rarely under 50 meters, putting waypoints closer than 150m each may lead to a chain of oscillations as the plane tries to hopelessly converge towards every single waypoint. FLEXIPILOT has special logic that significantly reduces 'sneak' flying but the general rule applies for all autopilots: it is always better to place waypoints far away (300-400m) and use the path following features of the autopilot instead of micro-management. Typical flying pattern consists of 4-16 waypoints that can be looped over creating complex patterns.

Automated navigation is full of tradeoffs:
-you can hit all the waypoints precisely, but there will be more overshoot on the patterns flown and it might require loitering around
-you can fly with minimal number of course adjustments, but in presence of crosswind your flying pattern will be significantly distorted
-you can make smooth and symmetric flying pattern like a box or triangle, but you will not fly closer than 30m to any of the corner waypoints
-what is a predictable pattern in calm weather, will degenerate into one long loop and lots of skipped waypoint in heavy wind
FLEXIPILOT automatically handles strong wind navigation skipping dangerously difficult waypoints, while providing balancing options for track following and waypoint hit precision vs flight track smoothness.

In general, all waypoints are met in order in the sense that the conditions associated with them (triggers, altitude changes) are executed, but when the wind or flight dynamics disallows actually reaching the waypoint and the geometry to the next waypoint is favorable,

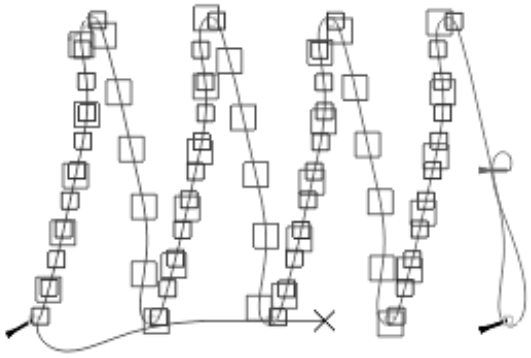
the next waypoint is selected. As a result, while the airplane ‘hits’ the waypoints, it is not guaranteed that it will be able to achieve desired altitude by that time, nor that it will always pass over a waypoint. With properly planned flight plane, however, you can expect the plane to hit a waypoint with 3-5m accuracy and maintain the altitude within 2-4m.

Defining waypoints is traditionally done using absolute coordinates. You can use Google Earth for this purpose. Please be aware that those maps are not necessarily exact, there is typically 10-50m positional error between cities in Europe. As the autopilot uses decimal latitude and longitude format, be sure to set decimal coordinate format in Google Earth menu Tools/Options/3D View tab.

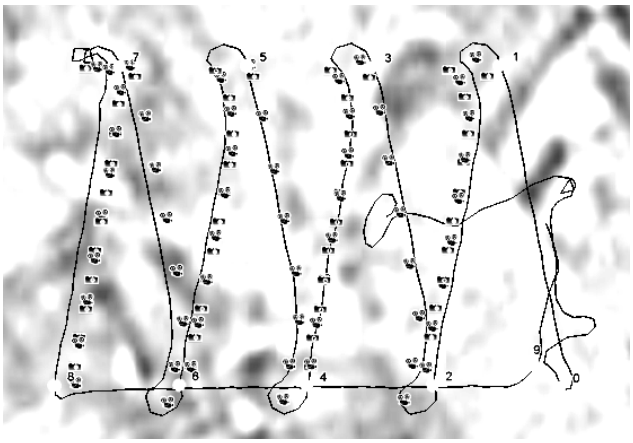


You can also define waypoints relative to takeoff position, previous waypoint, or fixed home position.

A simulated flight with 200m horizontal spacing of the waypoints, 400m pattern height:



The actual execution in calm weather:



Autopilot state machine

1. **Manual mode**
2. **Automatic mode**
    a. **Mission execution**
    b. **Loiter mode (by default 'loiter here')**
    c. **Magnet home (Loitering takeoff position)**
3. **Automatic assisted mode (autopilot assisted manual flying)**
    a. **Assisted flying for evasive action (on by default):**
    temporary turns or altitude change until RC sticks are centered RXOVR_ASSIST
    b. **Classic assisted flying (optional):**
    constant direction and altitude control by RC sticks RXOVR_ASSIST_EN
    c. **Conditional assisted flying (optional):**
    same as b. but only when within defined range RXOVR_ASSIST_MAXLOSDIST

Option 1
* can be disabled automatically following RXOVR_MODE rules (normally unused)
* can be disabled using RXOVR_SELECTOR_CHANNEL (normally unused)
* can be enabled/disabled using modem by RXOVRON, RXOVROFF, RXOVRAUTO (normally unused)

Option 2
* can be enabled by RC channel during normal operation (SCAP11 input channel)
* can be enabled using RC receiver failsafe settings (recommended)
* can be enabled/disabled using modem RXOVRON, RXOVROFF, RXOVRAUTO (normally unused)

Option 2a
* enabled in absence of configured MODE_CHANNEL (default mode)

Option 2b
* the autopilot enters this navigation mode using MODE_CHANNEL
* is configured using LOITER_MODE and can be absent, by default uses 'loiter here' logic
* can be enabled/disabled using modem commands WPTLOITERHERE, WPTLOITERTAKEOFF, WPTLOITERHOME, WPTLOITERTARGET, WPTLOITERPOS, WPTLOITER, WPTLOITERDONE(normally unused)
* loiter position can be updated in the background using WPTSETLOITERPOSHERE, WPTSETTARGETPOSHERE, WPTSETHOMEPOSHERE, WPTSETTAKEOFFPOSHERE

Option2c
* the autopilot enters this navigation submode using MODE_CHANNEL
* takeoff position is subject to offset/extrapolation using WAYPOINT_TAKEOFF_OFFDIST
* can be enabled by RC channel (recommended)
can be enabled/disabled using modem (normally unused)

Option 3a
* is normally configured and available anytime using aileron and elevator channel (default)
* strong deflections send turning order, when sticks are released, the plane resumes navigation
* is possible at any distance from takeoff, when enabled

Option 3b

* most useful for video surveillance or FPV flying (normally unused)
* might be dangerous if the plane stays in assisted mode out of transmitter range
* is automatically disabled when RXOVR condition has been triggered (also because of range)

Option 3c

* is useful for out-of sight flight. It prevents fail-safe values of the RC receiver interrupting the navigation.

Flight modes and failsafe settings

Suppose the RC receiver remembers pre-set channel value. The pilot should be aware how navigation modes interact in case of range loss and/or autopilot RETHOME decision:

RC selected	RC faisafe	Action upon RC signal loss
AUTO+RETHOME	**AUTO+RETHOME**	Will remain circling takeoff position
AUTO+RETHOME	**AUTO+LOITER**	Will move to last loiter position, which can be far away. **Unsafe!**
AUTO+RETHOME	**AUTO+MISSION**	Will start executing mission which might cross obstacles. **Unsafe if not in sight!** But RETHOME logic is still protecting.
AUTO+LOITER	**AUTO+RETHOME**	Will return home, probably closer to transmitter. However, might be disturbing during landing as loiter position might have been selected above a specific field.
AUTO+LOITER	**AUTO+LOITER**	Will remain in loiter and return home only if emergency RETHOME event will be detected. Using LOITER as failsafe in other scenarios is unsafe.
AUTO+LOITER	**AUTO+MISSION**	**Very misleading!** Every toggling from mission to loiter is setting new loiter position. Therefore every temporary signal loss will program a new loiter circle center (or other action, depending on LOITER_MODE)
AUTO+MISSION	**AUTO+RETHOME**	Recommended, universal setting. You could witness the plane interrupts the mission at the edge of the pattern just to resume it a few s later, yet you know it is the last leg and the mission plan is not going any further. In such case it is safe to program AUTO+MISSION as failsafe in flight and continue the mission. Keep AUTO+RETHOME as failsafe as often as possible as it gives the visual warning about signal range.
AUTO+MISSION	**AUTO+LOITER**	This will not let the plane escape, but will trap it exactly at the edge of RC range. Each toggle from good reception to bad reception will freeze plane in LOITER at that point, returning home only when RETHOME logic activates. This in fact might be too far to return back intact.
AUTO+MISSION	**AUTO+MISSION**	Will continue executing the mission. Normally safe, unless you might realize that a few km from takeoff position you have no option to force RETHOME manually, but RETHOME logic is still protecting.

Conclusion:
During test flights of a development platform, have MANUAL+RETHOME as failsafe, later using switching to AUTO+RETHOME as failsafe.
During all normal flying, have AUTO+RETHOME as failsafe, except when you deliberately fly out of range, in which case have AUTO+MISSION as failsafe.
If you want to stay in LOITER in place for a long time, program failsafe condition to either AUTO+LOITER or AUTO+RETHOME (most transmitters allow this to be changed in flight!).
Do it only well within visual range.

RC mode	RETHOME and RXOVR logic
MANUAL+any mode	RETHOME logic disabled, but RXOVR logic can disable RC receiver beyond specified range going to AUTO.
AUTO+RETHOME	• RXOVR will not activate unless the plane is sucked by thermals and RXOVR altitude is exceeded. • Otherwise emergency low voltage, flight time or similar events will occur unnoticed to the observer the result will be inability to start the mission but LOITER position can be set and assisted flying is possilbe. • During long distance return home, RETHOME logic might also disable climbing and allow gentle descent to minimal loiter altitude if the motor power is low.
AUTO+LOITER	Autopilot action: If the loiter is at lower altitude than minimal RETHOME altitude, immediate climb will be ordered and loiter mode is exited despite switch indicating LOITER navigation mode. If distance from loiter center to takeoff position is larger than {loiter radius (but not more than 200m) + 400m} loiter center position will be moved to takeoff position. This avoids mistake when the user keeps switching between loiter and RETHOME while loiter position is far from sight. • If the RETHOME_CLIMBTOAGL altitude is much higher and battery margin for RETHOME is low, it might happen also that the motor cuts off very shortly after low voltage emergency RETHOME. This occurs when loitering near ground level – a surprising sudden motor cutoff might results. • It could be LOITER mode was on while preparing for landing *but loiter position is far from takeoff*, therefore quick toggling to MISSION then LOITER will setup again loiter center in place (yet will not return to previous loiter position). • It could be LOITER mode was on while preparing for landing and *loiter position is close to takeoff position*, in such case old loiter position is not affected. One can toggle from LOITER to RETHOME and back to LOITER, this will not change the loiter center but will re-enable LOITER mode.
AUTO+MISSION	The MISSION position 'will stop working' and the plane will immediately turn to takeoff position. One can still enter LOITER for a brief time in order to walk for a better place to command the landing. Consult RETHOME emergency altitude diagram to understand better details of safety energy management, pay attention to 'CANNOT CLIMB' scenario.

Safety notes:

Physical damage to RC AUTO channel or signal malfunction sets to AUTO.

Physical damage to RC MODE channel or signal malfunction sets to RETHOME.

RC sanity and signal status are completely ignored if RXOVR logic disables possibility of RC control at specific distance or altitude.

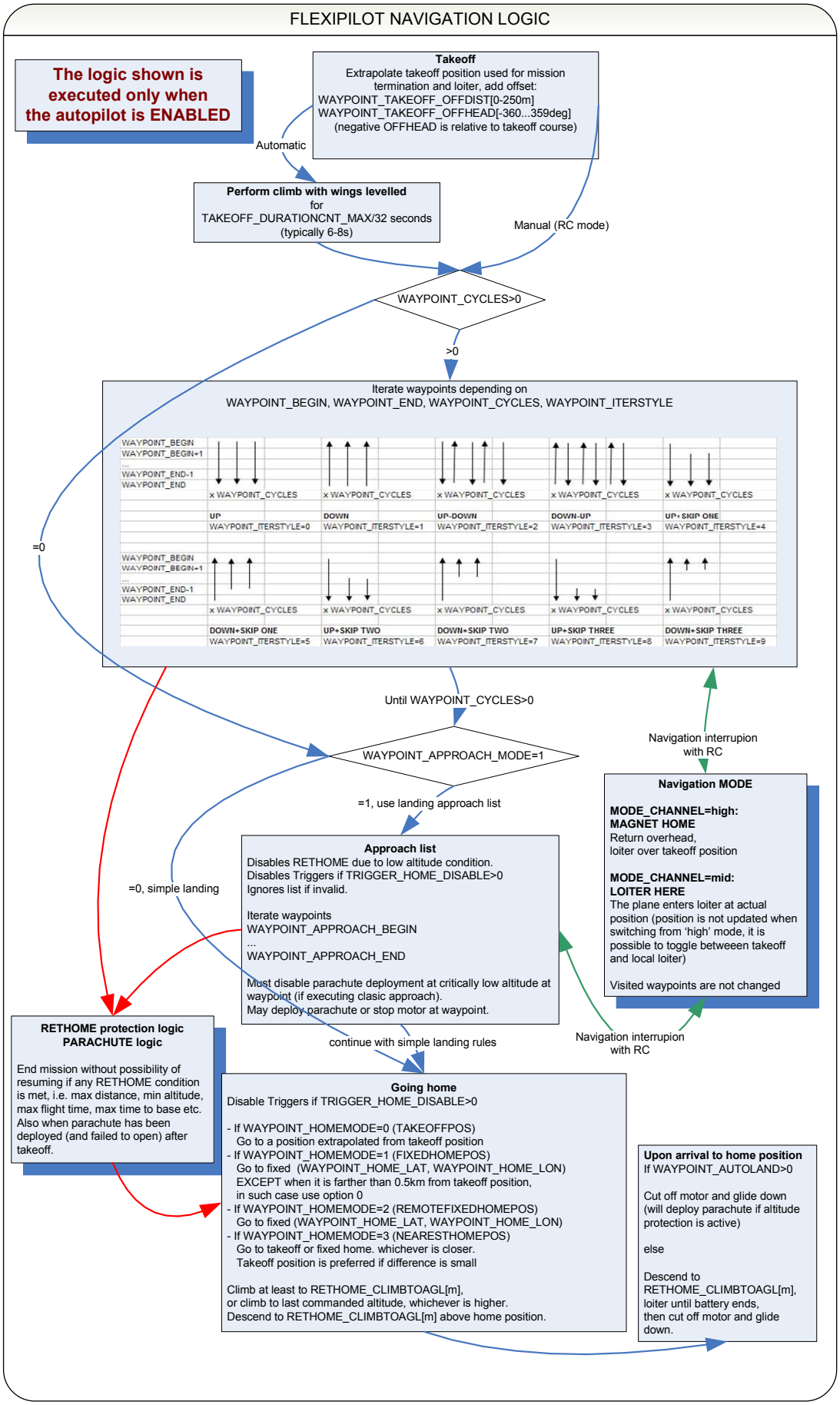
Key coordinates for waypoint engine

The whole concept of the autopilot revolves around a few key positions. They are:
Takeoff position – a point where the movement of the plane has been detected PLUS optional offset, max 250m (to avoid loiter above vehicle or catapult)
Previous position – the last position selected by waypoint engine, can serve as a base for next waypoint
Home position (or fixed home position) is one of the positions to go after mission termination. Another one is takeoff position
Target position – specified by the user and acts as a convenient hook for automatic pattern generation using waypoints
Loiter position – interactively updated in flight (initially equal to takeoff position) can serve as temporary station without recalling the UAV to takeoff position

Takeoff position offset is defined using WAYPOINT_TAKEOFF_OFFHEAD/OFFDIST.
Among those mentioned, only home position (WAYPOINT_HOME_LAT/LON) and target position (WAYPOINT_TARGET_LAT/LON) can be entered explicitly.
During simulation, takeoff position can be prescribed using SIM_LAT/LON.
If SIM_LAT/LON is undefined, WAYPOINT_TARGET_LAT/LON is used instead.
If the above is also null, WAYPOINT_TAKEOFF_LAT/LON is used instead.
If the above is also null, WAYPOINT_HOME_LAT/LON is used instead.

Waypoint sequencing engine

The waypoint engine is enabled only when the autopilot is enabled. Therefore you will not trigger any waypoint action by flying around in manual mode.
The most basic algorithm for the navigation engine is to cycle a list of waypoints N times then return home.
The number of cycles is defined by WAYPOINT_CYCLES. The range of waypoints defining a cycle is defined by WAYPOINT_BEGIN and WAYPOINT_END, both inclusive (WAYPOINT_END>=WAYPOINT_BEGIN). The order in which the waypoints within a cycle are executed is defined by WAYPOINT_ITERSTYLE. After all the cycles have been executed, or the initial number of cycles was 0, the plane returns to home. Every time when for any reason returning home is selected, if the plane is below RETHOME_CLIMBTOAGL, it will climb to that altitude. If it is below its last target altitude (say defined by the last waypoint), it will continue climbing to that waypoint, and will descend to RETHOME_CLIMBTOAGL after reaching home.


FLEXIPILOT NAVIGATION LOGIC
The logic shown is executed only when the autopilot is ENABLED
Takeoff
Extrapolate takeoff position used for mission termination and loiter, add offset:
WAYPOINT_TAKEOFF_OFFDIST[0-250m]
WAYPOINT_TAKEOFF_OFFHEAD[-360...359deg]
(negative OFFHEAD is relative to takeoff course)
Automatic
Perform climb with wings levelled
for
TAKEOFF_DURATIONCNT_MAX/32 seconds
(typically 6-8s)
Manual (RC mode)
WAYPOINT_CYCLES>0
>0
=0
Iterate waypoints depending on
WAYPOINT_BEGIN, WAYPOINT_END, WAYPOINT_CYCLES, WAYPOINT_ITERSTYLE
WAYPOINT_BEGIN
WAYPOINT_BEGIN+1
...
WAYPOINT_END-1
WAYPOINT_END
x WAYPOINT_CYCLES
UP
WAYPOINT_ITERSTYLE=0
DOWN
WAYPOINT_ITERSTYLE=1
UP-DOWN
WAYPOINT_ITERSTYLE=2
DOWN-UP
WAYPOINT_ITERSTYLE=3
UP+SKIP ONE
WAYPOINT_ITERSTYLE=4
DOWN+SKIP ONE
WAYPOINT_ITERSTYLE=5
UP+SKIP TWO
WAYPOINT_ITERSTYLE=6
DOWN+SKIP TWO
WAYPOINT_ITERSTYLE=7
UP+SKIP THREE
WAYPOINT_ITERSTYLE=8
DOWN+SKIP THREE
WAYPOINT_ITERSTYLE=9
Until WAYPOINT_CYCLES>0
WAYPOINT_APPROACH_MODE=1
Navigation interrupion with RC
Navigation MODE
MODE_CHANNEL=high:
MAGNET HOME
Return overhead,
loiter over takeoff position
MODE_CHANNEL=mid:
LOITER HERE
The plane enters loiter at actual position (position is not updated when switching from 'high' mode, it is possible to toggle betweeen takeoff and local loiter)
Visited waypoints are not changed
=1, use landing approach list
Approach list
Disables RETHOME due to low altitude condition.
Disables Triggers if TRIGGER_HOME_DISABLE>0
Ignores list if invalid.
Iterate waypoints
WAYPOINT_APPROACH_BEGIN
...
WAYPOINT_APPROACH_END
Must disable parachute deployment at critically low altitude at waypoint (if executing clasic approach).
May deploy parachute or stop motor at waypoint.
=0, simple landing
RETHOME protection logic
PARACHUTE logic
End mission without possibility of resuming if any RETHOME condition is met, i.e. max distance, min altitude, max flight time, max time to base etc.
Also when parachute has been deployed (and failed to open) after takeoff.
continue with simple landing rules
Navigation interrupion with RC
Going home
Disable Triggers if TRIGGER_HOME_DISABLE>0
- If WAYPOINT_HOMEMODE=0 (TAKEOFFPOS)
Go to a position extrapolated from takeoff position
- If WAYPOINT_HOMEMODE=1 (FIXEDHOMEPOS)
Go to fixed (WAYPOINT_HOME_LAT, WAYPOINT_HOME_LON)
EXCEPT when it is farther than 0.5km from takeoff position,
in such case use option 0
- If WAYPOINT_HOMEMODE=2 (REMOTEFIXEDHOMEPOS)
Go to fixed (WAYPOINT_HOME_LAT, WAYPOINT_HOME_LON)
- If WAYPOINT_HOMEMODE=3 (NEARESTHOMEPOS)
Go to takeoff or fixed home. whichever is closer.
Takeoff position is preferred if difference is small
Climb at least to RETHOME_CLIMBTOAGL[m],
or climb to last commanded altitude, whichever is higher.
Descend to RETHOME_CLIMBTOAGL[m] above home position.
Upon arrival to home position
If WAYPOINT_AUTOLAND>0
Cut off motor and glide down (will deploy parachute if altitude protection is active)
else
Descend to RETHOME_CLIMBTOAGL[m], loiter until battery ends, then cut off motor and glide down.

Home position is defined either by (WAYPOINT_HOME_LAT, WAYPOINT_HOME_LON) or is set automatically (WAYPOINT_TAKEOFF_LAT, WAYPOINT_TAKEOFF_LON).
WAYPOINT_HOMEMODE selects what kind of home position is being used upon return.
After returning home, the plane will be circling it until the ESC cuts off motor because of low battery. Alternatively, WAYPOINT_AUTOLAND can order motor cutoff upon returning home.
WAYPOINT_HOME_TURNDIR can select specific turning direction after returning home.

The sequencing engine guarantees no infinite cycles, no loiter around waypoint if the satisfy radius is too small, obeys turning direction defined by waypoints and has ability to select the next waypoint if the wind is not permitting to reach the current one without causing oscillation.

Navigation modes affecting iterator logic

There are several methods to interrupt (bypass) UAV navigation:

- **Full Manual Control**: (RXOVR is RC override, RC controller overrides the autopilot)
- **Assisted Flying/Manual Navigation**: RXOVR logic can override navigation by RC receiver using a specific channel, typically the same used for direction control in manual mode, while the autopilot controls plane orientation
- **Selecting Autopilot Navigation Mode**: use MODE_CHANNEL: either Magnet Home or Loiter modes can be selected. Those are frequently used before departing for a mission and are by-passing the waypoint engine.
- **Change mission plan** using console commands (@@@MISSIONSELECTn, @@@WPTRESET etc using a modem, if connected, or via USB during simulation)

MODE_CHANNEL at mid position is equivalent to @@@WPTLOITERHERE
MODE_CHANNEL at high position is equivalent to @@@WPTLOITERTAKEOFF

Try advancing waypoint when on the ground.
Use **@@@WPTGOn** in order to select any waypoint than don't have to be on the list. Note, however, this could disturb the navigation engine (but will never cause the software crash)

@@@WPTNEXT will advance to the next waypoint according to iterator engine and WAYPOINT_ITERSTYLE modifier. If the mission is ended the UAV will proceed to approach list or landing.

@@@WPTAPPROACH will advance iterator logic to approach list, even if the list was initially disabled using WAYPOINT_APPROACH_MODE. If there is no such list, will continue to landing logic.

@@@WPTHOME will trigger immediate returning home and associated actions like automatic landing. The landing place chosen will depend on WAYPOINT_HOMEMODE

@@@WPTRESET Restart mission (refresh waypoint lists only)

@@@WPTINIT Restart mission and reload all mission configuration variables.

@@@SETQNH pressure, altitude Allows setting barometer pressure for given altitude, thus adjusting barometric AGL altitude used normally for navigation. This allows precise landing approach, while cross-country waypoint can use MSL altitude which is unchanged.
Altitude defaults fro 0m.
Pressure in Pascals.
Example

@@@SETQNH 101375	sets standard atmosphere conditions
@@@SETQNH 101375,0	sets standard atmosphere conditions
@@@SETQNH 110040.2,-56	90040.2Pa is assumed at -56m altitude

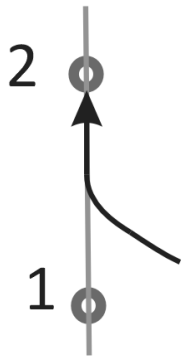
Note: standard atmosphere altitude is also assuming specific air temperature, while normal AGL barometric altitude calculations also adjust for actual temperature. Therefore STDalt will differ very slightly (about 1-3m) from AGL in most cases.
Before takeoff, AGL is self calibrating as ATL (Above Takeoff Level). Using this command will freeze automatic takeoff altitude detection.

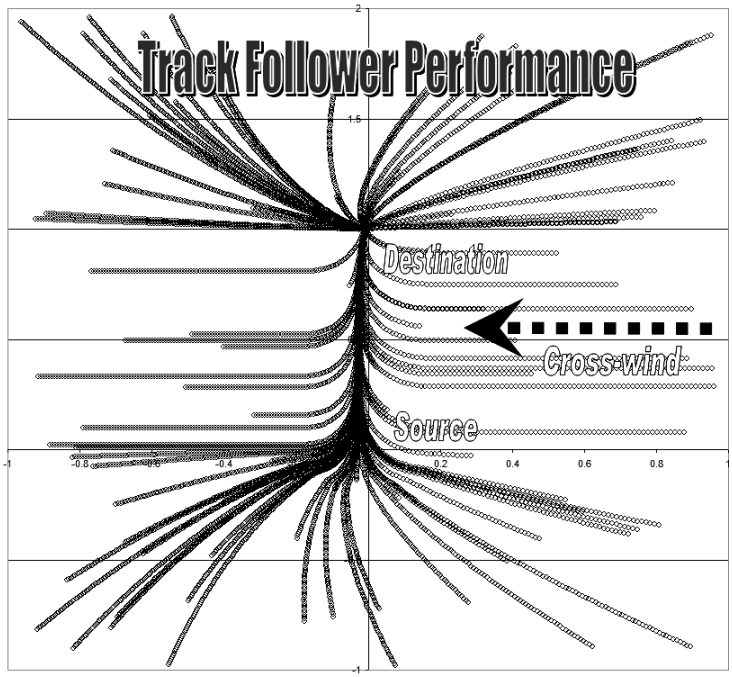
Mission programming

Certain versions of the autopilot contain MISPROG logic that allows altering the waypoint list interactively, using console, RC transmitter or using rotary mission selector.
Use of this logic is permanently affecting the variables WAYPOINT_CYCLES, WAYPOINT_BEGIN, WAYPOINT_END. Up to 128 missions can be stored; each can span any continuous list of waypoints that is iterated using fixed WAYPOINT_ITERSTYLE set of rules.
The mission can be 'loaded' during autopilot boot up or not, depending of MISPROG configuration. The approach list is executed independently and is not affected by mission selection.

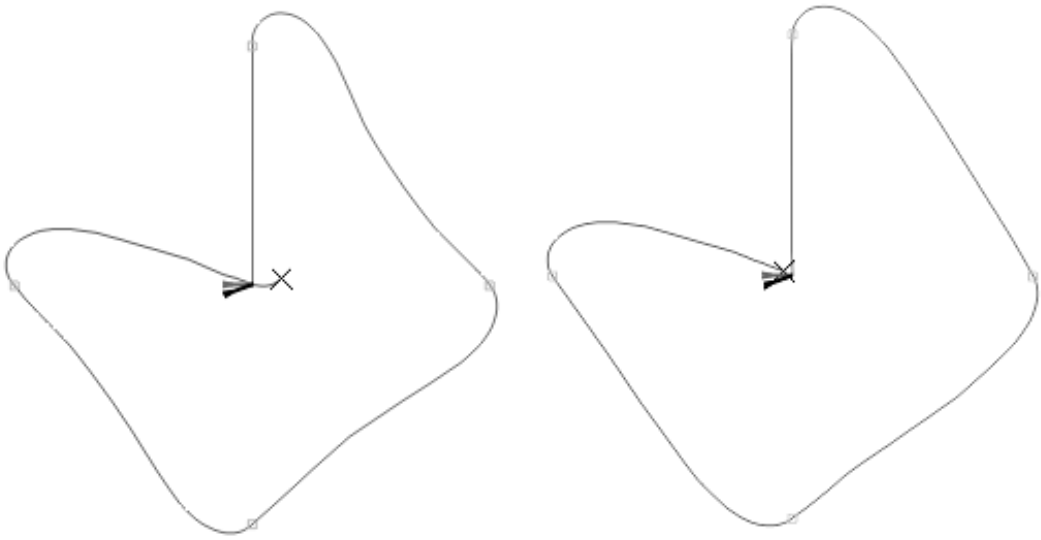
Track Following

A difficult scenario occurs when the autopilot is far form the line connecting two remote waypoints. Due to strong crosswind, large deviations must be corrected by enabling WAYPOINT_NAVIMODE Track Follower.





Track Following requires more turns than ‘magnet-style navigation’ and might be suboptimal for video recording. Consider the following example:

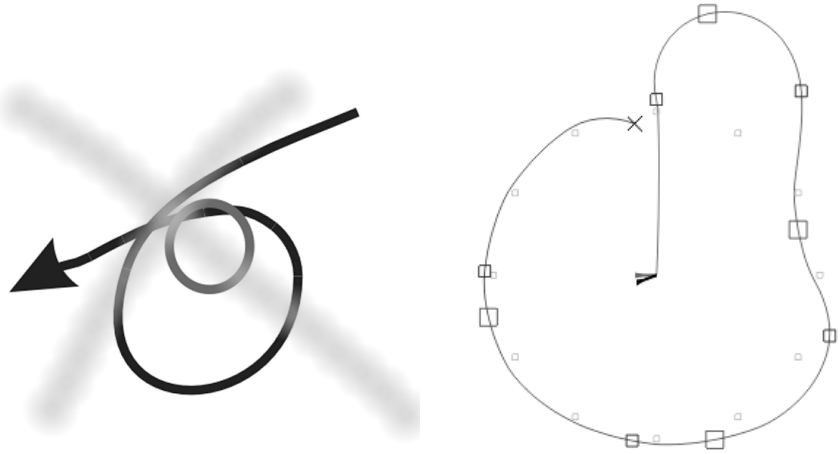


In null winds, a tight pattern (150m edge) with Track Follower (left) requires more turns in both directions than simple magnet-follower navigation (right). Both are possible with FLEXIPILOT.

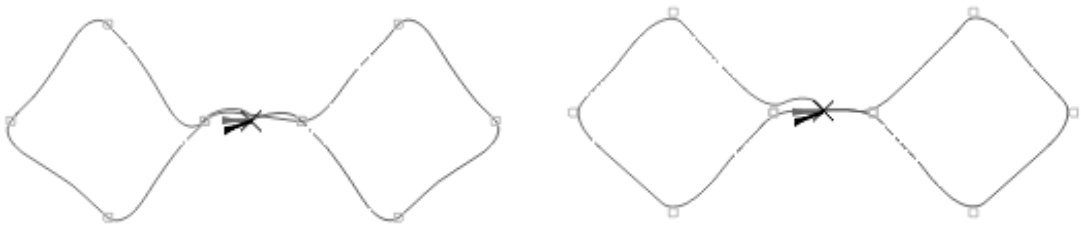
Hitting a waypoint

The conditions to mark a waypoint as visited are any of the following:

- The airplane must be no further than WAYPOINT_SATISFY_RADIUS in meters and the distance to the waypoint starts increasing; vertical position (altitude) is not taken into account in navigation.
- The airplane is closer than WAYPOINT_ROUNDCORNER_RADIUS (in meters) to the active waypoint
- @@@WPTNEXT command has been issued.
- The plane would need to loiter around waypoint; the navigation engine chooses to go to the next waypoint.



The navigation routine eliminates loitering if it is not possible
to overfly directly a waypoint at given airspeed and wind conditions.



Left: the airplane passes directly over waypoints,
Right: WAYPOINT_ROUNDCORNER_RADIUS=30m allows more regular flight pattern.

Interrupting the navigation

This can occur under the following conditions:

- MODE_CHANNEL connected to the RC receiver can request the plane to head home, without interrupting the navigation engine but selecting home heading as priority.
- Using rudder control on the RC receiver, when the autopilot is engaged, orders the plane to fly with specific turn rate left or right. This is supporting basic terrain avoidance without interrupting the mission, at long range. As soon as direction stick is released, the navigation resumes.
- RETHOME engine can trigger immediate return home upon certain conditions; those include changing the position of selected RC channel if connected. This is a fail-safe in dangerous situations and is not reversible in order to avoid multiple returns to the dangerous range zone.

Diagnostics

Upon bootup, observe the navigation section of the report if there are no clear errors.
An example of a wrong setting:

```
INVALID==> 500.000000 Variable read {float}: NAVI_WIND_HEADING=0.000000
```

Observe as the system assumed default, safe value instead of the incorrect one.

You can issue **@@@WPTRESET** or **@@@RESET** command in order to view that information.
@@@WPTRESET

```
Variable read {u16}: WAYPOINT_BEGIN=350
Variable read {u16}: WAYPOINT_END=360
Variable read {u16}: WAYPOINT_CYCLES=10
Variable read {u8} : WAYPOINT_APPROACH_MODE=0
Variable read {u16}: WAYPOINT_APPROACH_BEGIN=0
Variable read {u16}: WAYPOINT_APPROACH_END=0
Variable read {f32}: WAYPOINT_TAKEOFF_LAT=51.169243
Variable read {f32}: WAYPOINT_TAKEOFF_LON=16.987991
Variable read {f32}: WAYPOINT_TARGET_LAT=50.116287
Variable read {f32}: WAYPOINT_TARGET_LON=22.458261
Variable read {f32}: WAYPOINT_HOME_LAT=0.000000
Variable read {f32}: WAYPOINT_HOME_LON=0.000000
Variable read {f32}: WAYPOINT_ROTATION=0.000000
Variable read {u8} : WAYPOINT_HOMEMODE=0
Variable read {u8} : WAYPOINT_AUTOLAND=0
Variable read {u8} : WAYPOINT_NAVIMODE=2
Variable read {u8} : WAYPOINT_ALTIMODE=3
Variable read {u8} : WAYPOINT_SPECIALMODE=16
Variable read {u8} : WAYPOINT_ITERSTYLE=0
```

@@@RESET, at waypoint section

```
===== WAYPOINT_Init =====
Max Available Waypoint Storage: 1554
Variable read {u16}: WAYPOINT_SIZE=4096
Max Accessible No of Waypoints: 1554
Variable read {u8} : WAYPOINT_HOME_DISTREACHED=80
Variable read {u8} : WAYPOINT_SATISFY_RADIUS=80
Variable read {u8} : WAYPOINT_SATISFY_MUST_CLIMB=10
Variable read {u8} : WAYPOINT_SATISFY_MUST_DESCEND=10
Variable read {u8} : WAYPOINT_ROUNDCORNER_RADIUS=50
Variable read {u8} : WAYPOINT_HALFPLANE_RADIUS=200
Variable read {f32}: WAYPOINT_TAKEOFF_LAT=51.169243
Variable read {f32}: WAYPOINT_TAKEOFF_LON=16.987991
Variable read {u8} : WAYPOINT_TAKEOFF_OFFDIST=100
Variable read {i16}: WAYPOINT_TAKEOFF_OFFHEAD=-360
Variable read {i8} : MODE_CHANNEL=6
Variable read {u8} : WAYPOINT_HITMODE=0
Variable read {u8} : LOITER_MODE=1
Variable read {f32}: LOITER_CLIMBRADIUS=0.050000
Variable read {f32}: LOITER_DESCENTRADIUS=0.050000
Variable read {f32}: WLOITER1_TIME=-30.000000
Variable read {f32}: WLOITER2_TIME=162345.610000
Variable read {f32}: WLOITER3_TIME=0.000000
Variable read {f32}: WLOITER1_RADIUS=0.100000
Variable read {f32}: WLOITER2_RADIUS=-0.100000
Variable read {f32}: WLOITER3_RADIUS=0.300000
Variable read {f32}: LOITER_RADIUS=0.030000
Variable read {f32}: LOITER_RADIUS_MIN=-0.100000
Variable read {f32}: LOITER_RADIUS_MAX=0.100000
Variable read {i8} : LOITER_RADIUS_INCHANNEL=-1
Variable read {u16}: LOITER_RADIUS_INCHANNEL_MIN=11000
```

```
Variable read {u16}: LOITER_RADIUS_INCHANNEL_MAX=18400
WAYPOINT_Reset...
Variable read {u16}: WAYPOINT_BEGIN=350
Variable read {u16}: WAYPOINT_END=360
Variable read {u16}: WAYPOINT_CYCLES=10
Variable read {u8} : WAYPOINT_APPROACH_MODE=0
Variable read {u16}: WAYPOINT_APPROACH_BEGIN=0
Variable read {u16}: WAYPOINT_APPROACH_END=0
Variable read {f32}: WAYPOINT_TAKEOFF_LAT=51.169243
Variable read {f32}: WAYPOINT_TAKEOFF_LON=16.987991
Variable read {f32}: WAYPOINT_TARGET_LAT=50.116287
Variable read {f32}: WAYPOINT_TARGET_LON=22.458261
Variable read {f32}: WAYPOINT_HOME_LAT=0.000000
Variable read {f32}: WAYPOINT_HOME_LON=0.000000
Variable read {f32}: WAYPOINT_ROTATION=0.000000
Variable read {u8} : WAYPOINT_HOMEMODE=0
Variable read {u8} : WAYPOINT_AUTOLAND=0
Variable read {u8} : WAYPOINT_NAVIMODE=2
Variable read {u8} : WAYPOINT_ALTIMODE=3
Variable read {u8} : WAYPOINT_SPECIALMODE=16
Variable read {u8} : WAYPOINT_ITERSTYLE=0
Home is Sim...
WAYPOINT_ChangeWaypoint:350
#AREVE51.169235 16.987991 0#
Total N of waypoints: 1554
Actual waypoint: (51.169235,16.987991)
Last waypoint: (INVALID)(0.000000,0.000000)
Before-Last waypoint: (INVALID)(0.000000,0.000000)
Last takeoff: (51.169243,16.987991)
Fixed home:   (0.000000,0.000000)
WAYPOINT_TARGET=(51.169235,16.987991,200),0.000000,0.000000,200,0,20,0
```

Observe that waypoint storage using this particular autopilot firmware can fit 1554 waypoints and this number is not additionally limited as WAYPOINT_SIZE is 4096.

You can press **w** or use **@@@WPTSTATUS** at any time in order to show the waypoint report. Observe the typical printout:

```
@@@WPTSTATUS
Total N of waypoints: 1384
Actual waypoint: (51.119854,17.117723)
Last waypoint: (INVALID)(51.123455,17.123455)
Before-Last waypoint: (INVALID)(0.000000,0.000000)
Last takeoff: (51.123455,17.123455)
Fixed home:   (51.054993,17.379019)
WAYPOINT_TARGET=(51.119854,17.117723,200),-0.400000,-0.400000,200,0,1,3
```

It says that the next waypoint using rules
-0.400000,-0.400000,200,0,1,3
(param1=-.4, param2, -.4, alt=200m, 0, evalmode=1, triggerflags=3)
is evaluating to coordinates and altitude (lat=51.119854,lon=17.117723,alt_agl=200m).
This evaluation may depend on last takeoff position, since when you are typing this, the new takeoff has not occurred yet.

You can use **@@@WPTSHOWn** (default n=0) in order to view the waypoint n.
For example
@@@WPTSHOW12
#ARW12:(1.180252,7.223686),270.000000,0.200000,50,0,4,0#

@@@WPTSHOWLIST shows the waypoint list between WAYPOINT_BEGIN and WAYPOINT_END, inclusive. Note that the output is not printed in order of execution of those waypoints, as this depends on WAYPOINT_ITERSTYLE. Therefore the evaluation of those waypoints in flight can be different.

@@@WPTSHOWALL shows all the waypoint, not only those selected for flight. Observe if they evaluate to reasonable coordinates in order to avoid future problems related to configuration mistakes in the field. It is technically safe to have incorrectly defined waypoints, but might lead to confusing navigation before the RETHOME safety features kick in.

@@@WPTITERALL will iterate all existing waypoints given number of cycles. The results will be displayed in a format compatible with UAVStation, displaying all waypoint positions to be visited. Also, the command calculates expected flight length including SIM_LAT and SIM_LON as takeoff point, plus mission time using SIM_AVG_AIRSPEED_KMH plus safety margin.

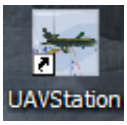
@@@TRACEDUMP will contain also waypoint hit events, this works also during simulation. The event ID=**3** (the last field) is waypoint hit and looks like this:

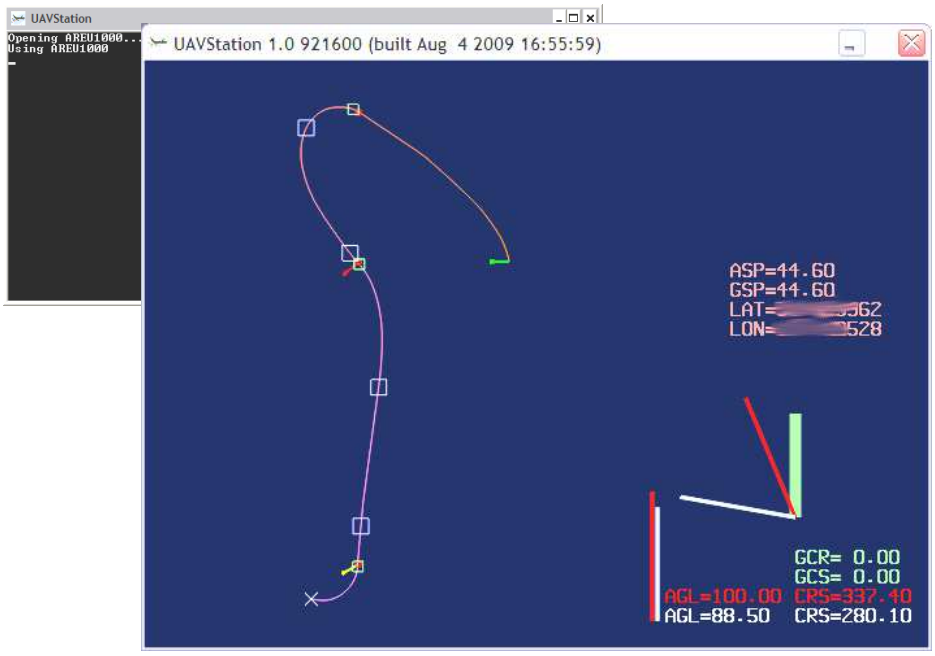
```
#ART00012:44631.2,+51.062206,+017.391760,+0000.1,-006,+002,+349,2009-08-11,12:23
:51.200,3#
```

Use **C** or **@@@TRACECLEAR** if the TRACE becomes polluted.

Using mission simulator

Close the HyperTerminal and launch the **UAVStation** console:





Red lines are AGL altitude target and CRS course target.
Light blue are simulated plane course and altitude.
White X cross is plane position.
@@@SIMENABLE turns off the motor using @@@THRDISABLE command.
If you will not issue the @@@RXOVROFF command, the autopilot would switch to emergency return home triggered by the RC transmitter/receiver.
Another method of preventing the fail-safe channels to disrupt the simulation, is to set the RC receiver to autopilot mode, and disable emergency channels: @@@MODE_CHANNEL-1, @@@RETHOME_CHANNEL=-1. However special care has to be taken as those channels are important safety feature.

Green arrow is takeoff position.
Blue arrow is fixed home position and will show on the screen only if you are close to the launch site.
Red arrow is 'before-last' waypoint.
Yellow arrow is already met waypoint.
White arrow is the next waypoint.
Visited waypoints are **Red boxes**.
Small **Green boxes** are Trigger1.
Large **Blue boxes** are Trigger2.

You can type all the commands to the graphic window as with HyperTerminal.
By issuing the commands @@@RXOVROFF and @@@SIMENABLE you can launch a simulation. Typical output will be:

```
Opening AREU1000...
Using AREU1000
@@@RXOVROFF
RX override DISABLED...
@@@SIMENABLE
Simulation enabled...
LOG_Enable...
TRACE_Enable...
```

@@@WPTITERALL issued before or after the simulation will display all waypoints in graphical form.

External factors impacting mission simulation

Servo moves

Warning: Trigger servos will move during the simulation as programmed. Make sure they will not damage themselves in the process.

The pitch damping and turn rate damping action can be observed on the control surfaces.

Engine throttle

Warning: Engine throttle is DISABLED automatically inside the autopilot logic when launching the simulation (@@@THRDISABLE) , however if @@@RXOVROFF is not issued, if you enable the RC override on the transmitter and increase throttle, the engine will start as you will have full authority during RC override, exactly as during the real flight.

RC transmitter and receiver

If @@@RXOVROFF command is not issued, the RC receiver inputs are used during the simulation.

In particular, emergency obstacle avoidance is still active and you can change the course by using extreme rudder deflection right or left on your transmitter. After the stick is released, the navigation resumes. Manual control of altitude in this mode is not possible as it would most likely lead to loss of altitude and airspeed control, what combined with average climb rate of most platforms, wouldn't help avoiding the obstacle.

Enabling the RC transmitter during the simulation facilitates testing of its features in flight. In particular, when magnet-home (MODE_CHANNEL) or RETHOME_CHANNEL is set incorrectly, you will observe the plane immediately returning home.

When starting the simulation while the RC transmitter is off, failsafe settings of the RC receivers will be used. Those will include typically the channel input value set in such a way that plane runaway is prevented, heading home immediately.

Manual movement of the airplane

The IMU module is actively calculating the yaw rate. You can change the real plane's course manually during simulation. You can also change the real plane pitch manually and observe its impact on airspeed and groundspeed. You cannot change the altitude this way which is

evaluated automatically, based on typical UAV performance. The reason is that it is impractical to keep the pitch within reasonable limits during simulation.

Mission simulation using video overlay: RVOSD

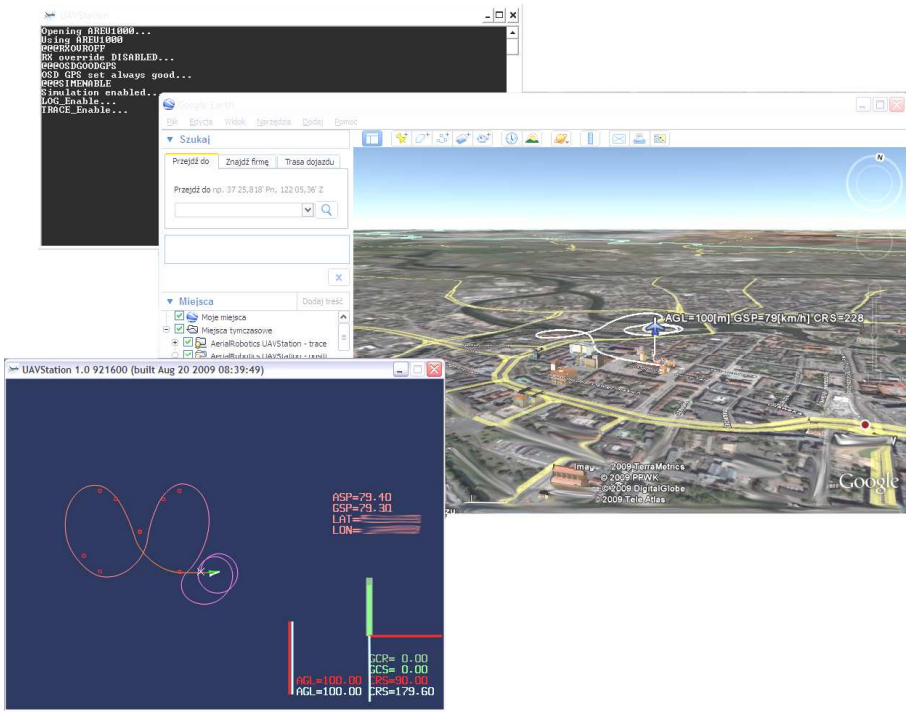
You can use a special version of RVOSD display during the simulation when the plane lies on the table. You will need a surveillance camera, as the RVOSD data will be laid over the PAL or NTSC signal. You can eliminate using wireless video transmission, but it is always good to keep both the video and RC transmitter on in order to test against any possible electronic interference.
Using RVOSD, waypoint number 0 is the target and waypoints 1,2,3,4 are recently hit waypoints sorted by age. Home position is marked H and corresponds to takeoff position evaluated by RVOSD.
The information can be displayed on a local TV set. You can also verify IMU response this way. Refer to FLEXIPILOT OSD manual.

Because the OSD connector can output NMEA data, you can use it for interfacing to any map viewing software during the simulation. Refer to FLEXIPILOT OSD manual for examples.

Using data logging during simulation

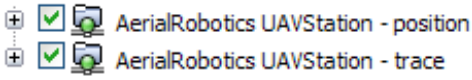
The FLEXIPILOT contains both running data log and even log. Both are fully functional during the simulation and you can download and examine the data contained after the virtual flight. Refer to logging and data processing manual.

Using 3D display in Google Earth

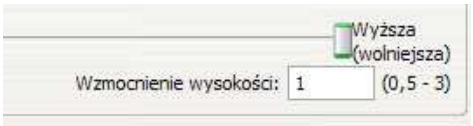


In addition to the mission simulator, you can use free Google Earth application in order to display the artificial trace flown every 1s. This is a very useful for determining obstacle avoidance and in order to evaluate environmental restrictions and hazards, like crossing a river or flying close to inhabited areas.

In order to use the realtime display, drag&drop two kml files bundled with FLEXIPILOT software to Google Earth application: *UAVStation - position.kml* and *UAVStation - trace.kml.* This will enable Google Earth periodic uploading of another two temporary files being written periodically by the UAVStation itself: the airplane icon will effectively move as the simulation progresses.
As a result those 2 positions will appear on the left panel. You can right-click those objects in order to change their properties.



In menu Tools/ Options…/3D View tab you can select altitude amplification to 1 in order to improve the perception of the track geometry.



Simulation setup

In order to improve simulation accuracy and get the feeling about real-world conditions, use the following variables:

SIM_AVG_AIRSPEED_KMH
Possible values: 10 …500 (default 55km/h)
Average airspeed of the airplane during simulation.
It will be altered by manually pitching the airplane during simulation.

SIM_COURSE
Possible values: 0 …360 (default 0deg)
Initial course at which the plane is being launched during simulation.

SIM_WIND_SPEED
Possible values: 0 …100 (default 0km/h)
The wind speed. Useful for tuning track follower.

SIM_WIND_HEADING
Possible values: 0 …360 (default 0deg)
The direction at which wind vector is pointing. This is 180deg opposite to METAR report.

SIM_LAT
Possible values: -90…90
SIM_ LON
Possible values: -180…180
Takeoff latitude and longitude for the simulation.
Particularly important for testing absolute coordinates.
If both are null, WAYPOINT_TARGET_LAT/LON is used instead.
If the above is also null, WAYPOINT_TAKEOFF_LAT/LON is used instead.
If the above is also null, WAYPOINT_HOME_LAT/LON is used instead.

SIM_ALTITUDE_AGL
Possible values:-30000 …30000 (default 0m)
Initial altitude above takeoff point during simulation.

SIM_CLIMBRATE_MIN
Default: -1m/s
SIM_CLIMBRATE_MAX
Default: 2m/s
Possible values: -100 …100m/s (default 0deg)
Set limits depending on platform capabilities, useful for testing approach list.

Also consult installation manual, HUMI_RHGUESS

Example navigation and emergency logic setup file

Consider the following values

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ ADVANCED LOGGING SETTINGS

@@@WAYPOINT_OFFSET=0
@@@WAYPOINT_SIZE=4096

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ WAYPOINT MANAGEMENT AND NAVIGATION

@@@WAYPOINT_AUTOLAND=0
@@@WAYPOINT_HOMEMODE=0 //0=Takeoffhome 1=FixedHome 2=RemoteFixedHome 3=FixedOrTakeoff
@@@WAYPOINT_HITMODE=0 // 0=direct 1=para
@@@WAYPOINT_TAKEOFF_LAT=11.455200
@@@WAYPOINT_TAKEOFF_LON=27.579630
@@@WAYPOINT_HOME_LAT=31.123456
@@@WAYPOINT_HOME_LON=47.123456
@@@WAYPOINT_TARGET_LAT=31.123456
@@@WAYPOINT_TARGET _LON=47.123456
@@@WAYPOINT_TAKEOFF_OFFDIST=100
@@@WAYPOINT_TAKEOFF_OFFHEAD=-360
@@@WAYPOINT_ROTATION=0

@@@WAYPOINT_CYCLES=1
@@@WAYPOINT_BEGIN=0
@@@WAYPOINT_END=0
@@@WAYPOINT_ITERSTYLE=0 // // 0=up 1=down 2=up-down 3=down-up 4=uphook 5=downhook more …
@@@WAYPOINT_SPECIALMODE=0 // 0=navionly 1=mustclimb 2=mustdescend 4=musthit  8=windrotates
16=climbbyloiter
@@@WAYPOINT_NAVIMODE=2    // 0=none 1=carrot 2=follower
@@@WAYPOINT_ALTIMODE=3    // 0=none 1=hold 2=step 3=smooth

@@@WAYPOINT_APPROACH_MODE=0 // 0=go to homepos, 1=use landing list
@@@WAYPOINT_APPROACH_BEGIN=0
@@@WAYPOINT_APPROACH_END=0

@@@WAYPOINT_HOME_DISTREACHED=80
@@@WAYPOINT_HOME_HOLDHEADING_RADIUS=0
@@@WAYPOINT_SATISFY_RADIUS=80
@@@WAYPOINT_SATISFY_MUST_CLIMB=10
@@@WAYPOINT_SATISFY_MUST_DESCEND=10
@@@WAYPOINT_ROUNDCORNER_RADIUS=50 // can be 0..100
@@@WAYPOINT_HALFPLANE_RADIUS=200

@@@NAVI_TRACK_CORRIDOR_WIDTH=0
@@@NAVI_TRACK_CONVERGENCE_WIDTH=0.1

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ MISSION SIMULATION

@@@SIM_AVG_AIRSPEED_KMH=40
@@@SIM_COURSE=0
@@@SIM_WIND_SPEED=0
@@@SIM_WIND_HEADING=90
@@@SIM_LAT=51.123456
@@@SIM_LON=17. 123456
@@@SIM_ALTITUDE_AGL=0
@@@SIM_CLIMBRATE_MAX=2.5
```

```
@@@SIM_CLIMBRATE_MIN=-1.3

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ WAYPOINTS
@@@WPTWRITE0:+0.00,-0.20,100,0,1,0

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ RETURN HOME CONDITIONS

@@@RETHOME_CHANNEL=-1                    //Glitch detection channel not defined
@@@RETHOME_RXOK_MIN=18000                //Glitch detection valid range
@@@RETHOME_RXOK_MAX=20000                //Glitch detection valid range
@@@RETHOME_MAXDIST=1                     //km
@@@RETHOME_MAXLOSDIST=1                  //km
@@@RETHOME_MAXTRIP=10                    //km
@@@RETHOME_MINAGL=80                     //That alt triggers rethome
@@@RETHOME_MINAGL_NOCLIMB=-1e6           //That alt triggers rethome and limits target altitude
@@@RETHOME_MINAGL_ARMABOVE=90            /Enables RETHOME_MINAGL checking
@@@RETHOME_MAXAGL=500                    //Above takeoff point
@@@RETHOME_MAXMSL=1000                   //Above Mean Sea Level
@@@RETHOME_MAXFL=0                       //Flight Level – checking disabled
@@@RETHOME_MAXTIME=1800                  //s
@@@RETHOME_MINLOSANGLE=-60               //disabled (-60deg down cone)
@@@RETHOME_CLIMBTOAGL=50                 //The plane will climb to this when rethoming
@@@RETHOME_CRUISESPEED=30                //Worst case glide speed
@@@RETHOME_WINDSPEED=0                   //km/h - worst case assumption
@@@RETHOME_WINDHEADING=0                 //wind vector heading - worst case assumption
@@@RETHOME_MAXTIME_RESERVE2BASE=60       //s
@@@RETHOME_MAXTIME2BASE=600              //s
@@@RETHOME_RXBAD_TIMEOUT=1               //Glitch detection channel timeout
@@@RETHOME_HEARTBEAT_TIMEOUT=-1          //-1 disabled
@@@RETHOME_VOLTMIN1=9                    //9V cutoff – set above ESC level
@@@RETHOME_VOLTMIN2=0                    //Secondary voltage checking disabled
@@@RETHOME_ZONE_MIN=-1                   //Zone checking disabled
@@@RETHOME_ZONE_MAX=-1                   //Zone checking disabled
@@@RETHOME_AHBATCAP1=0                   //Battery consumption counting disabled
@@@RETHOME_AHBATCAP2=0                   //Battery consumption counting disabled
@@@RETHOME_WHBATCAP1=0                   //Battery consumption counting disabled
@@@RETHOME_WHBATCAP2=0                   //Battery consumption counting disabled
@@@RETHOME_AMPMAX1=0                     //Motor failure protection
@@@RETHOME_AMPMAX2=0                     //Motor failure protection
@@@RETHOME_WATTMAX1=0                    //Motor failure protection
@@@RETHOME_WATTMAX2=0                    //Motor failure protection
@@@RETHOME_RPM1MIN=0                     //Motor failure protection
@@@RETHOME_RPM1MIN_THRMIN=0              //Check RPM1 min when THR above
@@@RETHOME_RPM1MIN_THRBMIN=0             //Check RPM1 min when THRB above
@@@RETHOME_RPM1MAX=0                     //Motor failure protection

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ VOLTAGE MONITORING

@@@VOLTAGE_MOTORCUT1=9
@@@VOLTAGE_MOTORCUT2=0
@@@VOLTAGE_SYSTEMLOW1=7
@@@VOLTAGE_SYSTEMLOW2=0
@@@VOLTAGE_MINTAKEOFF1=10.5
@@@VOLTAGE_MINTAKEOFF2=0

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ OPERATION MODE

@@@FB_THR_DIRECT=0
```

```
@@@FB_TUNING_MODE=0 //0  - NORMAL, 1 - RUDD, 2 - ELEV, 3 - NAVI
@@@MODE_CHANNEL=6
@@@LOITER_MODE=1 // at midpos: 0-noloiter,1-inplace,2-t/o,3-home,4-tgt
@@@TAKEOFF_MODE=5 //1+4: when throttle high in auto mode or using command
@@@TAKEOFF_DURATIONCNT_MAX=60 //3s at 20Hz
@@@RCADJ_CHANNEL1=-1
@@@RCADJ_CHANNEL2=-1
@@@RCADJ_CHANNEL3=-1
@@@PWMOUT_MODE=0
@@@MISPROG_MODE=0
@@@MISPROG_CHANNEL=-1
@@@MISPROG_RCMIN=10000
@@@MISPROG_RCMAX=20000
@@@MISPROG_RCIDMIN=0
@@@MISPROG_RCIDMAX=2
@@@LOITER_CLIMBRADIUS=0.05
@@@LOITER_DESCENTRADIUS=0.05
@@@LOITER_RADIUS=0.03
@@@LOITER_RADIUS_MIN=-0.1
@@@LOITER_RADIUS_MAX=0.1
@@@LOITER_RADIUS_INCHANNEL=-1
@@@LOITER_RADIUS_INCHANNEL_MIN=11000
@@@LOITER_RADIUS_INCHANNEL_MAX=18400
@@@WLOITER1_RADIUS=0.1
@@@WLOITER1_TIME=-30
@@@WLOITER2_RADIUS=-0.1
@@@WLOITER2_TIME=162345.600
@@@WLOITER3_RADIUS=0.3
@@@WLOITER3_TIME=0

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ RC CONTROL - RX OVERRIDE

@@@RXOVR_MODE=0                              //0=always possible, 1=only when conditions met, 2=disable
                                             //protection against RC hijack not used
@@@RXOVR_SELECTOR_CHANNEL=-1                 //Multi-UAV selector not used
@@@RXOVR_SELECTOR_CHANNEL_MIN=15000          //Specific UAV selection range
@@@RXOVR_SELECTOR_CHANNEL_MAX=24000          //Specific UAV selection range
@@@RXOVR_MAXDIST=4                           //km protection against RC hijack at extreme range
@@@RXOVR_MAXLOSDIST=4                        //km protection against RC hijack at extreme range
@@@RXOVR_MAXAGL=4000                         //m protection against RC hijack at extreme altitude
@@@RXOVR_ASSIST_MAXLOSDIST=5.0               //km RC assisted mode distance limit

@@@RXOVR_ASSIST=3                            //2 - Could use ENCHANNEL for assisted mode
                                             //1 - Can use direction and altitude override
@@@0=off
@@@1=during auto
@@@2=when channel enabled

@@@RXOVR_ASSIST_ENCHANNEL=-1                 //Not used
@@@RXOVR_ASSIST_ENCHANNEL_MIN=15000
@@@RXOVR_ASSIST_ENCHANNEL_MAX=22000

@@@RXOVR_ASSIST_DIRCHANNEL=0
@@@RXOVR_ASSIST_DIRCHANNEL_MIN=-4000         //Full left is at -4000 units
@@@RXOVR_ASSIST_DIRCHANNEL_MID=0             //Center position self-trimming in manual mode
@@@RXOVR_ASSIST_DIRCHANNEL_MAX=4000          //Full right is at +4000 units
@@@RXOVR_ASSIST_DIRCHANNEL_ALTMIX=0          //Mix with the output of ALTCHANNEL for Vtail

@@@RXOVR_ASSIST_ALTCHANNEL=1
@@@RXOVR_ASSIST_ALTCHANNEL_MIN=3000          //Full descending is at +3000 units
@@@RXOVR_ASSIST_ALTCHANNEL_MID=0             //Center position self-trimming in manual mode
```

```
@@@RXOVR_ASSIST_ALTCHANNEL_MAX=-3000            //Full climbing is at -3000 units
@@@RXOVR_ASSIST_ALTCHANNEL_DIRMIX=0             //Mix with the output of ALTCHANNEL for Vtail

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@ PARACHUTE FAILSAFE

@@@PARA_OUTPUT_INACTIVE_SERVO=10000
@@@PARA_OUTPUT_ACTIVE_SERVO=20000
@@@PARA_DEPLOY_CHANNEL=-1
@@@PARA_OUTCHANNEL=-1
@@@PARA_DEPLOY_CHANNEL_MIN=17000
@@@PARA_DEPLOY_CHANNEL_MAX=22000
@@@PARA_ARMABOVE=1e6 //80
@@@PARA_NODEPLOY_ABOVE=100
@@@PARA_DEPLOY_BELOW=60
@@@PARA_NODEPLOY_BELOW=30
@@@PARA_DEPLOYMENT_DELAY=0
@@@PARA_WIGGLERTIME=128
@@@PARA_WIGGLERMOVE=200
@@@PARA_MAXDIST=1e6
@@@PARA_ZONE_MIN=-1
@@@PARA_ZONE_MAX=-1
@@@PARA_DEPLOY_GPSDEAD_MAXTIME=30
@@@PARA_DEPLOY_GPSDEAD_MINDISTANCE=1e6          //Do not deploy when GPS dead
```

Under those conditions, the airplane is expected to fly 200m (-0.2km at **WPTWRITE0**) west from takeoff point, then come back to takeoff point (**WAYPOINT_HOMEMODE** =0), the manual override and return home is disabled (**@@@RETHOME_CHANNEL=-1, @@@MODE_CHANNEL=-1**), but the plane will head home if it is more than 1km away from takeoff point or 600s timeout elapses. During typical use, if the autopilot is already tuned for given platform, only the variables in **BOLD and with YELLOW marking** are most important for the everyday user. The user is not expected to change NORMAL TEXT variables as they are standard for wide range of applications.

Example logging setup file

LOG_SOURCE //0=std, 1=fromgps, 2=togps, 3=consolein, 4=consoleout, 5=pid
LOG_AUTOSTART //0=disabled, 1=boottime, 2=takeoff

Classic logging. start during takeoff, 2hz
@@@LOG_AUTOSTART=2
@@@LOG_PERIOD=10
@@@LOG_SOURCE=0
@@@LOG_DUMP_MODE=4
@@@LOG_MAXTIME=0

Classic logging, start at boottime, 120s, full rate 20Hz
@@@LOG_AUTOSTART=1
@@@LOG_PERIOD=1
@@@LOG_SOURCE=0
@@@LOG_MAXTIME=120

Console logging. start during takeoff
@@@LOG_AUTOSTART=2
@@@LOG_PERIOD=1
@@@LOG_SOURCE=4
@@@LOG_MAXTIME=0

PID logging, 5Hz (note, PID logging is recorded only in automatic mode anyway, but not before the log is enabled)
@@@LOG_AUTOSTART=2
@@@LOG_PERIOD=4
@@@LOG_SOURCE=5
@@@LOG_MAXTIME=0

Entire GPS output. start at boottime
@@@LOG_AUTOSTART=1
@@@LOG_PERIOD=1
@@@LOG_SOURCE=1
@@@LOG_MAXTIME=0

The waypoint logic

Each waypoint consists of 2 parameters defining the coordinates plus desired altitude, desired airspeed (reserved for future use), position mode (defining how the coordinates are evaluated) and trigger options. In order to write a waypoint at specified memory position, write a command like this on the console:
@@@WPTWRITE9:270,0.20,50,0,4,15
In this example, the waypoint #9 orders flying 0.2km with the course of 270deg, with target altitude 50m above takeoff point, airspeed target set to 0, style=4: the first 2 parameters {270,0.20} are relative to takeoff point, trigger flags=15: both triggers will be enabled when reaching this waypoint.

The simplest waypoint syntax, using waypoint number 0, you can use GPS coordinates in decimal format:
@@@WPTWRITE*0*:71.12345,51.12345,100,0,0,0
(absolute longitude=71.12345, latitude=51.12345, altitude=100m AGL – Above Ground Level)

Similar case, but using absolute altitude
(600m with style flag 256 is MSL – above Mean Sea Level – bit8 is on)
@@@WPTWRITE*0*:71.12345,51.12345,600,0,256,0

Additionally, it is possible to rotate all heading evaluation and dx/dy logic using WAYPOINT_ROTATION.
This option by design is not affecting waypoint style 6 (relative to previous by (hdg,dist)).

Syntax:

@@@WPTWRITE*wptnumber*:*param1*,*param2*,*AGL*,*airspeed*,*style*,*triggers*

wptnumber:

The waypoint numbering starts with 0.
Therefore with max 128 waypoints (observe WAYPOINT_SIZE) the waypoint number can be 0-127.

param1, param2:

Depending on style parameter, those can contain any of the following:
-earth coordinates: latitude, longitude (eastern longitudes are positive)
-dx, dy distance in km (relative distance along Xaxis W→E and Y axis S→N)
-heading[degrees], distance[km]
relative coordinates defined by degrees 0-360, 0 is north, clockwise
distance can be negative, reversing direction (adds 180deg offset)

***Altitude*:**

* By default:: **ATL**, Above Takeoff Level altitude (in practice close to **AGL**, Above Ground Level) in meters (-32000…32000). Negative altitudes are correct as they allow taking off from the hilltop.
* If the style has absolute altitude flag set, this number will be altitude in meters above sea level (**MSL** altitude). Using absolute altitudes is less accurate as depends on GPS altitude, rather than barometric altitude referenced to takeoff point.

You can expect 1m constant drift per 15min of flight worst case using barometric altitude, or up to 10m variation from GPS if the constellation is unlucky.
* If altirel2previous flag is set, the altitude is added numerically to the last commanded altitude. Putting 0 altitude will simply keep the last commanded value, also when flying in a waypoint loop. Note, however, the style can simultaneously change from ATL to MSL generating completely unexpected yet precisely defined values. Use advanced altitude flags with caution.

***Airspeed*:**

Desired airspeed in km/h (0-500)
(reserved for future use as the autopilot manages airspeed automatically for maximum endurance)

***Style*:**

Position decoding mode:

0 absolute (lat, lon in decimal format)
1 relative to **takeoff** by (dx,dy in km)
2 relative to **home** by (dx,dy in km)
3 relative to **previous** by (dx,dy in km)
4 relative to **takeoff** by (hdg,dist in deg, km)
5 relative to **home** by (hdg,dist in deg, km)
6 relative to **previous** by (hdg,dist in deg, km)
7 relative to **target** by (dx,dy in km)
8 relative to **target** by (hdg,dist in deg, km)

Style flags:

Waypoint style bits:	***15*** *{32768dec}* ***Unused, keep 0***	***14*** *{16384dec}* ***Unused, keep 0***	***13*** *{8192dec}* ***Unused, keep 0***	***12*** *{4096dec}* ***Unused, keep 0***	***11*** *{2048dec}* ***WLoiter Bit1***	***10*** *{1024dec}* ***WLoiter Bit0***	***9*** *{512dec}* ***MustKnow Wind***	***8*** *{256dec}* ***AltiAbsMSL***
Waypoint style bits:	***7*** *{128dec}* ***AltiRelPrevious***	***6*** *{64dec}* ***Must hit***	***5*** *{32dec}* ***Must descend***	***4*** *{16dec}* ***Must climb***	***3*** *{8dec}* ***Style ID bit3***	***2*** *{4dec}* ***Style ID bit2***	***1*** *{2dec}* ***Style ID bit1***	***0*** *{1dec}* ***Style ID bit0***

add flags to style ID in order to achieve special functionality at that waypoint:
16 must climb – will circle around waypoint until the altitude is **not less** than altitude specified at this waypoint (minus altitude precision band, 10m)
32 must descend – will circle around waypoint until the altitude is **not more** than altitude specified at this waypoint (plus altitude precision band, 10m)
Both 'must climb' and 'must descend' conditions are not evaluated in parallel with horizontal hitting of the waypoint: it is sufficient for the plane to reach the waypoint horizontally, then to meet the altitude band in order to proceed to the next waypoint. This allows flying in such a way, that the 2D position is approximate (due to plane loiter radius) and altitude band is enforced. In order to enforce the plane to go to desired 2D position as close as possible, place two consecutive waypoints at the same position (the second one doesn't have to require climbing or descending).
64 must hit – will circle around until the position is precisely over waypoint (+/-10m). Might require several approaches depending on weather and platform agility. Generally not recommended outside of visual range.

128 – altirel2previous - the altitude is expressed in meters
relative to the last target altitude
256 – altiabsmsl – the altitude is expressed in meters above mean sea level
512 – mustknowwind – before proceeding, the plane will loiter until at least 120s of wind speed samples is collected. This option is useful for defining a pattern with a direction dependent on actual wind. Consult WAYPOINT_SPECIALMODE.
1024 + **2048** – both WLoiter bits set (bit1 and bit0) is binary number 3, that means the plane will start loitering at that waypoint using WLOITER slot #3 using WLOITER3_TIME and WLOITER3_RADIUS.

Heading evaluation:
For each Style ID that is using heading, the heading to the next waypoint can be evaluated using different rules.
The heading is a periodic function with a period of 360deg, therefore both 90deg and 90+360=450deg normally mean 'to the right'. However, a different evaluation method is attributed to those values, allowing recursive patterns with little mission programming effort.

Randomized headings use special rule: using a uniform distribution, an angle is generated in a given range. For example, 4090 means an angle span of 90 degrees, in other words -45 to 45deg will be generated. This value will be added to the last heading between waypoints, creating a new course.
Using waypoints relative to wind heading is useful for scripting automatic landings.

0-360	absolute heading
1000-1360	heading relative to takeoff course
2000-2360	heading relative to wind heading using most recent wind estimation
3000-3360	relative to last heading between waypoints
4000-4360	random heading relative to previous heading
5000-5360	random heading relative to takeoff course
6000-6360	random heading relative to wind heading using most recent wind estimation

Example:
(replace n by your desired waypoint slot)
You can set WAYPOINT_TARGET_LAT and WAYPOINT_TARGET_LON using available digital maps,
If you want to fly to a position 4.5km East from that position, use style =8, hdg=90, dist=4.5 as follows:
`@@@WPTWRITEn:90,4.5,100,0,8,0`
Example:
You want to fly 1234 meters North and 789 meters East relative to takeoff position, to altitude 567 MSL, and circle at that position until the altitude is achieved:
`@@@WPTWRITEn:1.234,0.789,567,0, 305,0`
Note the style parameter is: 305=1+16+32+256
Example:
Fly to a random point on a half-circle, located in front of you using takeoff course:
`@@@WPTWRITEn:5180,1,100,0,4,0`
Observe: Style ID=4 is relative position to takeoff, using (hdg,dist).
Distance is 1km. Heading is 5180, what means a random number between -90 and 90deg, added to takeoff course (which is fixed and detected upon takeoff).
Altitude is 100m AGL.
Example:
Rotate flight pattern by 13.2 deg right:
`@@@WAYPOINT_ROTATION=13.2`

Note: The meaning of waypoint flags is decoded in detail and shown in Google Earth following WPTITERALL command. Consult software manual for details.

***Triggers*:**

Trigger parameter defines what actions will be taken when waypoint is met

3	TRIG1ON
1	TRIG1OFF
12	TRIG2ON
4	TRIG2OFF
7	1ON, 2OFF
13	2ON, 1OFF
15	both on
5	both off
16	Deploy parachute
32	Arm parachute (will open when plane descends below safe altitude)
64	Disarm parachute (stop altitude monitoring)
128	Disable Throttle
256	Enable Throttle
512	Disable ThrottleB
1024	Enable ThrottleB
2048	Radio Silence until next waypoint (modem off)
4096	Disable manual mode until next waypoint (RXOVR disable)

Trigger parameter defines what actions will be taken when waypoint is met.
Trigger parameters are in fact binary values:

trigger bits:	***15*** *{32768dec}* ***Reserved, Keep 0***	***14*** *{16384dec}* ***Reserved, Keep 0***	***13*** *{8192dec}* ***Reserved, Keep 0***	***12*** *{4096dec}* ***RXOVR Disable during 1 leg***	***11*** *{2048dec}* ***Radio Silence during 1 leg***	***10*** *{1024dec}* ***THRB on***	***9*** *{512dec}* ***THRB off***	***8*** *{256dec}* ***THR on***
trigger bits:	***7*** *{128dec}* ***THR off***	***6*** *{64dec}* ***disarm parachute***	***5*** *{32dec}* ***arm parachute***	***4*** *{16dec}* ***deploy parachute immediately***	***3*** *{8dec}* ***Trigger2 Enable/Disable***	***2*** *{4dec}* ***Trigger2 Change State***	***1*** *{2dec}* ***Trigger1 Enable/Disable***	***0*** *{1dec}* ***Trigger1 Change State***

Example:
When flying over forest, you might want to disable parachute deployment at this specific flight leg.
Use binary value of
0100 0000bin=64dec
…followed by re-enabling at next waypoint:
0010 0000bin=32dec
Example:
Enable Trigger1, disable Trigger2:
0000 0111bin=7dec
Example:
Deploy Parachute for landing:
0001 0000bin=16

Setup variables

Waypoint list management

WAYPOINT_CYCLES
Possible values: 0 …60000 (default 1)
Defines how many times the waypoint list has to be iterated before returning home.

WAYPOINT_APPROACH_MODE
Possible values: 0 …1 (default 0)
0 – Proceed directly to home position
1 – Use approach list before proceeding to home position

WAYPOINT_BEGIN
WAYPOINT_END
Possible values: 0 …WAYPOINT_SIZE-1
The largest and the smallest index of the waypoint on the list.
Always WAYPOINT_END>= WAYPOINT_BEGIN. Note that despite its name, if WAYPOINT_ITERSTYLE is DOWN or DOWNUP, the first waypoint is actually WAYPOINT_END.

WAYPOINT_APPROACH_BEGIN
WAYPOINT_APPROACH_END
Possible values: 0 …WAYPOINT_SIZE-1
The waypoints forming approach list. This one is always iterated in order BEGIN…END.
After terminating the list, the landing logic is executed. It is possible to include parachute opening or motor shutdown in this list.

WAYPOINT_ITERSTYLE
Possible values: 0 … 9 (default 0)
0 – UP: count UP (WAYPOINT_BEGIN to WAYPOINT_END)
1 – DOWN: count DOWN
2 – UPDOWN: count UP then DOWN
3 – DOWNUP: count DOWN then UP
4 – UPHOOK: count UP, but next cycles will skip WAYPOINT_BEGIN
5 – DOWNHOOK: count DOWN but next cycles will skip WAYPOINT_END
6 – UPDOUBLEHOOK: count UP, but next cycles will skip 2 waypoints
7 – DOWNDOUBLEHOOK: count DOWN but next cycles will skip 2 waypoints (from the end)
8 – UPTRIPLEHOOK: count UP, but next cycles will skip 3 waypoints
9 – DOWNTRIPLEHOOK: count DOWN but next cycles will skip 3 waypoints (from the end)

Note that in UP-DOWN or DOWN-UP mode,
a single cycle has naturally 2 times more waypoints than in other modes.

WAYPOINT_ROTATION

Possible values: -360 …360deg

Rotates all waypoint evaluation commands by specified angle. Absolute waypoints are not adjusted. Each rotation occurs in horizontal plane with axis of rotation defined by specific style of waypoint evaluation. For example, flying 0.1km north relative to WAYPOINT_TARGET_LAT will be rotated relative to this position.

This variable can be set in flight, after reaching a waypoint and loitering until wind estimation is known, when a flag MustKnowWind is given set for given waypoint and WAYPOINT_SPECIALMODE has flag WindImpliesRotation. Then, new effective waypoint pattern rotations will be calculated by summing actual wind heading and preset WAYPOINT_ROTATION.

Example:

One can create a pattern and then rotate it with this variable to make sure a specific part of flight plan is for example directed to the north. Suppose it requires WAYPOINT_ROTATION=46deg to be north-aligned.

Suppose the wind heading found at new waypoint (the direction at which the wind blows) is 350deg. The pattern will be rotated in such a way, that the part normally facing north will be rotated downwind, that is, by 350+46=36deg to the right.

You can change WAYPOINT_ROTATION before takeoff to 46+180=226deg, then upon reaching a waypoint with wind rotation enabled, a specific part of the flight plan will be facing against the wind.

Note:

For designating landing approach direction only, use waypoint style relative to certain position, using heading modified so that it implies wind direction by itself. Therefore global flight plan will be preserved.

Loitering at waypoint

Upon reaching given waypoint, the plane can loiter with given radius, in given direction, for given duration or until UTC time elapses (no more than 12h) or indefinitely while maintaining waypoint altitude. RETHOME conditions can force exiting this loiter, the same applies to MDO_CHANNEL switching from MISSION to LOITER and then again to MISSION. You can continue with command @@@WLOITERDONE or setting shorter loiter elapse time with @@@WPTLOITERTIME. The radius and direction can be changed with RC channel as during normal loiter. Analogically, the altitude can be adjusted with RXOVR altitude channel during loiter mode.
There are 3 WLOITER slots with custom set of parameters, a specific slot is chosen using waypoint style bits (see WPTWRITE command).
At any time, you can use @@@WPTWLOITERn (n=1-3) command in order to enforce specific loiter slot parameters to the actual loiter.

WLOITER1_RADIUS
WLOITER2_RADIUS
WLOITER3_RADIUS
Possible values: -1 …1km (default 0km)
Positive or 0 – clockwise
Negative – counterclockwise
This option sets the loitering that may or may not be satisfied as a function of plane agility. You can adjust this value in realtime using RC channel or modem command.
0 is minimal achievable radius, clockwise.
-1e-6 is minimal achievable radius, counterclockwise.

WLOITER1_TIME
WLOITER2_TIME
WLOITER3_TIME
Possible values: -1e6s …236000 (default 0 - infinite)
Sets loiter time:
0 infinite
Negative values – remaining time to stay in loiter, in seconds (approx 277hrs max)
Positive values: stay in loiter until UTC time elapses, in the format: HHMMSS.
Loiter until UTC midnight is 236000 (23h 60min 00s).

Mission programming

Each mission consists of WAYPOINT_BEGIN, WAYPOINT_END, WAYPOINT_CYCLES and WAYPOINT_ITERSTYLE.
You can change those variables using a rotary mission selector, RC channel or by executing a command from the console. This feature allows simplifying field operations by creating sets of missions that are selectable using mechanical rotary encoder or by keyboard before takeoff.

After takeoff, rotary encoder is ignored, but changing the mission with RC is possible.
Any RC input channel can be used for changing a mission, the ID of mission ranges to be selected must be continuous (ascending, descending or a single mission). If the input channel is not connected or out of input range temporarily, when the signal falls again in the valid range, a mission is selected. This way it is possible to define RC channel to be able to restart any given mission. Selecting a mission takes 0.5s after the input channel value stabilizes.

@@@MISSIONSELECTn
Select mission n=0…127

@@@MISSIONEXECUTEn
Select mission n=0…127 and reset waypoint execution. This allows changing flight plan in flight.

@@@ MISSIONWRITEn:ncycles,wptbegin,wptend[,iterstyle]
or
@@@ MISSIONWRITE n ncycles wptbegin wptend[iterstyle]
Create a mission. Same rules apply as for variables WAYPOINT_BEGIN, WAYPOINT_END, WAYPOINT_CYCLES, WAYPOINT_ITERSTYLE. If iterstyle is not specified, the actual value is implicitly saved. This way, when executing MISSIONSELECT, the ITERSTYLE will be actually loaded and will represent this implicitly saved value.

MISPROG_MODE
Possible values: 0 … 15 (default 0)
Flags:
1 update upon startup (if not, last mission or custom flight plan will be used)
2 must use rotary for automatic takeoff
(must be connected and valid but can remain disabled)
4 enable rotary
8 enable RC

MISPROG_CHANNEL
Possible values: -1 … 11 (default -1)
Select RC channel for changing a mission.
-1 is disabled

MISPROG_RCMIN
Default: 10000

MISPROG_RCMAX
Default: 20000
Possible values: 6000-24000
Select RC receiver channel range for defining switchable missions.
RCMAX must be >= than RCMIN or the mission switching logic is ignored.

MISPROG_RCIDMIN
Default: 0
MISPROG_RCIDMAX
Default: 1
Possible values: 0-127
Defines a range of missions that can be selected using RC.
RCIDMIN can be larger than RCIDMAX, reversing the order of mission list.
Disabling: use MISPROG_MODE

Examples:
On a specific system, when using RC transmitter, you observe that SCAP8 can change from 9520 to 21500.
A. Using CHANNEL=8, RCMIN=9000, RXMAX=22000, RCIDMIN=1 RCIDMAX=2 you can select between 2 missions.
B. Using CHANNEL=8, RCMIN=10000, RXMAX=22000, RCIDMIN=1 RCIDMAX=2 you can use the pre-programmed waypoint list (SCAP input values 9520 to 10000) or select between 2 missions, or restart and execute one of the 2 missions by moving briefly into 9520..10000 zone then back.
C. Using CHANNEL=8, RCMIN=10000, RXMAX=22000, RCIDMIN=15 RCIDMAX=15 you can use the pre-programmed waypoint list and restart mission #15 every time the channel exits, then enters specific zone.
D. Using CHANNEL=8, RCMIN=9000, RXMAX=22000, RCIDMIN=27 RCIDMAX=20 you can switch between 8 missions, numbered in reversed order.

Rotary mission selector

An optional Rotary Mission selector allows selecting a subset of missions using 2 knobs.
The missions are named A1..A6, B1…B6 … to F1..F6.
They map to mission ID 0…5 10..15 … to 50..55, correspondingly.
However, there is as much as 128 possible missions IDs (0..127). The user can still use the mission ID that cannot be accessed using mechanical selectors by entering MISSIONSELECTn command. For example, mission 127 will map to letter M8 but is not related to available mission selector position.

When using mission selector, you can observe the notification in the format:
Mission A2 ID=1: 5 x (9...16,s6) + o
In this case you have selected the mission A2 with ID=1, which spans 5 times waypoints 9 to 16, using ITERSTYLE 6, the mission ends with ‘o’ - loitering overhead (L would mean using the Landing list). The post-mission landing list is defined using WAYPOINT_APPROACH_MODE and is not saved in the mission slot itself.

Takeoff position and automatic takeoff

You can alter the meaning of takeoff position. You can takeoff from a catapult, then extrapolate 'takeoff position' in order to make sure the landing during emergency situation will be in a safe area, yet within visual range.

WAYPOINT_TAKEOFF_OFFDIST

Possible values: 0 …250m (default 0

The distance used for position extrapolation in meters.

WAYPOINT_TAKEOFF_OFFHEAD

Possible values: -360 …359 (default -360)

The course used for position extrapolation can be absolute (using nonnegative values, 0…359) or relative top takeoff course evaluated after the launch (negative values, the angle used for extrapolation will be OFFHEAD+takeoff course)

Examples:

-360 will mean using detected takeoff course-360deg will be used

-270 will result in extrapolating to the right from takeoff course

90 will move takeoff position to the east

WAYPOINT_TAKEOFF_LAT and WAYPOINT_TAKEOFF_LON are updated automatically. It is possible to change those variables for simulation purposes, as they may affect waypoint evaluation.

See VOLTAGE_MINTAKEOFF1 and VOLTAGE_MINTAKEOFF2 for defining safe voltage for automatic takeoff.

TAKEOFF_MODE
Default: 0
Values: 0-15
Flags:
0 manual mode takeoff will be detected when moving more than 20km/h
1 automatic takeoff starts when throttle position is held max
when autopilot is enabled
2 automatic takeoff after button press
4 automatic takeoff by command
8 allow takeoff detection when RC disconnected (by movement)
(act as logger or droppable glider)
This option enables additional methods of initiating a takeoff. It is always possible to takeoff in manual mode, the autopilot will detect moving object as any other takeoff. Post-takeoff climb phase or throttle pre-spin is not executed when takeoff was manual, even if it would be executed during automatic takeoff.

TAKEOFF_DURATIONCNT_MAX
Default: 120 (6s)
Values: 20..400 (1-20s)
The plane will fly straight for a certain time after takeoff. This counter defines the number of seconds to fly straight, multiplied by 32.

When automatic takeoff is selected (TAKEOFF_MODE>0), holding the button is starting slow spinning of the propeller. If you continue holding the button for 2s, the takeoff will begin. At this stage the only method to stop motor will be to switch to manual mode or to disconnect the power. It could be that pressing the button will not spin the propeller and the takeoff could not begin. The reason will be displayed on the console 'TAKEOFF_ERRCODE=XXX', the value is a sum of the following error codes:

```
TAKEOFF_ERRCODE_ALREADY_STARTED=1
    // the takeoff has began, and holding takeoff button is ignored
TAKEOFF_ERRCODE_MANUAL_MODE=2
    // the autopilot is in manual (RC) mode, makes no sense to perform automatic takeoff
TAKEOFF_ERRCODE_BARO_NOT_CALIBRATED=4
    // pressure sensor (altimeter) is not ready, wait 10s
TAKEOFF_ERRCODE_GPS_NOT_READY=8
    // GPS lock is not secure enough for safe navigation
TAKEOFF_ERRCODE_VOLTAGE_TOO_LOW=16
    // VOLTAGE_MINTAKEOFF1 or VOLTAGE_MINTAKEOFF2 are lower than selected thresholds,
    the ESC could cut the motor off during takeoff
TAKEOFF_ERRCODE_USB_CONNECTED=32
    // USB cable is still plugged – this diagnostic is not always present
TAKEOFF_ERRCODE_MISPROG_FAILURE=64
    // mission programmer failure, broken wires or incorrect selection
TAKEOFF_ERRCODE_SETUP_ERRORS=128
    // there are configuration errors, look at console first
TAKEOFF_ERRCODE_PARACHUTE_DEPLOYED=256
    // the parachute latch is open, unsafe to takeoff
```

Target position

WAYPOINT_TARGET_LAT
Possible values: -90…90
WAYPOINT_ TARGET _LON
Possible values: -180…180
Latitude and longitude for fixed home position in decimal format.
Used for waypoint evaluation, can be updated in flight using video head.

Loitering

It is possible to use MODE_CHANNEL to bring the plane back home when in HIGH position. Switching from LOW (mission) to MID position (loiter) can be used for defining loitering point. Also switching from manual mode to automatic, when MODE_CHANNEL is in MID position, defines loiter point. Independently on that, the plane can ignore the actual 'loiter mark' if LOITER_MODE is different from 1. Loiter position still can be set that way and can be used by other logic like stabilized head. When RETHOME condition occurs, loiter point is moved to takeoff point once (but can be set manually again at different location).
Below find a family of commands that can affect actual loitering state.
Besides those commands, you can change loiter diameter and direction using RC channel.

LOITER_MODE
Default: 0
Values:
0 no loiter
1 loiter in place set when autopilot is enabled or toggled from mission to loiter
2 loiter takeoff position (same as MODE_CHANNEL in HIGH position)
3 loiter fixed home position
4 loiter target position

LOITER_RADIUS
Default: 0.05km
Values: -1..1km
Positive or 0 – clockwise
Negative – counterclockwise
This option sets the loitering that may or may not be satisfied as a function of plane agility.
You can adjust this value in real-time using RC channel or modem command.
0 is minimal achievable radius, clockwise.
-1e-6 is minimal achievable radius, counterclockwise.

LOITER_CLIMBRADIUS
Default: 0.1km
Values: -1..1km
Positive or 0 – clockwise
Negative – counterclockwise

If WAYPOINT_SPECIALMODE – climbbyloiter bit is set, this value will be used during climbing, when the 2D position of the waypoint has already been met and waypoint style requires achieving altitude before proceeding.

LOITER_DESCENTRADIUS
Default: 0.1km
Values: -1..1km
Positive or 0 – clockwise
Negative – counterclockwise
If WAYPOINT_SPECIALMODE – climbbyloiter bit is set, this value will be used during descending, when the 2D position of the waypoint has already been met and waypoint style requires achieving altitude before proceeding.

LOITER_RADIUS_INCHANNEL
Possible values: -1 … 11 (default -1)
Select RC channel for changing loiter radius in real-time.
-1 is disabled

LOITER_RADIUS_INCHANNEL_MIN
Possible values: 6000…24000 (default 6000)
LOITER_RADIUS_INCHANNEL_MAX
Possible values: 6000…24000 (default 24000)
Input range of RC channel for setting loiter radius in real time.

LOITER_RADIUS_MIN
Possible values: -1…1km (default -0.1km - counterclockwise)
LOITER_RADIUS_MAX
Possible values: -1…1km (default 0.1km -clockwise)
Output range of RC channel for setting loiter radius in real time.
Ensure MAX>MIN, otherwise disabled.

@@@WPTLOITERHERE
Enter loiter mode in place, the altitude is not updated (last target altitude is conserved)

@@@WPTLOITERTAKEOFF
Enter loiter mode above extrapolated takeoff position (the same as used for magnet home).

@@@WPTLOITERHOME
Enter loiter mode above fixed home position.

@@@WPTLOITERTARGET
Enter loiter mode above target position. The position can be fed using console commands or stabilized head.

@@@WPTLOITER
Enter loiter mode without affecting the position.
May override loiter set by RC transmitter using MODE_CHANNEL.

@@@WPTLOITERDONE
Exits all loiter states and continue navigation.
May override loiter set by RC transmitter using MODE_CHANNEL.

@@@WPTLOITERSETPOS lat,lon
Loiter position indicated by lat and lon. Target altitude is not changed.
Is not entering loiter mode.

@@@WPTLOITERDXDY dx_km,dy_km
Moves loiter position along X and Y axes, relative to last loiter position.
Is not entering loiter mode.

@@@WPTLOITERHDGDIST hdg_deg,dist_km
Moves loiter position dist_km in direction of hdg_deg, relative to last loiter position.
Both values can be negative or larger than 360deg.
Is not entering loiter mode.

@@@WPTLOITERSETRADIUS radius_km
Values: -1..1km
Positive or 0 – clockwise
Negative – counterclockwise
Allows setting the actual loiter radius directly.
Is not entering loiter mode.

@@@WPTLOITERSETAGL alt
Set loiter altitude only, relative to takeoff level.
The value can be negative.
Is not entering loiter mode.

@@@WPTLOITERSETMSL alt
Set loiter altitude only, relative to mean sea level.
The value can be negative.
Is not entering loiter mode.

Automatic landing, home position

WAYPOINT_HOME_LAT
Possible values: -90…90
WAYPOINT_HOME_LON
Possible values: -180…180
Latitude and longitude for fixed home position in decimal format

WAYPOINT_HOMEMODE
Possible values: 0 …3 (default 0)
0 – TAKEOFFPOS return to takeoff position
1 – FIXEDHOMEPOS return to fixed home position defined by
(WAYPOINT_HOME_LAT, WAYPOINT_HOME_LON)
if its distance from takeoff position is less than 0.5km, else use takeoff position
2 – REMOTEFIXEDHOMEPOS return to fixed home position defined by
(WAYPOINT_HOME_LAT, WAYPOINT_HOME_LON)
regardless the distance
3 – TAKEOFF_OR_FIXEDHOMEPOS choose takeoff position or fixed home position, whichever is closer. Prefers takeoff position if about to be equal distance.
4 – Nearest waypoint from the list defined by
WAYPOINT_HOMEPOS_BEGIN... WAYPOINT_HOMEPOS_END
It is recommended to use either mode 0 or 1 in order to prevent 'returning to the wrong home'. Note that RETHOME logic and other failsafe returns always use the takeoff position.

WAYPOINT_APPROACH_BEGIN
WAYPOINT_APPROACH_END
Possible values: 0 …WAYPOINT_SIZE-1
Waypoints defining a list of home positions.

WAYPOINT_AUTOLAND
Possible values: 0 … 1 (default 0)
0 - loiter after returning home
1 - turn motor off upon returning home, glide down in circles
The condition of returning home depends on WAYPOINT_HOME_DISTREACHED

WAYPOINT_HOME_DISTREACHED
Possible values: 5 … 250m (default 80m)
This is the distance for detecting the moment when it is safe to descend above home position. Using too small value will result in several attempts to fly over home position before turning off the motor.

Airstrip landing

Note: for most precise approach, it is best to use approach list and fixed waypoint coordinates. It can be safeguarded by the following logic.

Intelligent navigation options

WAYPOINT_NAVIMODE
Possible values: 0 … 2 (default 2)
0 – Disabled, no direction control
1 – Simple carrot/magnet style navigation
2 –2D track Follower

WAYPOINT_ALTIMODE
Possible values: 0 … 3 (default 3)
0 – Disabled, no altitude control
1 –Altitude hold only
2 – Step target altitude updates
3 – Smooth target altitude updates

WAYPOINT_SPECIALMODE
Possible values: 0 … 31 (default 0)
0 – Can skip waypoints if passed around
1 – Must climb to all waypoints before proceeding
2 – Must descend to all waypoints before proceeding
4 – Must hit precisely all waypoints before proceeding
8 – Wind analysis affects pattern rotation
16 – Change altitude using LOITER_CLIMBRADIUS and LOITER_DESCENTRADIUS
The values are additive (binary flags).
This mode also affects approach list, and is useful mostly for testing or single-waypoint missions.
Flag 8, when set, causes wind heading to be affected to WAYPOINT_ROTATION variable, thus rotating the whole pattern since that waypoint. The affected value is not stored, so the new flight will use the original WAYPOINT_ROTATION variable.

NAVI_TRACK_CORRIDOR_WIDTH
Possible values: 0 … 0.1km (default 0.005km)
When the plane is within this distance to the line connecting 2 waypoints, the track follower is inactive thus reducing any possible course oscillations.

NAVI_TRACK_CONVERGENCE_WIDTH
Possible values: 0 … 1km (default 0.15km)
When the plane is within this distance to the line connecting 2 waypoints, the heading convergence begins. Outside the band, the plane heads straight towards the line joining the 2 waypoints. Typically, this value should not be smaller than aircraft turn radius or course overshoot will result.

WAYPOINT_SATISFY_RADIUS
Possible values: 5 … 250m (default 50m)
If the airplane is closer than this distance and start flying away, next waypoint is selected.

WAYPOINT_SATISFY_MUST_CLIMB
Possible values: 1 … 250m (default 10m)
Altitude band precision when the plane is required to reach prescribed altitude when climbing.

WAYPOINT_SATISFY_MUST_DESCEND
Possible values: 1 … 250m (default 10m)
Altitude band precision when the plane is required to reach prescribed altitude when descending.

WAYPOINT_ROUNDCORNER_RADIUS
Possible values: 0 … 250m (default 0m)
This option can reduce navigational overshot by selecting next waypoint earlier. 0m means flying directly over all waypoints. Values around 30-50m give best results. The value is adjusted in flight depending on weather and airspeed.

WAYPOINT_HALFPLANE_RADIUS
Possible values: 0 … 250m (default 250m)
If missed the waypoint because of wind influence and is closer than this radius, mark the waypoint as unreachable and proceed to next.

WAYPOINT_HITMODE
Possible values: 0-1
0 Must flight above given waypoint precisely
1 The waypoint is reached when parachute landing point is at given location, adjusted for wind
Defines how the position is adjusted when MUSTHIT flag is enabled for given waypoint

Safe loiter, magnet-home

Before departing for a mission, it is possible to takeoff with magnet-home mode enabled. This will keep the plane in view before the navigation stabilizes and the altitude is reached.

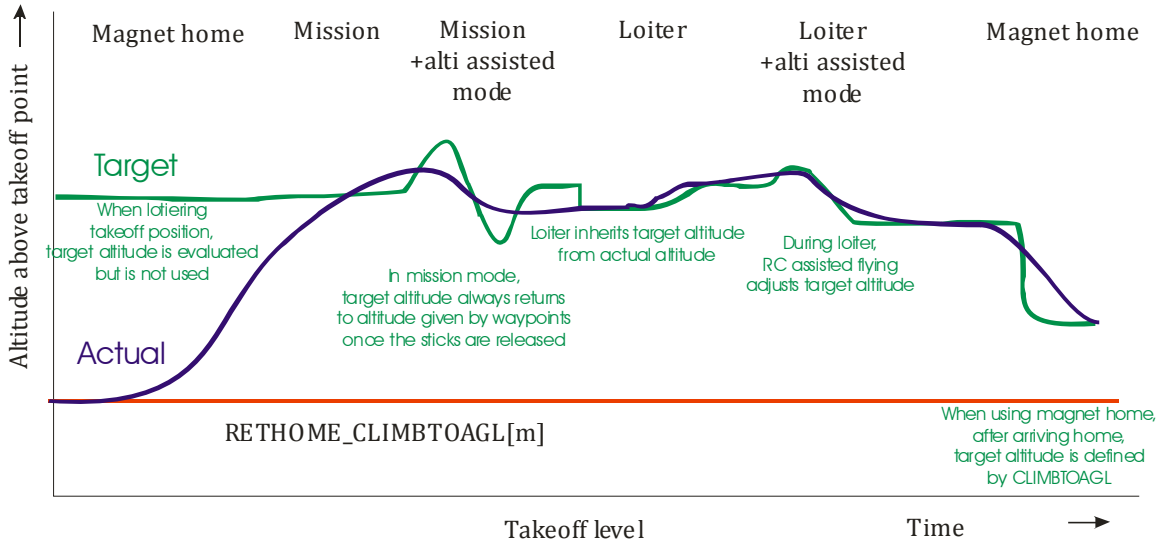
MODE_CHANNEL
Possible values: -1…11 (default -1)
Defines RC input channel used for magnet-home function.
When in HIGH position, the actual navigation is ignored and the plane heads home (Magnet Home).
When in MIDDLE position, the plane enters loiter mode in place.
Changing from LOW to MIDDLE position or enabling the autopilot when in MIDDLE position, will update loiter point. That point is a reference for 'loiter in place' when LOITER_MODE=1.
Releasing the slider/switch resumes navigation, contrary to using RETHOME channel which is 'non-recoverable'.
-1 disables the functionality.

MODE_CHANNEL is also used in tuning modes but in high position always defaults to return-home. It is highly recommended to use this channel and start autonomous flight with failsafe setting in your RC receiver set to return-home using MODE CHANNEL.

Contrary to MODE channel which is a convenient method of testing the airplane's return action before departing for a mission, the RETHOME engine is a mission termination system that can only take its decision once. If RETHOME condition is met at least once, you cannot recover it without returning home physically, the plane must be landed and the autopilot rebooted/power cycled.

LOITER_MODE
Possible values: 0…4 (default 1)
0 – Disabled
1 – In place
2 – Over takeoff (this way, middle position acts always as magnet home)
3 – Over fixed home
4 – Over target position
Defines action to be taken when mode channel is in MIDDLE position.
It is possible to navigate a plane between takeoff position and HOME or TARGET by continuously changing MODE_CHANNEL, if mode 1, 3 or 4 is selected.

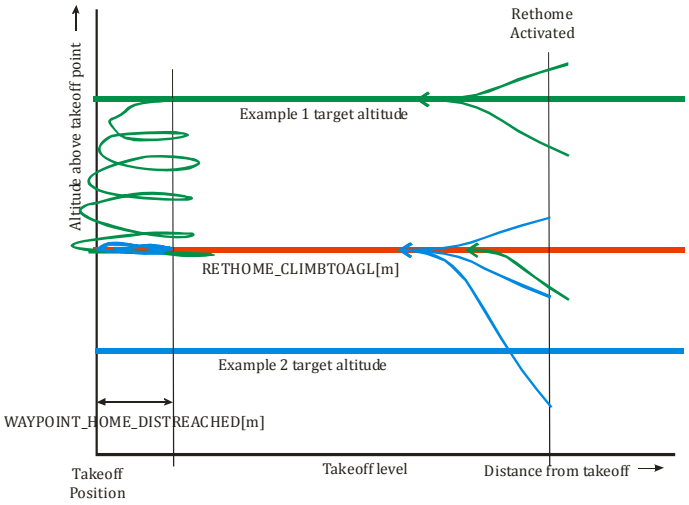
While defining *loiter position* using RC transmitter is self-explanatory, the functioning of *altitude target* differs between modes:



The logic has been defined in order to provide safe handling, however has a counterintuitive consequence. When switching from manual mode to automatic mode with LOITER selected, the altitude is set to the actual altitude of the aircraft. You can change this altitude flying in RXOVR assisted mode. The altitude set this way remembered when toggling between magnet-home and loiter – therefore you can set MODE_CHANNEL to magnet-home by default, and in case of glitches loiter position is not lost.
As a consequence in order to set a new loiter position, you have to select mission mode first, then switch to loiter. It is also possible by entering and exiting manual RC control mode.

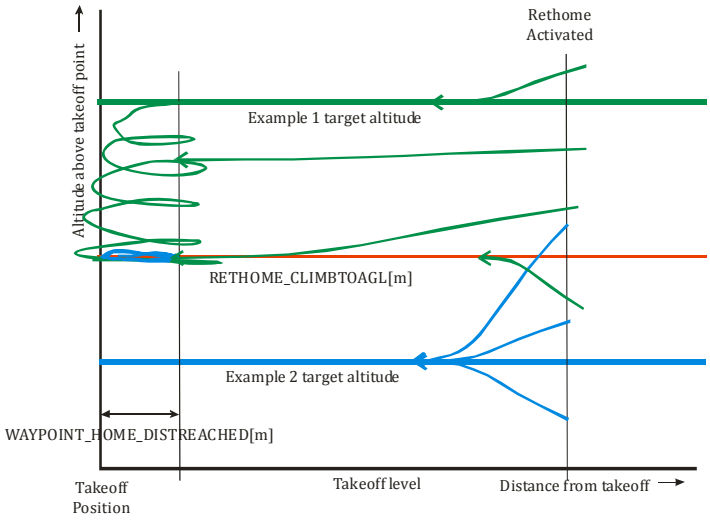
RETHOME mission aborting in case of emergency

The following diagram describes interaction of target altitude calculated from the last waypoint and RETHOME_CLIMBTOAGL, what is minimal safe altitude for loitering overhead. The same logic applies equally when using RETHOME irrecoverable return, as when using Magnet Home mission interruption. Note how the plane is NOT climbing to its waypoint altitude if return home occurred when the plane was physically below minimal altitude (**Green line**).

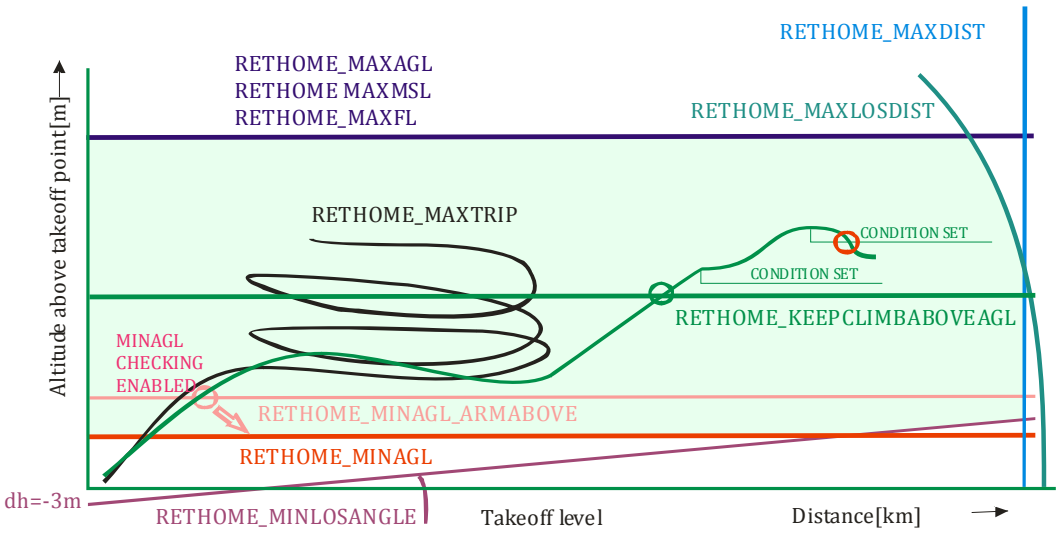


Target altitude during RETHOME

If RETHOME condition indicates inability of the plane to climb, it tries to not cause peak power consumption by maintaining at least CLIMBTOAGL altitude, or the last TARGET ALTITUDE, whichever is less.



Target altitude during RETHOME using '**cannot climb'** scenario



Using security distances and altitude conditions with RETHOME

RETHOME_LOITERRADIUS
Default: 0.05km
Values: -1..1km
Positive or 0 – clockwise
Negative – counterclockwise
This option sets the loitering that may or may not be satisfied as a function of plane agility.
0 is minimal achievable radius, clockwise.
-1e-6 is minimal achievable radius, counterclockwise

RETHOME_CLIMBTOAGL
Possible value: 0…1e6 (default 100m)
The airplane will climb to this altitude when returning from a mission or during emergency.

RETHOME_CHANNEL
Possible value: -1…11 (default -1)
RC input channel number that is used for triggering emergency return home.
That channel should be preprogrammed on the RC receiver using RC transmitter settings so that it has fixed value when the transmitter is turned off. Once the value on that channel goes off specified range, the return home is executed and the navigation is completely stopped. -1 is disabled.

RETHOME_RXOK_MIN
Possible value: 0…24000 (default 12000)
RETHOME_RXOK_MAX
Possible values: 0…24000 (default 18000)
Specifies valid Emergency Return Home RC signal range (1ms is 10000 PWM pulse duty cycle)
0..24000 is always valid for all possible stick positions. When the channel value is OUT of specified range for RETHOME_RXBAD_TIMEOUT seconds, emergency return home begins.
RETHOME_RXBAD_TIMEOUT
-1..1800s (max 30min)

If there is no valid RC signal for that period of time, irrecoverable return home occurs.
0 is immediate, -1 is disabled.

RETHOME_HEARTBEAT_TIMEOUT
-1..1800s (max 30min)
If there are no @@@PING commands received for that period of time, irrecoverable return home occurs. For use with modem only.
0 is immediate, -1 is disabled.

RETHOME_MAXDIST
Possible value: 0…1e6 (default 1km)
Maximum horizontal distance to fly in autopilot mode.
Note: MAXDIST and MAXLOSDIST conditions are also checked during waypoint evaluation. If a waypoint is beyond safe distance, the mission is aborted immediately instead of negotiating the extreme range.

RETHOME_MAXLOSDIST
Possible value: 0…1e6 (default 1km)
Maximum distance along the Line of Sight to fly in autopilot mode.

RETHOME_MAXTRIP
Possible value: 0…1e6 (default 1km)
Maximum distance to travel.
1e6 is disabled.

RETHOME_MINAGL_ARMABOVE
Possible value: -1e6…1e6 (default 180m)
Testign of minimal altitude will begin only after reaching that altitude.
-1e6 is immediate, must enable autopilot above RETHOME_MINAGL in order to avoid mission abort.
1e6 is disabled.
RETHOME_MINAGL
Possible value: -1e6…1e6 (default 170m)
-1e6 is disabled.
RETHOME_MINAGL_NOCLIMB **← Altitude logic assumes 'cannot climb' scenario**
Possible value: -1e6…1e6 (-1e6m, disabled)
-1e6 is disabled.
RETHOME_MAXAGL
Possible value: -1e6…1e6 (default 1e6m)
Minimum and maximum altitude above takeoff point to fly in autopilot mode.
1e6 is disabled.
Note: MINAGL condition is disabled when the mission has been terminated and the plane is flying *approach list* what allows classic landing approach. Similar-looking parachute MINAGL is not affected by landing approach and is still active.

RETHOME_KEEPCLIMBABOVEAGL
Possible value: -1e6…1e6m (default 1e6 - disabled)

Defines minimum altitude, above which the plane must climb in automatic mode (no loss of altitude more than 20m is allowed). The altitude gained using thermals is not counted; only the altitude achieved by following control command or waypoint order increases the current altitude bar. This feature is useful for high altitude flights that also fly far away from takeoff point. Thanks to this option, for several flight scenarios, RETHOME condition can be triggered earlier than using MINAGL logic, assuring safe return in glide mode. 1e6 is disabled.

RETHOME_MAXFL
Possible value: 0...3281 (default 0)
Maximum Flight Level using
-Standard Atmosphere Model and pressure altitude
or
-GPS MSL (uses Earth Geoid approximation of the GPS module)
(whichever is less)
Standard Atmosphere Altitude, to be flown in automatic mode.
Used for complying with aviation traffic. The actual flight level might evaluate to small negative given local pressure.
Typical value around 95 (2897m) or 190 (5795m), depending on air traffic laws.
3281 is roughly 100km.
0 is disabled.

RETHOME_MAXMSL
Possible value: -1e6…1e6 (default 1e6m)
Maximum altitude above Mean Sea Level, evaluated using
-Standard Atmosphere Model and pressure altitude
or
- GPS MSL (uses Earth Geoid approximation of the GPS module)
(whichever is less)
1e6 is disabled.

RETHOME_MAXTIME
Possible value: 0...1e38 (default 1800s)
Maximum flight length in seconds.
1e38 is disabled.

RETHOME_MINLOSANGLE
Possible value: -90…90 (default 0 deg above horizon)
Minimum 'line of sight' cone angle, above which the aircraft must fly. This simulates both visual range and RC balloon-shaped reception range. Negative values allow flying in a valley and takeoff from the mountain.
-90 is disabled.

Remarks

Observe that some conditions: MAXDIST, MAXAGL, MAXFL, MAXMSL are tested also during waypoint evaluation. The reason is that violations of those rules are results of major mistakes in the waypoint plan. Following the flight path until the actual plane position meets the limit is often safe, but unnecessarily misleading about 'true UAV intentions' therefore preemptive evaluation of those conditions is useful for overall safety.

MAXAGL, MAXFL and MAXMSL conditions are tested also during waypoint evaluation. If a waypoint exceeds those limits, the mission is aborted. Other altitude conditions cause appropriate RETHOME action only once they are reached physically.
MAXFL and MAXMSL use either GPS or barometric altitude. While Flight Level by definition is using barometric altitude only, for emergency purposes both GPS and barometer is used for evaluating MAXFL and MAXMSL. Both conditions force returning home if the plane flies too high (supposedly near passenger traffic levels), yet the laws refer usually to barometric altitude only. However, it is safer to evaluate the flight levels and MSL altitudes using both GPS and barometric condition (whichever is more pessimistic), which helps avoid traffic of other planes that did not calibrated their barometers to QNH correctly during low pressure weather and as a consequence fly lower than intended. Those conditions are in fact redundant, but are easier for eliminating mistakes.

Rethome commands

Those commands are to be execute via modem or during mission simulation. They facilitate testing more complex conditions.

@@@PING	Resets PING timeout
@@@RETHOME	Navigate to failsafe home
@@@RETHOMERESET	Reset rethome logic, disarm minagl, clear faults
@@@RETHOMEZONESHOW	Show allowed zones guarded by rethome

Wind model for RETHOME

Pessimistic estimate of wind:
RETHOME_CRUISESPEED
Possible values: 0... NAVI_AVG_AIRSPEED_KMH km/h (default is NAVI_AVG_AIRSPEED_KMH)
Using NAVI_AVG_AIRSPEED_KMH assumes gilde speed equal to cruise speed, therefore use the value some 20% smaller. This variable means that the cruise airspeed might be impossible to exceed that value, worst case.
Using 0 disables pessimistic logic and RETHOME logic will use real time cruise speed estimate and wind estimate, silently assuming the potentially favorable winds will not calm down when returning home.
RETHOME_WINDSPEED
Possible values: 0...100km/h (default equal to SIM_WIND_SPEED)
0 assumes no wind, heading will be ignored.
RETHOME_WINDHEADING

Possible values: 0...359deg
Direction at which the wind blows (navigational wind direction).
Set those variables from worst-case meteo prediction in order to make return home timing calculation more restrictive (safer).

Using the above variables and real time wind estimation, the most pessimistic time to return home will be calculated. Therefore the return home conditions are calculated twice, once assuming RETHOME_WINDSPEED&WINDHEADING and once using real time estimate.
The following conditions are evaluated independently:

1. Worst case time to return home (calculated)
must be less than
total allowed return flight time *(**RETHOME _MAXTIME2BASE**)*
2. Estimated remaining flight time *(**RETHOME_MAXTIME** minus time flown)*
must be less than
worst case time to return home (calculated)
plus
time reserve *(**RETHOME_MAXTIME_RESERVE2BASE**)*

Depending on weather conditions either condition can be stricter.
Condition 1 is protecting mostly against weather change and pilot's lack of experience: return home will be kept reasonably short but the mission radius might be surprisingly short.
Condition 2 is more related to battery capacity: even if battery consumption is estimated incorrectly and voltages are drifting around, total flight time for given platform at constant airspeed is fairly fixed.

RETHOME_MAXTIME2BASE
Possible values: 0...1e38 (default 600s)
Max time to fly to base must not be longer than that. It will change with wind speed. 0 is disabled.

RETHOME_MAXTIME_RESERVE2BASE
Possible values: 0...1e38 (default 60s)
Time reserve used to evaluate critical flight time: this is the time to fly that should remain available. 0 is disabled.

RETHOME_VOLTMIN1 **← Altitude logic assumes 'cannot climb' scenario**
RETHOME_VOLTMIN2 **← Altitude logic assumes 'cannot climb' scenario**
Possible values: 0..32V
Voltage levels below which the plane must return home. Both measurements can use the same or different batteries. The actual scaling factors and acceptable max voltage may differ from installation to installation due to use of voltage divider (raw input is 0-5V). 0 is disabled.

RETHOME_AHBATCAP1
RETHOME_AHBATCAP2
Possible values: 0..100Ah
After consuming more than this capacity, the plane must return home. Both measurements can use the same or different batteries. The actual scaling factors and acceptable max

voltage may differ from installation to installation due to use of voltage divider (raw input is 0-5V).
Also used for calculating remaining Ah capacity and triggering AHRESERVE1, AHRESERVE2 conditions (BINGO fuel condition).
0 is disabled.

RETHOME_WHBATCAP1
RETHOME_WHBATCAP2
Possible values: 0..1e5Wh
After consuming more than this capacity, the plane must return home. Both measurements can use the same or different batteries. The actual scaling factors and acceptable max voltage may differ from installation to installation due to use of voltage divider (raw input is 0-5V).
Also used for calculating remaining Wh capacity and triggering AHRESERVE1, AHRESERVE2 conditions (BINGO fuel condition).
0 is disabled.

RETHOME_AMP1MAX **← Motor cutoff**
RETHOME_AMP2MAX **← Motor cutoff**
Possible values: 0..200A
If the actual current consumption exceeds this limit, RETHOME decision is taken and both motors are cut off.
0 is disabled.

RETHOME_WATT1MAX **← Motor cutoff**
RETHOME_ WATT2MAX **← Motor cutoff**
Possible values: 0..5000W
If the actual power consumption exceeds this limit, RETHOME decision is taken and both motors are cut off.
0 is disabled.

RETHOME_RPM1MIN **← Optional motor cutoff**
RETHOME_RPM1MAX **← Optional motor cutoff**
Possible values: 0..60000
Default: 0 (disabled)
If the RPM1 sensor connected to servo capture SCAP10 is configured and its value falls strictly outside MIN…MAX range, RETHOME decision is taken. Besides, both motors could be cut off, or one of them can be enabled depending in FB_TH_MODE and FB_THRB_MODE.
0 is disabled.

RETHOME_RPM1MIN_THRMIN
RETHOME_RPM1MIN_THRBMIN
Possible values: 0..24000
Default: 0 (always check)
The RPM1MIN condition is evaluated only if the value of THR **or** THRB is larger than this value and the motor is enabled (not gliding). In essence, the condition RPM1MIN will only

work if the request for thrust has been applied, but the engine is not spinning. This value filters out false positives.

RETHOME_ZONE_MIN
RETHOME_ZONE_MAX
Possible values: -1..127
Defines ZONE index range. When actual position is out of all the zones listed, RETHOME decision is taken.
-1 is disabled

Time to return home

RETHOME logic uses 2 return home times:
1. Using static RETHOME_CRUISESPEED , RETHOME_CRUISESPEED , RETHOME_WINDHEADING
2. Using dynamic cruise speed, wind speed, wind heading
RETHOME_MAXTIME_RESERVE2BASE is a bonus flight time left after returning home.

The most pessimistic return home time is taken into account during evaluation of:
RETHOME_MAXTIME2BASE
RETHOME_AHBATCAP1
RETHOME_AHBATCAP2
RETHOME_WHBATCAP1
RETHOME_WHBATCAP2

Example:
Assuming the return home will be with the wind the wind speed will imply higher return groundspeed and minimize time to return home.
Assume the UAV has cruise speed 60km/h (NAVI_AVG_AIRSPEED_KMH=60km/h) and some 45km/h glide speed, battery capacity 3.1Ah (REAL discharge capacity under load) monitored using AMP1 input.
3 approaches are possible:
1. Assume the wind will not change direction in flight or lessen if blows towards home, maximize battery usage (OPTIMAL CAPACITY USAGE, universal)
RETHOME_CRUISESPEED=0
RETHOME_WINDSPEED=0 //ignored
RETHOME_WINDHEADING=0 //ignored
RETHOME_AHBATCAP1=3.1
Will return home when actual estimate would indicate the battery would be empty overhead, if the wind blows favorably to return home, return home time will be assumed shorter and maximal possible fly-out distance will be longer.
'Wind can help or can act against'

2. Assume the wind could change, no known a priori weather (SAFEST, universal)
RETHOME_CRUISESPEED=45
RETHOME_WINDSPEED=0
RETHOME_WINDHEADING=0 //ignored
RETHOME_AHBATCAP1=3.1
Will return home when actual estimate would indicate the battery would be empty overhead. Even if the wind will blow favorably to return home, it would be neglected and it will be assumed the plane cannot travel faster than CRUISESPEED relative to ground. If, however, the real time wind will be more unfavorable, it will be taken into account and the plane will return earlier. Even if the wind stops helping during return home, the plane will return. *'Wind cannot help, but can act against'*

3. Assume the wind could change, known a priori weather (BALANCED SAFETY)
RETHOME_CRUISESPEED=45
RETHOME_WINDSPEED=30 (meteo prediction, km/h)
RETHOME_WINDHEADING=145 (direction at which the wind blows)
RETHOME_AHBATCAP1=3.1
Similar to case 2, but this time a priori estimate of the wind is more optimistic. The wind direction and speed is prescribed and if it is favorable will be assumed the return home will be faster and shorter, unless real time wind estimate is more pessimistic. However, even if the wind somehow will become more favorable for return, it will be neglected.
'Wind can help within prescribed limits, can act against'

In other words, difference between cases is:

1. **Says the wind can help or act against and always uses this**
2. **Says the wind cannot help but can act against**
3. **Says the wind can act against or can help, but won't help more than prescribed wind pattern**

Example:

If you are planning a maximum range flight in one direction then back, perform the following calculations:

1. Set RETHOME_MAXTIME [s] as maximum flight time when loitering overhead minus some comfortable margin
2. Set RETHOME_RESERVE2BASE [s] to your preferred time you need to land with comfort, this margin is independent from condition 1.
3. Assume worst case wind speed from weather predictions and find the max time it should take for UAV to return home flying against it (use or adjust a guess for RETHOME_CRUISESPEED [km/h]). Use this time it for RETHOME_MAXTIME2BASE [s].
4. Enter pessimistic (strong) estimates for RETHOME_WINDSPEED [km/h] and RETHOME_WINDHEADING [deg, navigator direction=blowing at] as wind direction
5. Enter RETHOME_MAXDIST [km] and RETHOME_MAXLOSDIST [km] slightly larger than distance from takeoff to farthest point (usually with 0.5km margin)
6. Enter RETHOME_MAXTRIP [km] as about 90% of the longest mission travel over ground experienced with the plane. This is doubling the protections.
7. Make sure the failsafe setting of the RC receiver is set to MISSION (MODE_CHANNEL low or unused) instead of usual MAGNET HOME.

The reason of triggering return-home is logged in TRACE.
Event ID are as follows:

8 RETHOME: MINAGL DISABLING	MIN AGL logic disabled, usually during approach list
9 RETHOME: MINAGL ARMING	MAX AGL logic enabled, after reaching minimal altitude
10 RETHOME: MINAGL	altitude out of range
11 RETHOME: MAXAGL	altitude out of range
12 RETHOME: MAXDIST	horizontal distance too large
13 RETHOME: MAXLOSDIST	straight-line distance too large
14 RETHOME: TIME	flight time too long
15 RETHOME: RX MIN	RETHOME channel out of range
16 RETHOME: RX MAX	RETHOME channel out of range
17 RETHOME: MINLOSANGLE	too low angle over horizon (obstructing trees?)
18 RETHOME: EXCLZONE	flight takes place in a no-flight zone (fixed limit)
19 RETHOME: ALLOWZONE	flight takes place out o allowed zone (fixed limit)
20 RETHOME: GPSCONN	GPS communication problem for more than 4s
21 RETHOME: GPSLOCK	GPS lock problem during light for more than 10s
22 RETHOME: MAXTIME2BASE	Time needed to return to base too long
23 RETHOME: TIMERESERVE2BASE	Estimated total flight time reach consume time reserve
24 RETHOME: MAXTRIP	Total travel length too long
25 RETHOME: TARGET_MAXDIST	Target waypoint too far
26 RETHOME: VOLTMIN1	Voltage 1 below minimum
27 RETHOME: VOLTMIN2	Voltage 2 below minimum
28 RETHOME: HEARTBEAT_TIMEOUT	@@@PING not received
29 RETHOME: ZONE	Flying outside of regions defined by ZONE
52 RETHOME: BYCOMMAND	@@@RETHOME command received
57 RETHOME: AHCONSUMED1	Ah consumed exceeded RETHOME_AHBATCAP1
58 RETHOME: AHCONSUMED2	Ah consumed exceeded RETHOME_AHBATCAP2
59 RETHOME: RPM1MIN	RPM1 below min value
60 RETHOME: RPM1MAX	RPM1 above max value

Parachute management

It is possible to assign parachute servo output to any output channel except throttle.
The parachute can be opened in automatic mode following autopilot's logic, RC input channel, meeting specific conditions like altitude, or activating at waypoint.
In order to test the parachute servo connection, a Wiggler feature is moving servo back in forth during boot-up a few times (around 2Hz) by very small amount and this movement is audible.
Opening the parachute implies immediate motor cutoff. It also aborts mission execution, and the plane returns home in glide mode if the parachute didn't opened.
Parachute opening is immediate in emergency situations or due to RC transmitter switch, but can be delayed by a few seconds when deployed at waypoint. This allows slowing down the speed and allows propeller to stop.
Note: once armed, the parachute condition will remain active even when executing approach list. If you want to disable parachute protection, you must use specific waypoint action flags.

@@@PARA
Immediate parachute deploy.

@@@PARALAND
Start loitering, cut off motor, descend to safe altitude then deploy the parachute below PARA_NODEPLOY_ABOVE, or immediate opening if already below that altitude.

@@@PARARESET
Reset parachute logic, close latch, will stay armed.

@@@PARAZONESHOW
Show allowed zones guarded by parachute.

PARA_OUTCHANNEL
Possible values: -1...5 (default -1)
Select parachute output channel.

PARA_OUTPUT_INACTIVE_SERVO
(default 10000)
PARA_OUTPUT_ACTIVE_SERVO
(default 20000)
Possible values: 0…24000
Parachute latch servo position. 0 or 1 are possible, indicating TTL levels high/low.

PARA_DEPLOY_CHANNEL
Select parachute input channel. When its value is between PARA_DEPLOY_CHANNEL_MIN and PARA_DEPLOY_CHANNEL_MAX, the parachute will be deployed. When the channel will move to inactive state or will be disconnected, the parachute latch will retract (open) what allows manually closing the parachute door.

PARA_DEPLOY_CHANNEL_MIN
(default 18000)
PARA_DEPLOY_CHANNEL_MAX
(default 24000)
Possible values: 6000... 24000
Outside this input range at PARA_DEPLOY_CHANNEL, the parachute will open immediately.

PARA_ARMABOVE
Possible values: -1e6...1e6 (default 80m)
Altitude above takeoff point when the parachute logic is armed. Automatic parachute deployment is activated.
1e6 is disabled.

PARA_NODEPLOY_ABOVE
Possible values: -1e6...1e6 (default 100m)
In automatic mode, if the parachute is scheduled for opening, it will not open until the plane descends below this altitude. Emergency opening by RC or console command are not subject to this protection.
-1e6 is disabled.

PARA_DEPLOY_BELOW
Possible values: -1e6...1e6 (default 60m)
The parachute will be scheduled for deployment if is already armed.
-1e6 is disabled.

PARA_NODEPLOY_BELOW
Possible values: -1e6...1e6 (default 30m)
In automatic mode, if the parachute is scheduled for opening, it will not open until the plane is above this altitude. Emergency opening by RC or console command are not subject to this protection.
-1e6 is disabled.

PARA_DEPLOYMENT_DELAY
Possible values: 0...5s (default 1s)
The delay between motor cutoff and parachute opening. Not used during emergency opening (low altitude) or manual (RC channel) deployment which are always immediate.
0 is no delay.

PARA_WIGGLERTIME
*Possible values: 0...256 (default 4*32=4s)*
The time during which the parachute servo is moving after booting the autopilot.
PARA_WIGGLERMOVE
Possible values: 0...5000 (default 200)
Parachute servo travel for wiggler. The movements are around PARA_OUTPUT_INACTIVE_SERVO position.

PARA_MAXDIST
Possible values: 0...1e6km (default 1e6)
Hard protection against runaway – immediate parachute will occur when the plane is outside this radius from takeoff, regardless the altitude. 1e6 is disabled.

PARA_ZONE_MIN
PARA_ZONE_MAX
Possible values: -1..127
Defines ZONE index range. When actual position is out of all the zones listed, immediate parachute opening occurs.

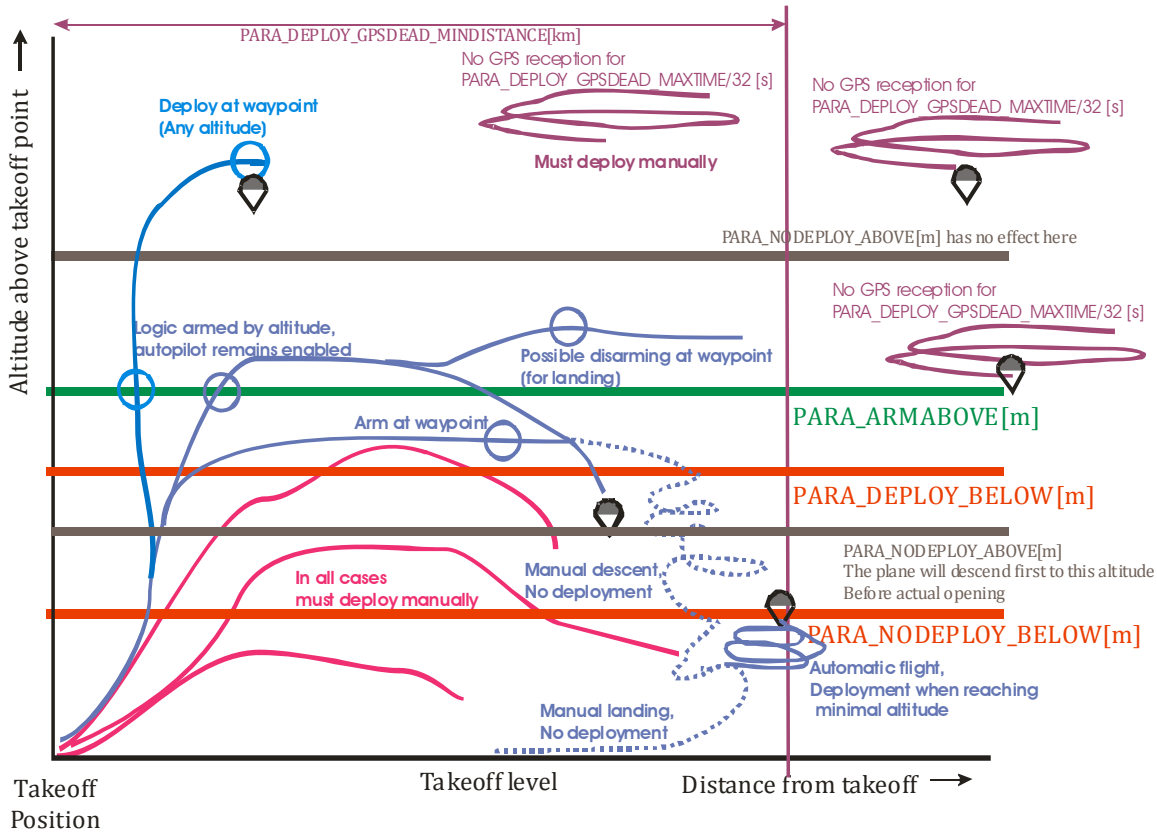
PARA_DEPLOY_GPSDEAD_MAXTIME
*Possible values: 0... 19200 (max 32*600s, default 32*30s=960)*
Hard protection against GPS signal loss – will cause immediate parachute opening since there is no guarantee the plane can descend in place. 0 disables.

PARA_DEPLOY_GPSDEAD_MINDISTANCE
Possible values: 0..1e6km (default 0.2km)
The parachute should not deploy overhead due to persistent GPS signal loss, as it is in visual range at that time.

Parachute entries in trace:

ID	Action
34	MOTOR CUTOFF FOR PARACHUTE
35	PARACHUTE ARMING AT PARA_ARMABOVE
36	PARACHUTE DEPLOYED BELOW PARA_DEPLOY_BELOW
37	PARACHUTE ARMED AT WAYPOINT
38	PARACHUTE DEPLOYED AT WAYPOINT
39	PARACHUTE DEPLOYED AFTER DESCENT (was earlier scheduled for opening by PARA_DEPLOY_BELOW)
40	PARACHUTE DEPLOYED BY RC (using PARA_DEPLOY_CHANNEL)
41	PARACHUTE_DEPLOYED_BY_MESSAGE (using @@@PARA)
47	PARACHUTE RESET BY RC (using PARA_DEPLOY_CHANNEL)
48	PARACHUTE DISARMED AT WAYPOINT (using special trigger flags)
49	PARACHUTE_DEPLOYED_FAILED_GPS (after PARA_DEPLOY_GPSDEAD_TIME)
50	PARACHUTE_DEPLOYED_MAXDIST (too far)
51	PARACHUTE_DEPLOYED_ZONE (outside of all parachute zones)
56	PARACHUTE_LANDING_BY_MESSAGE (beginning descent due to @@@PARALAND)

Interaction of parachute logic and flight altitudes



Navigation zones

RETHOME and PARA logic can use rectangular zones for defining complex shapes of regions that are safe to fly. If the plane is in automatic mode and outside of a list of zones, an action can be triggered. ZONE storage is independent from the actual logic: Both RETHOME and PARA can use none, all or overlapping sub-regions of the zone storage.
A zone is defined by a pair of coordinates (lat1,lon1,lat2,lon2) where lat2>=lat1, lon2>=lon1. Invalid zones are rejected during writing. The zones can geographically overlap.

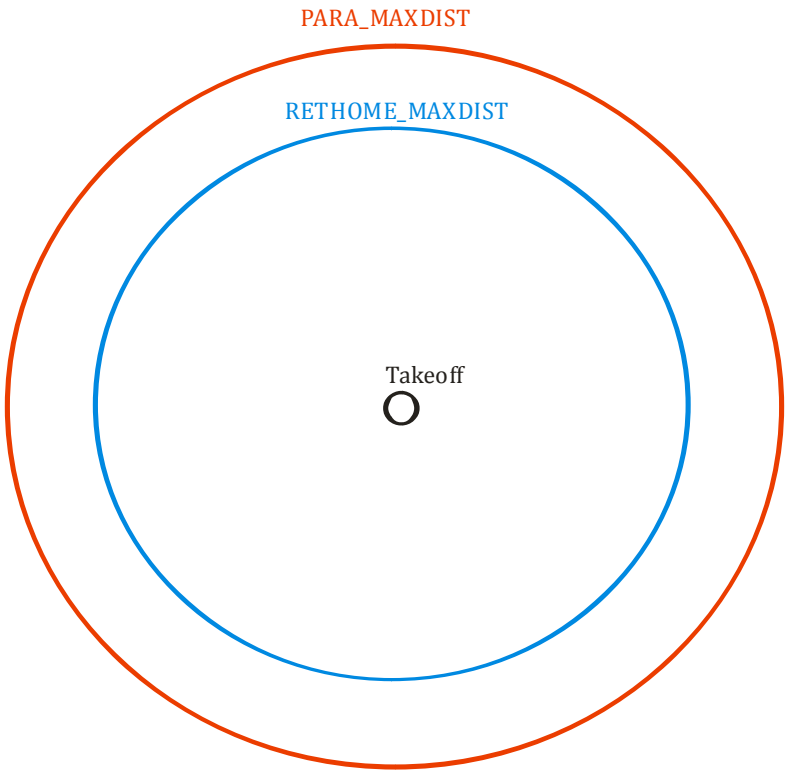
@@@ ZONESHOW N
Displays zone coordinates

@@@RETHOMEZONESHOW displays zones used for triggering RETHOME

@@@PARAZONESHOW displays zones used for deploying a parachute

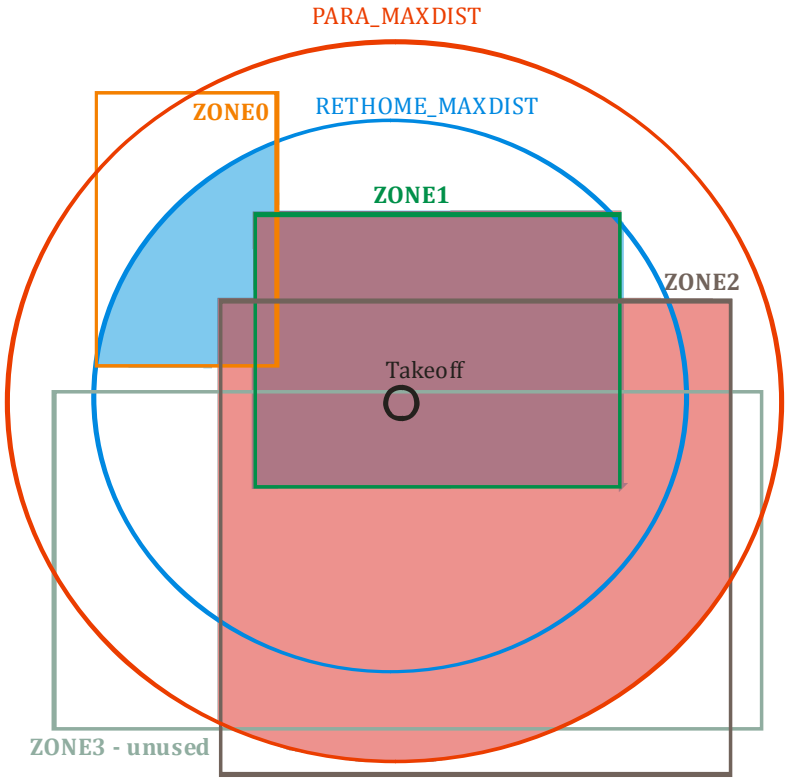
@@@ZONEWRITE N,lat1,lon1,lat2,lon2
or
@@@ZONEWRITE N lat1 lon1 lat2 lon2
N=0-127
lat1, lat2
Possible values: -90…90
lon1, lon2
Possible values: -180…180
Defines a rectangular zone

Interaction or parachute and RETHOME safety zones/distances



PARA_MAXDIST=2.5
RETHOME_MAXDIST=2
PARA_ZONE_MIN=-1
PARA_ZONE_MAX=-1
RETHOME_ZONE_MIN=-1
RETHOME_ZONE_MAX=-1

Will parachute outside of RED cicrcle
Will return home outside of BLUE cicle
If parachute not deployed will return home



PARA_MAXDIST=2.5
RETHOME_MAXDIST=2
PARA_ZONE_MIN=1
PARA_ZONE_MAX=2
RETHOME_ZONE_MIN=0
RETHOME_ZONE_MAX=1

Will parachute outside of RED zone
Will return home outside of BLUE zone
If parachute not deployed will return home

Advanced settings

WAYPOINT_SIZE
Possible values: 0 … 4096 (default 4096 - unlimited)
Limit the number of accessible waypoints. Reserved for future use; designed for facilitating portability of the flight plans. Typical available number of waypoints is 1200-1300.

Advanced autopilot override management using RC receiver (RX)

As a rule, it is possible to take total control of all the servos and motor by a single switch on the RC transmitter. This is called manual RC override and uses a separate, privileged channel on the RC transmitter. The minimum condition to fly with the FLEXIPILOT is to connect rudder, elevator, throttle and RC override input channels.
In order to prevent hijacking or to prevent problems when flying in high interference environment, you can define zones where the autopilot will ignore the RC receiver.
Another benefit is the possibility to disable the RC receiver during simulations, as typically you may setup the RC transmitter/receiver with failsafe feature to the values forcing the plane to return home – in this case turning the transmitter off would make the virtual simulation simply return home.
Another exotic benefit is to ignore the RC receiver in high altitude flights, when the temperature will far exceed commercial temperature range of the RC transmitters.
In order to perform the simulation with RC TX off, you can use **@@@RXOVROFF** command.

RC override capability

It is possible to ignore or the RC receiver under specific conditions. This section also defines assisted mode flying, used for safe obstacle avoidance when flying in automatic mode.

RXOVR_MODE
Possible values: 0 … 2 (default 0)
0 - Manual RC override always possible (passing to manual mode on demand)
1 - Manual RC override possible within RXOVR range zone, self-switching
2 - Manual RC override not possible (100% autonomous)
Do not set this value to anything other than 0 unless you have a good reason!
This option is increasing safety of autonomous flight at the edge of RC reception, typically far beyond line of sight but might make the UAV fully autonomous when you are not ready for it.
The following commands change this value in real time
(without saving them to nonvolatile memory):
@@@RXOVRON *=0*
@@@RXOVRAUTO *=1*
@@@RXOVROFF *=2*

The distance of RXOVR zone is calculated relative to takeoff position.

RXOVR_MAXAGL
Possible values: -1e6…1e6 (default 1000m)
Altitude above takeoff point to disable RC override.
Active if RXOVR_MODE=1.

RXOVR_MAXDIST
Possible values: 0…1e6 (default 1km)
Horizontal distance to disable RC override.
Active if RXOVR_MODE=1.

RXOVR_MAXLOSDIST
Possible value: 0…1e6 (default 1km)
Line-of-sight distance to disable RC override.
Active if RXOVR_MODE=1.

RXOVR_ASSIST_MAXLOSDIST
Possible value: 0…1e6 (default 1km)
Line-of-sight distance to disable RC assisted mode.
If RXOVR disables reading RC controller, naturally assisted mode will not work anyway.
0 is disabling this check.

Note: you can also disable the RC override at specific waypoint, until the next waypoint is met. This requires setting appropriate bit in WAYPOINT/TRIGGER bitfield. When the next waypoint is met and the plane is out of RXOVR range, the manual mode can remain disabled.

Multiple UAV support with a single RC transmitter

RXOVR_SELECTOR_CHANNEL
Possible values:-1..11
Default: -1 (disabled)
Defines which RC input channel will be used for selecting a UAV. If the channel is not between MIN and MAX, all RC transmitter orders will be ignored.
Use with caution!
Understand that the consequences of toggling between UAVs can be inherently difficult to manage: consider when parachute deployment channel is selected, cycling through UAVs will make all of them deploy the parachutes! Similar condition affects most UAV logic.
However, if an UAV is unselected, it enters automatic flight mode automatically: it is equivalent to ignoring the RC receiver, but is not equivalent to switching off receiver; the RC receiver failsafe values are not being generated in it as the receiver is receiving a valid signal. Therefore unselecting the UAV is not always equivalent to switching off the receiver.

RXOVR_SELECTOR_CHANNEL_MIN
Default: 6000
RXOVR_SELECTOR_CHANNEL_MAX
Default: 24000
By default the zone is as wide as possible, always selected.

Assisted mode flying (stabilized mode, flying by turnrates and altitude target)

RXOVR_ASSIST
Possible values:0…2
Default: 1 (only large deflections)
Selects what style of assisted mode flying is possible.
1 – only large stick deflections will affect autopilot navigation
2 – it is possible to keep the autopilot in direct flying mode when ENCHANNEL is enabled
3 – both modes

	Bit1 {2dec}	Bit0 {1dec}
RXOVR_ASSIST	When channel enabled	Only strong deflections

RXOVR_ASSIST_ENCHANNEL
Possible values:-1..11
Default: -1 (disabled)
When this channel is active, the UAV will ignore completely the waypoints and will fly all the time in assisted mode. This is different from normal operation when the UAV enters assisted mode only when stick deflections are high (assuming emergency user's request).
Use with caution!
Make sure that failsafe position in RC transmitter is set to disable or the UAV will not return home when out of RC range. Enabling this channel is equivalent to flying like RV plane in stabilized mode, no exceptions. Rethome conditions are periodically tested, however, and the autopilot can adjust engine controls, or even enable mission abort, but the latter will not come into effect until the navigation is re-enabled by disabling assisted mode flying.

RXOVR_ ASSIST_ENCHANNEL_MIN
Default: 18000
RXOVR_ ASSIST_ENCHANNEL_MAX
Default: 24000
By default the assist will be enabled when the channel is in position HIGH.

RXOVR_ASSIST_DIRCHANNEL
Possible values:-1..11
Default: -1 (disabled)
Recommended: 0 is RUDD
RXOVR_ASSIST_DIRCHANNEL_MIN
Default: 10000
Possible values: -6000..24000
RXOVR_ASSIST_DIRCHANNEL_MID
Default: 15000
Possible values: 0..24000
RXOVR_ASSIST_DIRCHANNEL_MAX
Default: 20000
Possible values: -6000..24000
A range of channel deflections used for reading the value of turnrate. MIN can be >MAX, reversing the action. The commanded turn rate range is defined by separate control law. MID value can be NULL, in such case the MID position is learned during flight from central stick position (recommended).
When MIN or MAX absolute values are less than 6000 (-6000..6000), they signify amount of deflection relative to MID position.
MIN=2000, MAX=-1500, MID=0 means that central position will be learned in manual mode, Maximum deflection will be -1500 units less than that; MIN position will be 2000 units more than learned MID position.

RXOVR_ASSIST_ALTCHANNEL
Possible values:-1..11
Default: -1 (disabled)
Recommended: 1 is ELEV
RXOVR_ASSIST_ALTCHANNEL_MIN
Default: 10000
Possible values: -6000..24000
RXOVR_ASSIST_ALTCHANNEL_MID
Default: 15000
Possible values: 0..24000
RXOVR_ASSIST_ALTCHANNEL_MAX
Default: 20000
Possible values: -6000..24000
A range of channel deflections used for reading the value of target altitude. MIN can be >MAX, reversing the action. The commanded altitude range is defined by separate control law. Note that in some modes (after return home or when altitude track following is disabled, also during loiter) the altitude is not updated by the autopilot, therefore before meeting the next waypoint, last commanded altitude stays. This allows lowering flight in loiter mode over a safe area before parachuting.
MID value can be NULL, in such case the MID position is learned during flight from central stick position (recommended).
When MIN or MAX absolute values are less than 6000 (-6000..6000), they signify amount of deflection relative to MID position.
MIN=2000, MAX=-1500, MID=0 means that central position will be learned in manual mode, Maximum deflection will be -1500 units less than that; MIN position will be 2000 units more than learned MID position.
NOTE: neutral servo position is not recorded for channels SCAP6-11, therefore value of 15000 is assumed. Those channels are not connected to control surfaces, they are most often not spring-loaded and the notion of automatically detected neutral position doesn't apply for them.

RXOVR_ASSIST_ALTCHANNEL_DIRMIX
Default: 0
Possible values: -6000…6000
Tells how much deflection mix from direction override channel into altitude channel.

RXOVR_ASSIST_DIRCHANNEL_ALTMIX
Default: 0
Possible values: -6000…6000
Tells how much deflection mix from altitude override channel into direction channel.

Example setup of manual override for VTAIL

Vtail configuration uses RC channel 1 as left aileron and channel 2 as left vtail, connected to SCAP0 and SCAP1 accordingly. Left aileron uses asymmetric travel.

It turns out that SCAP0 changes asymmetrically when you turn left and right, but the altitude deflection has no effect on it. Possible configuration would be:
@@@RXOVR_ASSIST_DIRCHANNEL=0
@@@RXOVR_ASSIST_DIRCHANNEL_MIN=-3800 //90% of observed max difference between center and full left
@@@RXOVR_ASSIST_DIRCHANNEL_MID=0
@@@RXOVR_ASSIST_DIRCHANNEL_MAX=2000 //90% of observed max difference between center and full right
@@@RXOVR_ASSIST_DIRCHANNEL_ALTMIX=0 //no impact from changing altitude command on SCAP0

SCAP1, the left vtail, is impacted by aileron movement (since there is a mixer from aileron to rudder, which is v-shaped, on the transmitter side): It has been observed that up-down movement adds -2100…2100 units, the full deflection of direction override right from the center add 2432 units.

@@@RXOVR_ASSIST_ALTCHANNEL=1
@@@RXOVR_ASSIST_ALTCHANNEL_MIN=-2100
@@@RXOVR_ASSIST_ALTCHANNEL_MID=0
@@@RXOVR_ASSIST_ALTCHANNEL_MAX=2100
@@@RXOVR_ASSIST_ALTCHANNEL_DIRMIX=-2432

Navigation setup examples

Using default values as a base, try the following:

```
@@@WAYPOINT_HOMEMODE=0 //0=Takeoffhome 1=FixedHome 2=RemoteFixedHome 3=FixedOrTakeoff
@@@WAYPOINT_CYCLES=0
@@@WAYPOINT_AUTOLAND=0
@@@RETHOME_CLIMBTOAGL=50 // The plane will climb to this when rethoming
```

After enabling the autopilot, the plane should start circling overhead at 50m altitude

```
@@@WAYPOINT_HOMEMODE=0 //0=Takeoffhome //1=FixedHome //2=RemoteFixedHome
@@@WAYPOINT_CYCLES=1
@@@WAYPOINT_AUTOLAND=1
@@@RETHOME_CLIMBTOAGL=50 // The plane will climb to this when rethoming
@@@WAYPOINT_BEGIN=15
@@@WAYPOINT_END=15
@@@WPTWRITE15:045,0.20,100,0,4,0
```

After enabling the autopilot, the plane should fly climbing to 100m at the course of 45 deg, then return back maintaining its altitude target to 100m and turn the motor off upon reaching home.

```
@@@WAYPOINT_HOMEMODE=0 //0=Takeoffhome 1=FixedHome 2=RemoteFixedHome 3=FixedOrTakeoff
@@@WAYPOINT_CYCLES=1
@@@WAYPOINT_AUTOLAND=0
@@@RETHOME_CLIMBTOAGL=50 // The plane will climb to this when rethoming
@@@WAYPOINT_BEGIN=3
@@@WAYPOINT_END=3
@@@WPTWRITE3:-0.156,0.123,50,0,2,0
@@@WAYPOINT_HOME_LAT=23.45678
@@@WAYPOINT_HOME_LON=-178.12717
```

Waypoint in mode 2, relative to home by (x,y).
After enabling the autopilot, the plane should fly climbing to 50m and fly to the point 156m to the West and 123m North from fixed home point, then return to fixed home position. Make sure fixed point position is reasonable; otherwise the flight will return to takeoff point after 1km flight according to RETHOME settings.

```
@@@WAYPOINT_HOMEMODE=1 //0=Takeoffhome //1=FixedHome //2=RemoteFixedHome
@@@WAYPOINT_AUTOLAND=0
@@@RETHOME_CLIMBTOAGL=100 // The plane will climb to this when rethoming
@@@WAYPOINT_BEGIN=56
@@@WAYPOINT_END=59
@@@WAYPOINT_CYCLES=2
@@@WAYPOINT_ITERSTYLE=1 // 0=up 1=down 2=up-down 3=down-up
@@@WPTWRITE56:045,0.20,100,0,4,15
@@@WPTWRITE57:135,0.20,100,0,4,15
@@@WPTWRITE58:225,0.20,100,0,4,15
@@@WPTWRITE59:315,0.20,100,0,4,15
@@@WAYPOINT_HOME_LAT=23.45678
@@@WAYPOINT_HOME_LON=-178.12717
```

After enabling the autopilot, the plane should fly climbing to 100m and fly a box centered at the takeoff position, perform the pattern twice, at each waypoint enabling both camera triggers, than return back to fixed defined coordinates and circle overhead.
Note the waypoints are iterated down, in order: 59, 58, 57, 56, 59, 58, 57, 56, fixed home.