

Additional GPS settings da EOSBandi

GPS Filtering

Since GPS positioning is not 100% precise, there is always a small random error in the coordinates that are given by the GPS receiver. This "noise" can make PosHold twitchy. To overcome this twitch and sudden attitude changes caused by GPS inaccuracy, there is a 5 element moving average filter. Which filters the coordinates that are given by the GPS receiver. Use it only if you have a 10Hz update rate.

To enable GPS filtering, set the **#define GPS_FILTERING true**

Low speed D term cancelation

During **position hold**, the control loop's D term can induce errors when copter is moving slowly. To cancel D term when speed is below .5m/s set the **#define GPS_LOW_SPEED_F_FILTER true**.

Waypoint radius

If the distance between the copter and the active waypoint is less than this value, then the waypoint (home point) is considered reached.

You can set the waypoint radius at **#define GPS_WP_RADIUS**, the value is in cm.

The following four defines are currently (r33) is in the MultiWii.ino. They will be moved to config.h in the upcoming releases.

Minimum navigational speed

You can set the minimum speed what is used while navigating. This will be **the target speed** when copter is arrives to the waypoint (home point). The value is in cm/sec **#define NAV_SPEED_MIN 100**

Maximum navigational speed

You can set the maximum speed for navigation. The value is in cm/sec. I **recommend to lower it to 200cm/sec first**, then increase it later. **#define NAV_SPEED_MAX 400**

Slow navigation

When the slow navigation is set, then It allows the copter to reduce its speed to zero when approaching the next waypoint. **#define NAV_SLOW_NAV true**

Maximum allowed banking

This parameter limits the banking output from the navigation routines. By default the maximum banking that can be induced by the navigation routines is 30deg (1deg = 100). For testing, or developing you can decrease it to avoid runaways. **#define NAV_BANK_MAX 3000**

PID controls

As many other flight characteristics, the Position hold and the RTH is controlled by PID control loops.

Position Hold

The location error is calculated in centimeters for X (Longitude) and Y(Latitude) which is fed to a PI controller. The first stage of the controller takes the XY position error and decides how fast the copter should go to reach the correct location.

PI Controller - Poshold

Parameter: **POSHOLD_P** is by default 0.4 or 30 cm/s for a 1m error. The desired rate maxes out at 150cm/s. This doesn't limit the pitch of the copter, The rate controller will do what it takes to maintain this speed as it approaches the location.

Parameter: **POSHOLD_I** is used to overcome wind that may be pushing us away from our target. **The higher the number the faster the copter will compensate**. If your number is too high it will cause oscillations and overshoot.

PID Poshold Rate Control

Now that we have a desired rate, we need to change the copter pitch and roll to give us that rate of travel.

Parameter: **POSHOLD_RATE_P** are the **proportional response**. Your copter's optimal setting will depend on the weight and thrust of your engines. If your copter **overshoots the target, lower this value**.

Parameter: **POSHOLD_RATE_I** is set to 0 by default. Use this value to maintain tight control of the speed of the copter. If the copter is not achieving the speed it needs, this term will make up the difference by tilting the copter more or less.

Parameter: **POSHOLD_RATE_D** is the dampening part. If the value is too high you will see small oscillations in pitch or roll. Once the **POSHOLD_RATE_P** value is dialed in, start at 0, and increment slowly. You should see oscillations die down.

RTH and Navigation

The navigation (RTH) has only one PID control, the desired speed is directly calculated from the distance to the target and the defined speed constrains. The **NAV_P, NAV_I and NAV_D** parameters have the **same functions as the POSHOLD_RATE**.