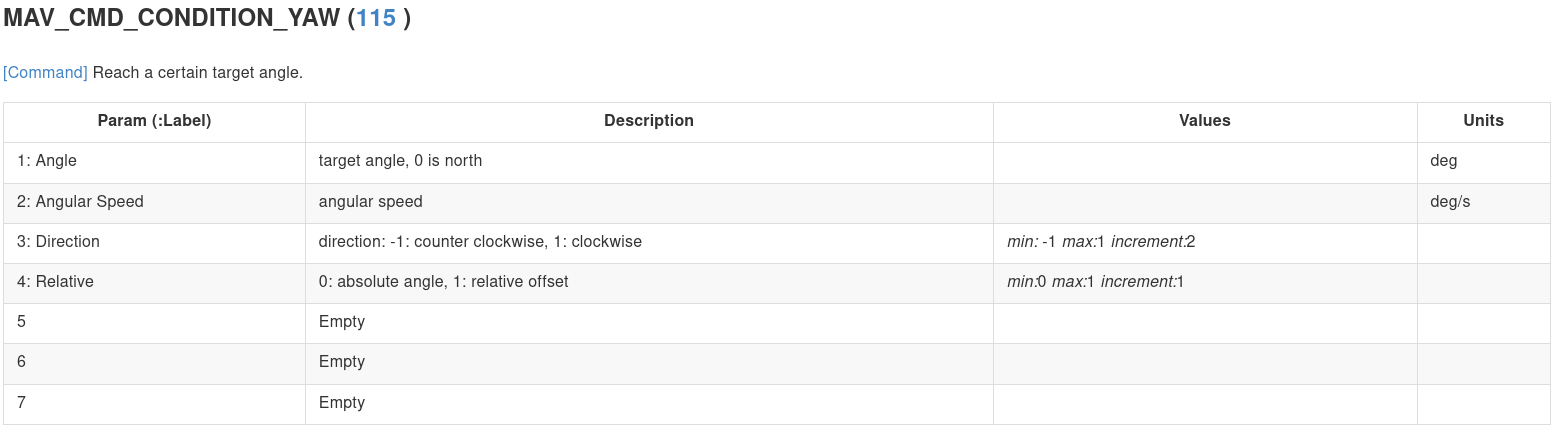
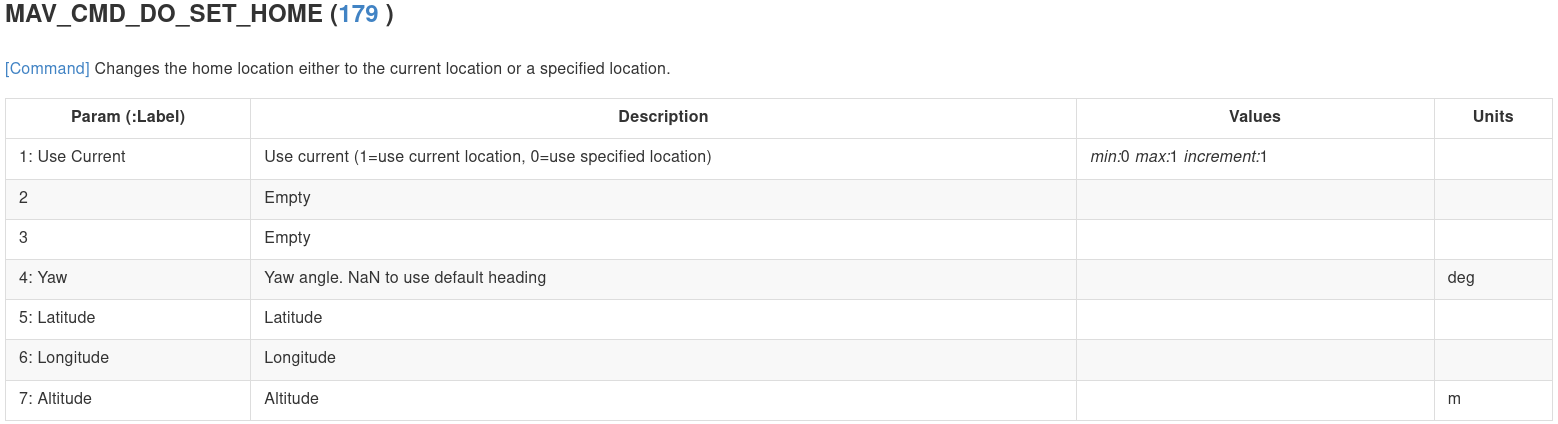
**Yaw and Speed:**

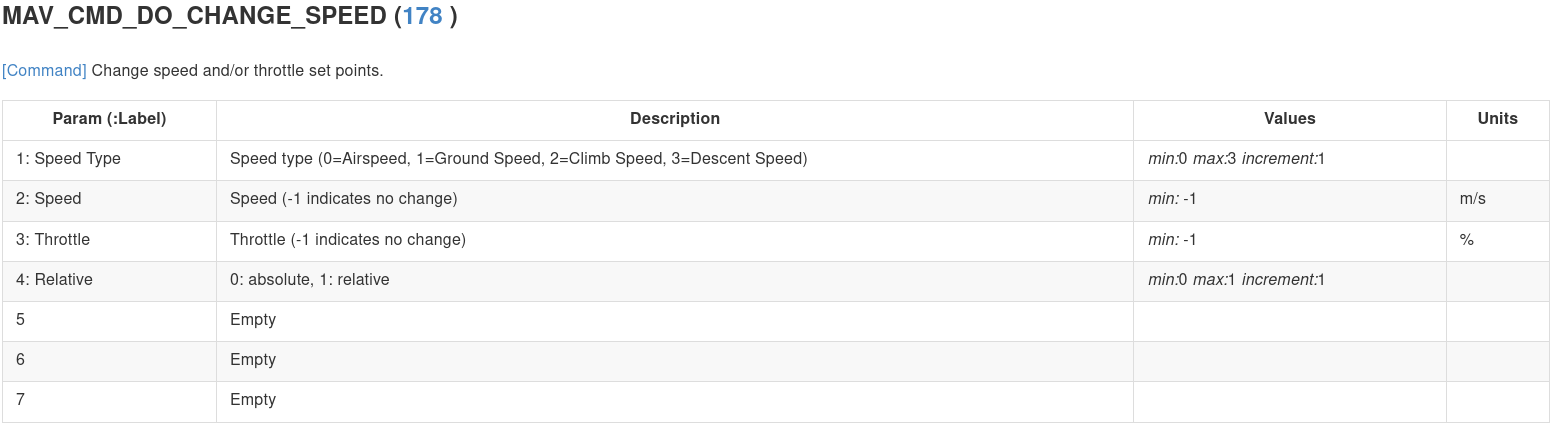
<https://mavlink.io/en/messages/common.html#MAV_CMD_CONDITION_YAW>



<https://mavlink.io/en/messages/common.html#MAV_CMD_DO_SET_HOME>

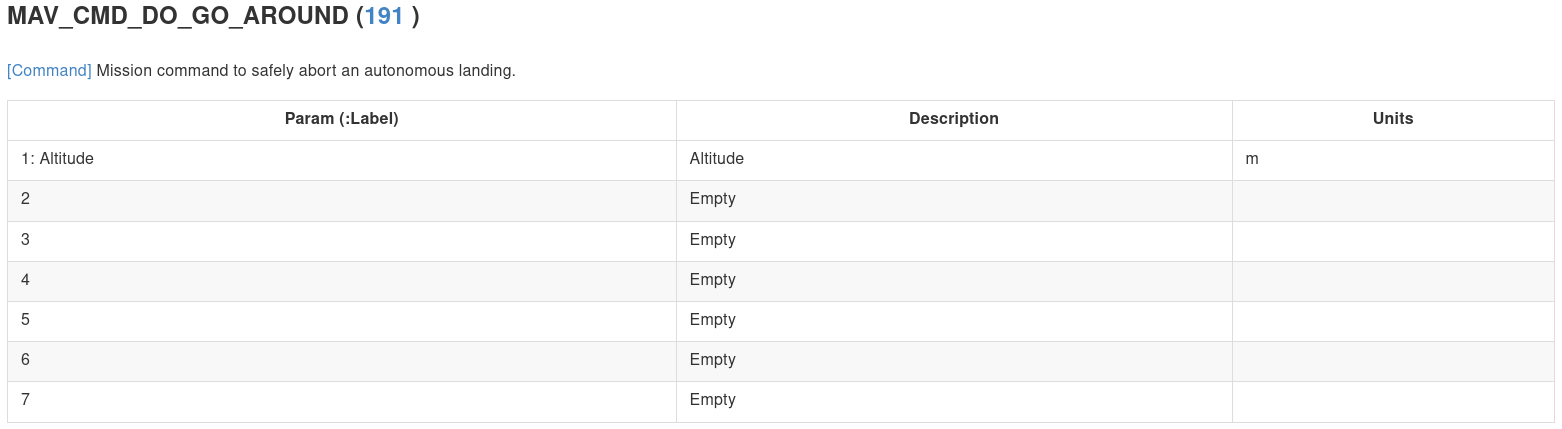


<https://mavlink.io/en/messages/common.html#MAV_CMD_DO_CHANGE_SPEED>



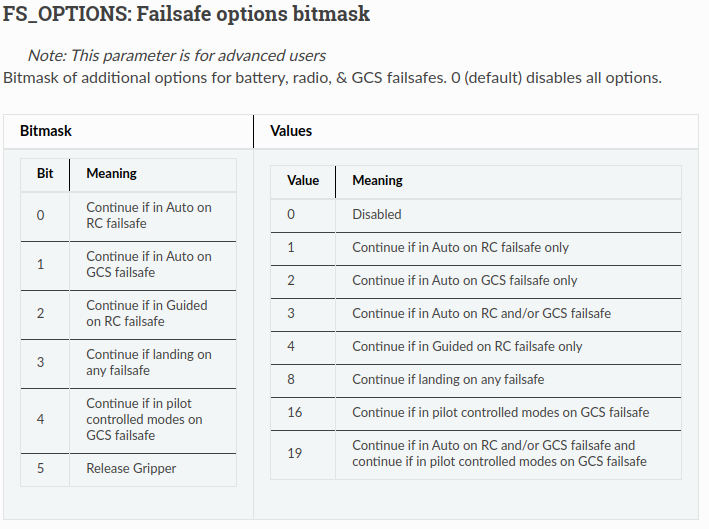
**Control Communication Loss:**

<https://mavlink.io/en/messages/common.html#MAV_CMD_DO_GO_AROUND>



There are messages that can control the failsafes of the vehicle:

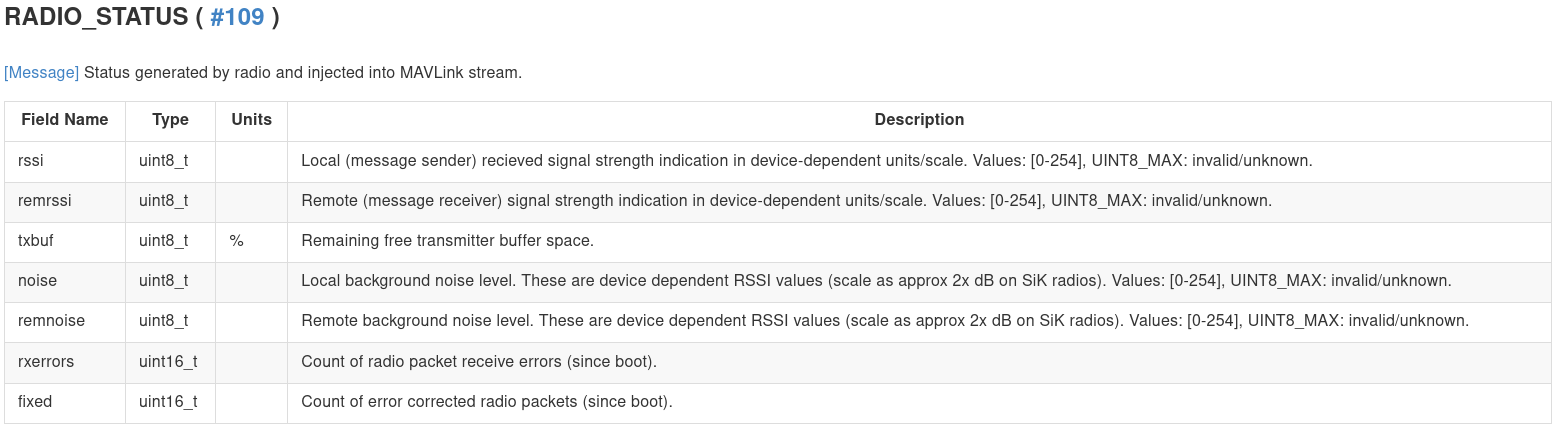
It came from this page:<https://ardupilot.org/copter/docs/radio-failsafe.html>



After disabling the RC failsafe, it may be possible to check the rc state every second and if it is disconnected - custom messages can be sent.

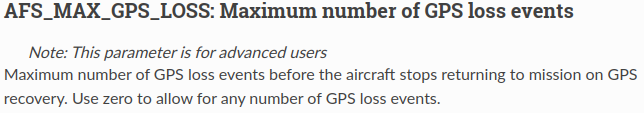
For checking the RC state:

<https://mavlink.io/en/messages/common.html#RADIO_STATUS>

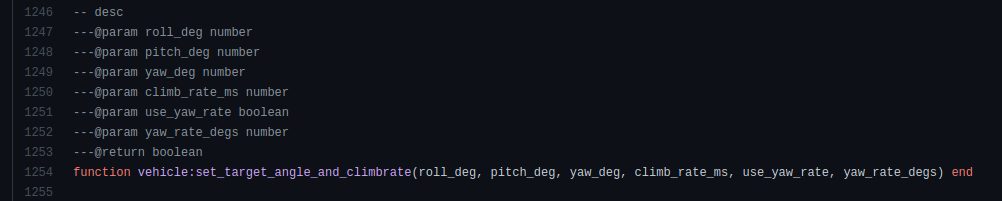


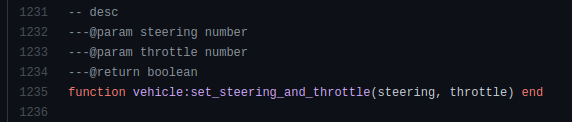
From there, custom mavlink messages as shown in the previous part can be sent.

Parameters to set for the gps fs to not take place:



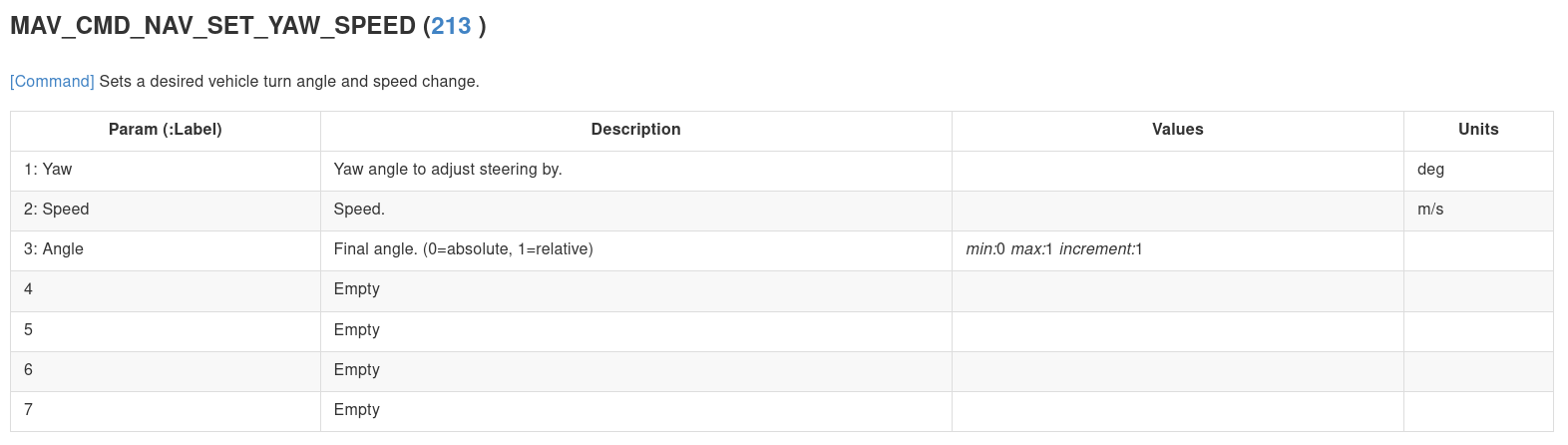
For controlling the vehicle (from the lua documentation -<https://github.com/ArduPilot/ardupilot/blob/master/libraries/AP_Scripting/docs/docs.lua>):





Another option is to create a new MAVLink message:<https://ardupilot.org/dev/docs/code-overview-adding-a-new-mavlink-message.html> ,<https://discuss.ardupilot.org/t/how-to-create-your-own-custom-messages-in-mavlink/26818/2>

This message will probably not work, because it does not exist in the ardupilot documentation, but I did not check that.



# DR With No Gps and No RC

If we want our drone to come back home with no GPS and no RC, we will need to make him turn to the home direction and then pitch forward until GPS will begin working again or the RC will connect and let someone to manual control it.

## Step 1 - Disable auto land

The default of ardupilot is to auto land when ekf failsafe is preset.

For disabling this, we can change the FS\_EKF\_ACTION to be 2 (change mode to alt\_hold

## Step 2 - Find home direction

Without GPS we can’t know for sure what is the direction of the home station (unless we know what is our destination and we are flying straight, then it is configurable before takeoff).

So unless we pre-configured the home direction, we should check our current position and save it as the last\_saved\_position periodically as long as the GPS is active.

When the GPS signal is gone, we can find the correct compass bearing to get home with the last GPS location (latitude and longitude) and the home location (which is saved automatically and can be received easily).

For computing the compass bearing (the yaw angle relative to the north, clockwise) we should:

let lat\_h, lon\_h be: the latitude and longitude of the home location.

let lat\_d, lon\_d be: the latitude and longitude of the drone location.

define delta\_lon = lon\_h - lon\_d

define x = cos(lat\_h) \* sin(delta\_lon)

define y = cos(lat\_d) \* sin(lat\_h) - sin(lat\_d) \* cos(lat\_h) \* cos(delts\_lon)

define compass\_bearing\_radians = atan2(x, y)

Then, when the GPS signal is lost and there is no RC, we can turn the drone to the correct yaw angle (yaw can still be measured using the magnetometer) and then just pitch forward until the GPS signal will return.

## 

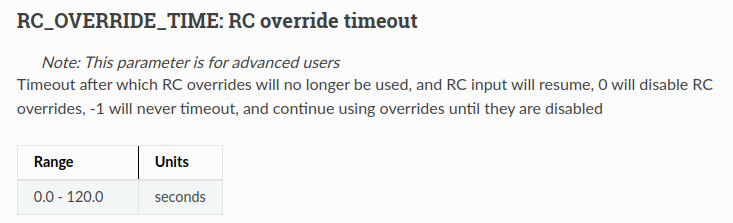
## Step 3 - Make the drone stay in air when the RC disconnect

After a check, when overriding the RC commands with RC\_CHANNELS\_OVERRIDE (#70) even when there is no RC connected, the RC failsafe does not kick in.

So if an RC failsafe is taking place, we can easily override the RC to keep the drone up.

A small but important thing to know is that the RC failsafe changes the mode of the drone to LAND so for stopping the drone from landing after sending it fake RC commands, we should change the mode to ALT\_HOLD.

A usefull paramter in that case is:

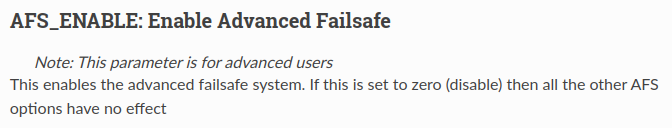


This is good because it makes sure that the aircraft will not fall down even if the mission computer is restarted.

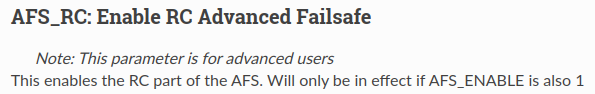
There are several parameters that may help to:

For more documentation about those advanced failsafe parameters: <https://ardupilot.org/plane/docs/advanced-failsafe-configuration.html>

First, for changing any rc advanced failsafe we need to enable advanced failsafes:

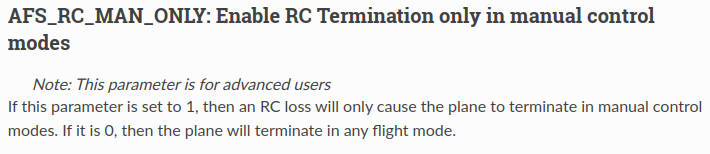


and enable advanced RC failsafes:

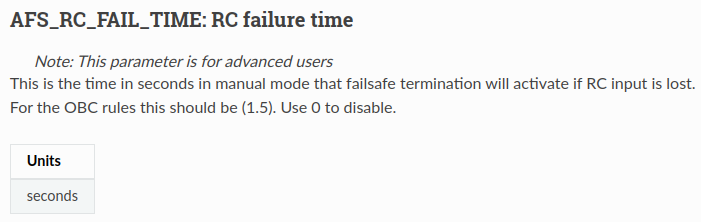


Then, we can enable those:

This parameter will let the drone keep up with its mission with no RC connection when it is not on one of the manual modes.



This parameter will set a timeout until the RC failsafe termination will take place (when on manual modes). We can set this to be very high so it will never happen. This documentation is probably not correct and the units are milliseconds.



RC is not the only concern, GCS failsafe is also a concern so we should probably set this parameter to be 0:

