MultiWii NAV Protocol and Types

This document describes a number of values and enumerations for the stated version of the beta NAV development for **MultiWii**. As iNav implements a part of this specification it is documented in the iNav wiki.

This information is provided in the hope it might be useful NO WARRANTY.

Note that all binary values are little endian (MSP standard).

MultiWii NAV Version

This document should match the 2.3-pre8 / b5 MultiWii / Wingui release, as well as iNav 1.2 and the implementations in mwp and ezgui.

WayPoint / Action Attributes

Each waypoint has a type and takes a number of parameters, as below. These are used in the MSP_WP message. The final column indicated if the message is implemented for iNav 1.2.

Value	Enum	P1	P2	P3	Lat	Lon	Alt	iNav
1	WAYPOINT	speed [1]			✔	✔	✔	✔
2	POSHOLD_UNLIM				✔	✔	✔	✔
3	POSHOLD TIME	Seconds			✔	✔	✔	
4	RTH	Land					✔ [2]	✔
5	SET POI				✔	✔		
6	JUMP	WP#	Repeat (-1 = forever)					
7	SET HEAD [3]	Heading						
8	LAND				✔	✔	✔	

1. Leg speed in an iNav extension
2. Not used by iNav
3. Once SET_HEAD is invoked, it remains active until cleared by a P1 value of -1.

Uploading

For safety, if no mission is defined, a single RTH action should be sent.

Enum	P1	P2	P3	Lat	Lon	Alt	Flag
RTH	0	0	0	0	0	25m [1]	0xa5

1. your choice, really.

In general, flag is **0**, unless it's the last point in a mission, in which case it is set to **0**xa5. When waypoints are uploaded, the values are also returned by the FC, thus enabling the application to verify that the mission has been uploaded correctly.

FC Capabilities (MW only)

Note that 32bit flight controllers (baseflight, cleanflight) use capability == 16 for a different purpose (CAP_CHANNEL_FORWARDING). It is advised to use other messages for checking for capabilities on non-MW platforms.

Capability	Value
BIND	1
DYNBAL	4
FLAP	8
NAV	16

Messages (Nav related)

MNEMONIC	Value	Direction (relative to FC)
MSP_NAV_STATUS	121	Out
MSP_NAV_CONFIG	122	Out
MSP_WP	118	Out
MSP_RADIO	199	Out
MSP_SET_NAV_CONFIG	215	In
MSP_SET_HEAD	211	In
MSP_SET_WP	209	In (& out)

MSP_WP / MSP_SET_WP

Special waypoints are 0 and 255. 0 is the RTH position, 255 is the POSHOLD position (lat, lon, alt).

Name	Type	Usage
wp_no	uchar	way point number
action	uchar	action (wp type / action)
lat	int32	decimal degrees latitude * 10,000,000
lon	int32	decimal degrees longitude * 10,000,000
altitude	int32	altitude (metre) * 100
p1	int16	varies according to action
p2	int16	varies according to action
p3	int16	varies according to action
flag	uchar	0xa5 = last, otherwise set to 0

The values for the various parameters are given in the section “WayPoint / Action Attributes”

MSP_NAV_STATUS

The following data are returned by a MSP_NAV_STATUS message.

Name	Type	Usage
gps_mode	uchar	None PosHold RTH Mission
nav_state	uchar	None RTH Start RTH Enroute PosHold infinite PosHold timed WP Enroute Process next Jump Start Land Land in Progress Landed Settling before land Start descent
action	uchar	(last wp, next wp?)
wp_number	uchar	(last wp, next wp?)
nav_error	uchar	Navigation system is working Next waypoint distance is more than the safety limit, aborting mission GPS reception is compromised - pausing mission, COPTER IS ADRIFT! Error while reading next waypoint from memory, aborting mission. Mission Finished. Waiting for timed position hold. Invalid Jump target detected, aborting mission. Invalid Mission Step Action code detected, aborting mission. Waiting to reach return to home altitude. GPS fix lost, mission aborted - COPTER IS ADRIFT! Copter is disarmed, navigation engine disabled. Landing is in progress, check attitude if possible.
target_bearing	int16	(presumably to the next WP?)

MSP_NAV_CONFIG

The following data are returned from a MSP_NAV_CONFIG message. Values from config.h. Values may also be set by MSP_SET_NAV_CONFIG.

Name	Type	Usage
flags1	uchar	Bitmap of settings from MW config.h b0 : GPS filtering b1 : GPS Lead b2 : Reset Home b3 : Heading control b4 : Tail first b5 : RTH Head b6 : Slow Nav b7 : RTH Alt
flags2	uchar	Bitmap of settings from MW config.h b0 : Disable sticks b1 : Baro takeover
wp_radius	uint16	radius around which waypoint is reached (cm)
nav_max_altitude	uint16	Maximum altitude for NAV (m)
safe_wp_distance	uint16	Maximum permitted first leg of mission (m, assumed?)
nav_max_altitude	uint16	Maximum altitude for NAV (m)
nav_speed_max	uint16	maximum speed for NAV (cm/sec)
nav_speed_min	uint16	minimum speed for NAV (cm/s)
crosstrack_gain	uchar	MW config.h value*100
nav_bank_max	uint16	maximum bank ??? for NAV, MW config.h value*100
rth_altitude	uint16	RTH altitude (m)
land_speed	uchar	Governs the descent speed during landing. 100 ~= 50 cm/sec unknown units
fence	uint16	Distance beyond which forces RTH (m)
max_wp_number	uchar	maximum number of waypoints possible (read only)

MSP_RADIO

If you have a 3DR radio with the MW/MSP specific firmware, the follow data are sent from the radio, unsolicited.

Name	Type	Usage
rxerrors	uint16	Number of RX errors
fixed_errors	uint16	Number of fixed errors, if error correction is set
localrssi	uchar	Local RSSI
remrssi	uchar	Remote RSSI
txbuf	uchar	Size of TX buffer
noise	uchar	Local noise
remnoise	uchar	Remote noise

Implementations

The MSP NAV message set is implemented by mwptools (Linux), ezgui (Android) and WinGUI (MS Windows).