

SoloHeroBus

Feb 13th, 2015

Author: angus@3dr.com

What is SoloHeroBus?

SoloHeroBus is the abstraction layer between the mobile applications and the gimbal controlling the proprietary “HeroBus” interface on the GoPro 3+/4 cameras. SoloHeroBus is defined by a set of custom MAVLink messages listed below in this document.

Supported GoPro Cameras

- GoPro Hero 3+ Silver
- GoPro Hero 3+ Black
- GoPro Hero 4 Silver (In the future)
- GoPro Hero 4 Black (In the future)

Firmware Support

A “3DR Only” (as of writing, unreleased) firmware is required for the Gimbal to talk to the GoPro. By the time Solo launches the Hero 3+ firmware will have been public for some time, however it's unlikely that the Hero 4 firmware will be public.

The firmwares can be found here:

<https://drive.google.com/a/3drobotics.com/folderview?id=0B78TvfhudJqHYUM0NjJnQUJqQkU&usp=sharing>

Protip for flashing the FW:

1. Put the GoPro MicroSD card into a computer using a USB reader (not the GoPro)
2. Copy the “UPDATE” folder onto the root of the MicroSD card.
3. Put the MicroSD card in the camera and boot, it will show that it's updating on the screen a reboot a few times (maybe 3 or 4...)
4. The firmware should now support SoloHeroBus. Helpfully the firmware version displayed on startup isn't incremented from the current public release, so don't expect that to change!

Document Changelog:

- 21/02/2015 Added GOPRO_GET_ALL_RESPONSE message.

MAVLink Messages

The communication structure is standard request/response style. Each set or get request will generate a response message containing either the success of the set or the desired value from a get request.

Example Message Flows

These examples are written in from the perspective of an app controlling the GoPro camera.

Powering the camera on, switching to video and starting video recording

While GOPRO_HEARTBEAT == 2

	<i>MAVLink Message</i>	<i>COMMAND</i>	<i>Value/Result</i>
Send >	GOPRO_SET_REQUEST	POWER	1 (On)
Recv <	GOPRO_SET_RESPONSE	POWER	1 (Success)
Send >	GOPRO_SET_REQUEST	CAPTURE_MODE	0 (Video)
Recv <	GOPRO_SET_RESPONSE	CAPTURE_MODE	1 (Success)
Send >	GOPRO_SET_REQUEST	SHUTTER	1 (Start Video)
Recv <	GOPRO_SET_RESPONSE	SHUTTER	1 (Success)

Powering the camera on, reading camera model and battery level

While GOPRO_HEARTBEAT == 2

	<i>MAVLink Message</i>	<i>COMMAND</i>	<i>Value/Result</i>
Send >	GOPRO_SET_REQUEST	POWER	1 (On)
Recv <	GOPRO_SET_RESPONSE	POWER	1 (Success)
Send >	GOPRO_GET_REQUEST	MODEL	
Recv <	GOPRO_GET_RESPONSE	MODEL	11 (Hero 3+ Black)
Send >	GOPRO_GET_REQUEST	BATTERY	
Recv <	GOPRO_GET_RESPONSE	BATTERY	77 (77%)

GOPRO_HEARTBEAT

Heartbeat messages are **transmitted from the camera** at a periodic rate of 1Hz.

While the status is **0 or 1**, no messages should be **transmitted to the camera**.

While the status is 3, only SHUTTER set requests should be **transmitted to the camera**.

ID	Message Name	Status
215	GOPRO_HEARTBEAT	0: No GoPro detected 1: Incompatible GoPro firmware 2: GoPro Connected 3: GoPro Recording 4: GoPro Error: Over Temperature 5: GoPro Error: Storage missing or full

GOPRO_GET_REQUEST

A get request message is **transmitted to the camera**, which will respond with the requested message with a Get Response Message.

ID	Message Name	Command ID
216	GOPRO_GET_REQUEST	Desired Get Response Message ID

GOPRO_GET_RESPONSE

A get response message is **transmitted from the camera**.

ID	Message Name	Command ID	Value
217	GOPRO_GET_RESPONSE	See GOPRO_COMMANDS table for possible IDs	See GOPRO_COMMANDS table for possible values

GOPRO_GET_ALL_RESPONSE

A get all response message is **transmitted from the camera**.

ID	Message Name	Value	Info
not implemented	GOPRO_GET_ALL_RESPONSE	CAPTURE_MODE	See GOPRO_COMMANDS
		PWRON_MODE	** To Implement **
		EXP_MODE	** To Implement **
		TI_INTVL	** To Implement **
		PWROF_DLY	** To Implement **
		FOV	** To Implement **
		PHOTO_RES	** To Implement **
		AUDIO_SEL	** To Implement **
		BEEP_VOL	** To Implement **
		NUM_LED	** To Implement **
		FLAGS1	Bit 6: LOC_EN 0=OFF 1=ON Bit 5: VID_MODE 0=NTSC 1=PAL Bit 4: OSD_EN 0=DISABLED 1=ENABLED Bit 3: OBM_EN 0=DISABLED 1=ENABLED Bit 2: ORIENT 0=NORMAL 1=UP_DOWN
		BATTERY	0 = 10% 1 = 20% 2 = 40% 3 = 70%

			4 = 100%
		PHOTOS_REMAINING	16 Bit Unsigned Integer
		PHOTOS_TAKEN	16 Bit Unsigned Integer
		VIDEO_REMAINING	16 Bit Unsigned Integer (Minutes)
		VIDEOS_TAKEN	16 Bit Unsigned Integer
		SHUTTER_STATUS	See GOPRO_COMMANDS
		FLAGS2	Bit 6: LOW_LIGHT 0=OFF 1=ON Bit 3: SDCARD_ERR 0=OK 1=ERROR Bit 1-2: PT_EN 0=OFF 1=ON 2=RESET Bit 0: CAMERA_BUSY 0=FALSE 1=TRUE

GOPRO_SET_REQUEST

A set request message will be **transmitted to the camera**. The camera which will attempt to act on the set and then send a set response with the result.

ID	Message Name	Command ID	Value
218	GOPRO_SET_REQUEST	See GOPRO_COMMANDS table for possible IDs	See GOPRO_COMMANDS table for possible values

GOPRO_SET_RESPONSE

A set response is **transmitted from the camera**, which will respond with the set request identified by the Message ID field.

ID	Message Name	Command ID	Result
219	GOPRO_SET_RESPONSE	See GOPRO_COMMANDS table for possible IDs	0: Failure 1: Success

GOPRO_COMMANDS

This table shows all of the commands which cam

ID	Type*	Name	Request/Response Values
0	GS	POWER	0: Off 1: On
1	GS	CAPTURE_MODE	0: Video 1: Photo 2: Burst 3: Time-Lapse
2	S	SHUTTER_STATUS**	0: Stop Video or Time-Lapse 1: Start Video or Time-Lapse or Take Photo 2: Stop Continuous Photo Capture 3: Start Continuous Photo Capture
3	G	BATTERY	0: 0% ... 100: 100%
4	G	MODEL	10: HERO3+ Silver Edition 11: HERO3+ Black Edition
250	G	ALL	See GOPRO_GET_ALL_RESPONSE message.
255	G	REQUEST_FAILED	X: Don't care

* The Type column indicates the following

- G - Get Only - This message will be sent **from** the camera and can not be set.
- S - Set Only - This message will be sent **to** the camera and can not get requested.
- GS - Get Set - This message can be sent **to** the camera, or requested to be sent **from** the camera.

** Once the message SHUTTER is sent with a start value (0x01 or 0x03) and returns successful, the only message (excluding error messages) that can be sent is the equivalent stop message (0x00 or 0x02).

A read message will be **transmitted from the camera** shortly after processing a get request for the desired message ID. The camera will always respond with a get response even if the request failed.