

NanoCom

NanoCom ANT-6F UHF VHF

Manual

UHF/VHF antenna for GomSpace 6U structure

Table of Contents

| | | |
|----------|--|-----------|
| 1 | Changelog | 1 |
| 2 | Introduction | 2 |
| 2.1 | Unpacking and Handling Precautions | 3 |
| 3 | Hardware | 4 |
| 3.1 | Mounting | 4 |
| 3.1.1 | Standard Tightening Torque for Screws | 4 |
| 3.1.2 | Mounting the ANT-6F UHF and VHF on top of NanoStructure 6U | 4 |
| 3.2 | Antenna Deployment System | 5 |
| 3.2.1 | ANT-6F UHF Deployment | 5 |
| 3.2.2 | ANT-6F VHF Deployment | 6 |
| 4 | Software | 7 |
| 4.1 | Getting Started | 7 |
| 4.1.1 | Connecting via the P60 Dock | 7 |
| 4.2 | Interfaces | 8 |
| 4.2.1 | GSRB over I^2C | 8 |
| 4.2.2 | GOSH | 8 |
| 4.2.3 | ANT-6F UHF Client API | 10 |
| 5 | Disclaimer | 13 |

1. Changelog

Table 1.1: Changelog

| Date | Description | Revision | Author | Description |
|------------|-------------|----------|----------|---------------|
| 23/01/2018 | | 3.1.0 | JESM/ANM | First release |

2. Introduction

The ANT-6F comes in two versions:

- **UHF** (340-680 MHz)
- **VHF** (130-210 MHz)

The ANT-6F antenna is designed to fit GomSpace's 6U nano-satellite structure. It is based on the GomSpace Modular 6U Antenna System, which allows for multiple components to be mounted - see [Fig.2.1](#).

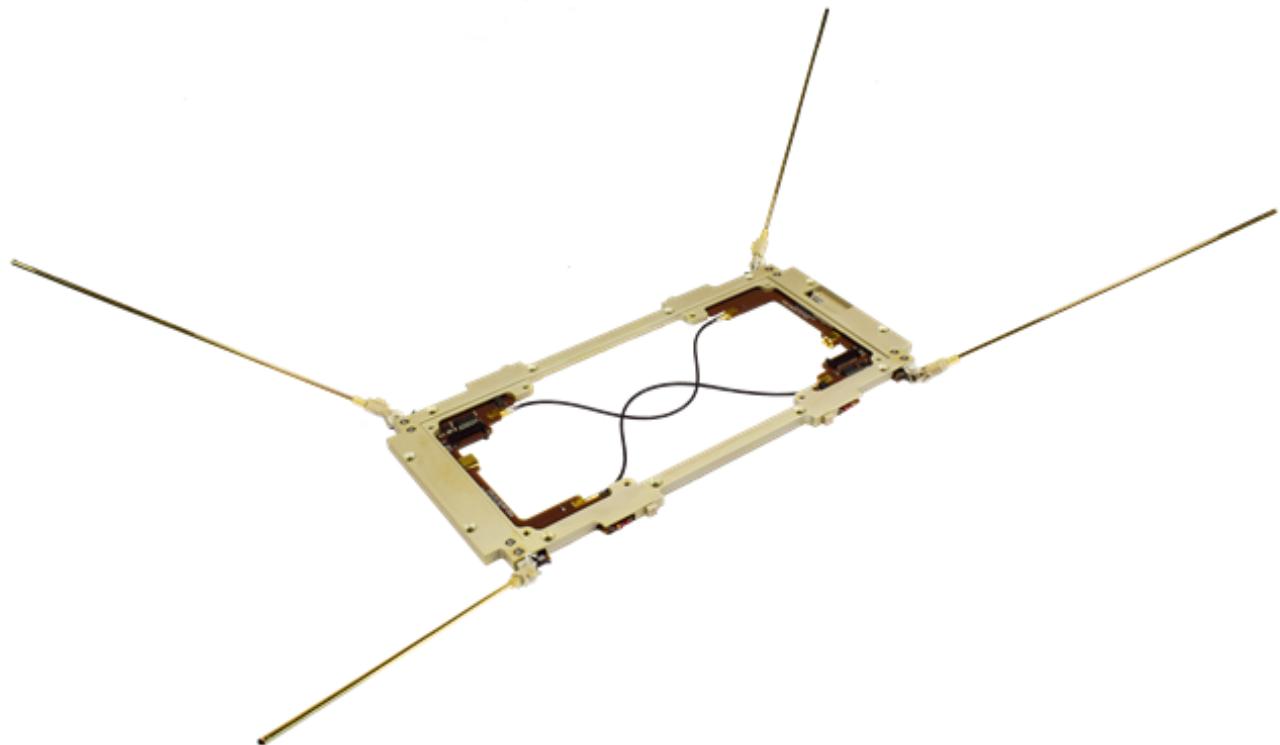


Fig. 2.1: ANT-6F without mounted components

The system supports the following components:

- GomSpace NanoCom S-band patch antenna
- GomSpace NanoPower Modular solar panels 1 cell or 4 cell version
- GomSpace NanoUtil MSP-FPP - Flight Preparation Panel
- Propulsion adaptor for NanoSpace propulsion module

The ANT-6F with solar panels and S-band antenna is shown in [Fig.2.2](#).

The antenna system PCB is designed to be the least obstructive to any top or bottom mounted payload or panels. It has a low profile that allows a solar panel to be mounted on top, and a large aperture in the center suited for a protruding camera lens, propulsion hardware or similar.

The interface of the ANT-6F is GSRB (GomSpace Release Bus) which is a variant of the GSSB (GomSpace Sensor Bus) using the same protocol. The GSRB makes it easy to connect multiple devices on the same bus, as it supplies both power and data.

The ANT-6F consist of two separate boards, each releasing two antennas. One board needs to chain to the other via GSRB and the other one needs to connect to external power/data supply like NanoDock P60. They each hold an individual I^2C address (5 and 6).

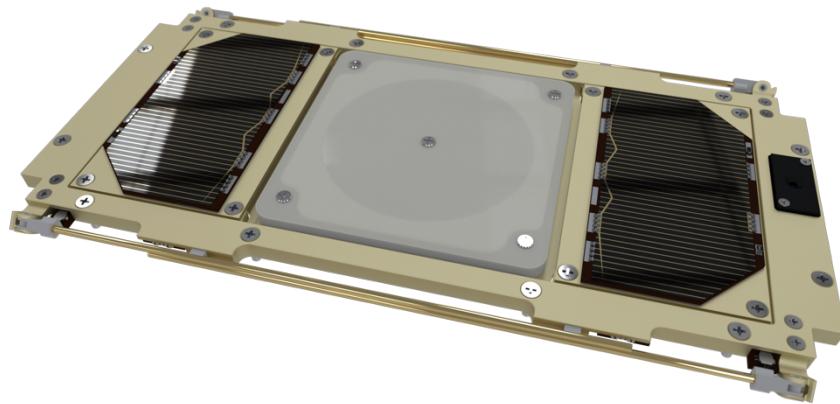


Fig. 2.2: ANT-6F with mounted components (solar panels and S-band antenna)

The two versions of ANT-6F is further described in their datasheets: gs-ds-nanocom-ant6-6f-uhf-<version>.pdf and gs-ds-nanocom-ant6-6f-vhf-<version>.pdf

2.1 Unpacking and Handling Precautions

Warning: The ANT-6F system employs components based on FETs and therefore requires anti-static handling precautions to be observed. Do not touch or handle the product without proper grounding.

Warning: The antenna rods are quite thin and bend easily. Please handle them with care while testing the release system. In case of a broken hinge or antenna, the hole system, which is lock-tied, needs to be reopened and assembled again. This can only be done by GomSpace. It is not an easy replacement and it is a major rework.

3. Hardware

3.1 Mounting

This section describes how to mount the ANT-6F antennea on the 6U nanosatellite structure.

3.1.1 Standard Tightening Torque for Screws

Below is shown a table for GomSpace suggested standard tightening torque for screws.

| Screw Diameter [mm] | Torque [Nm] |
|---------------------|-------------|
| 2.5 | 0.65 |

3.1.2 Mounting the ANT-6F UHF and VHF on top of NanoStructure 6U

To assemble the ANT-6F frame on the 6U structure, use M2.5 X 8 on the short side and M2.5 X10 screws on the long side of the antenna frame. First screw in all the screws loosely. Afterwards tighten the screws one by one in a star pattern across the bracket with a quarter turn at a time until above mentioned torque is reached.

The ANT-6F UHF with the antennas stoved along the sides of the ANT-6F frame is shown in [Fig.3.1](#).

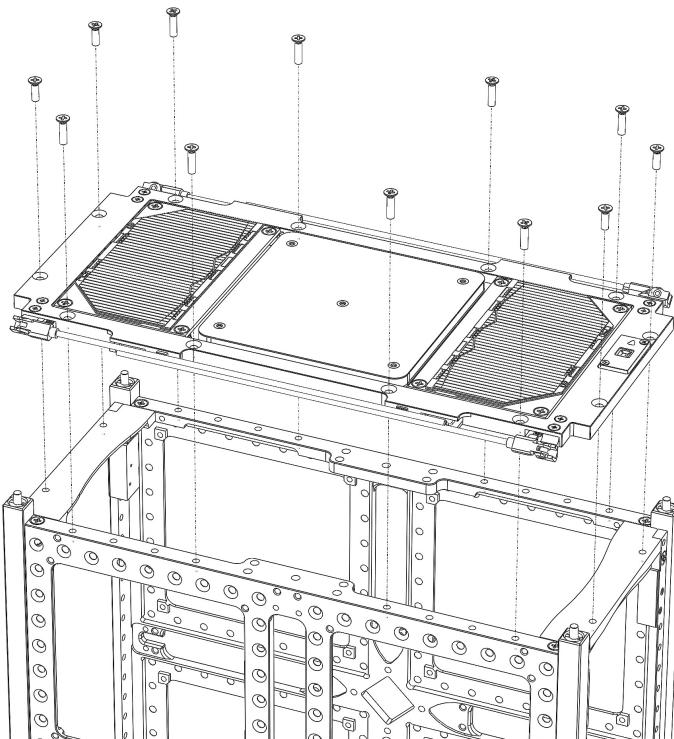


Fig. 3.1: How to mount ANT-6F UHF on NanoStructure 6U

The ANT-6F VHF with the antennas stoved along the sides of the 6U structure is shown in [Fig.3.2](#).

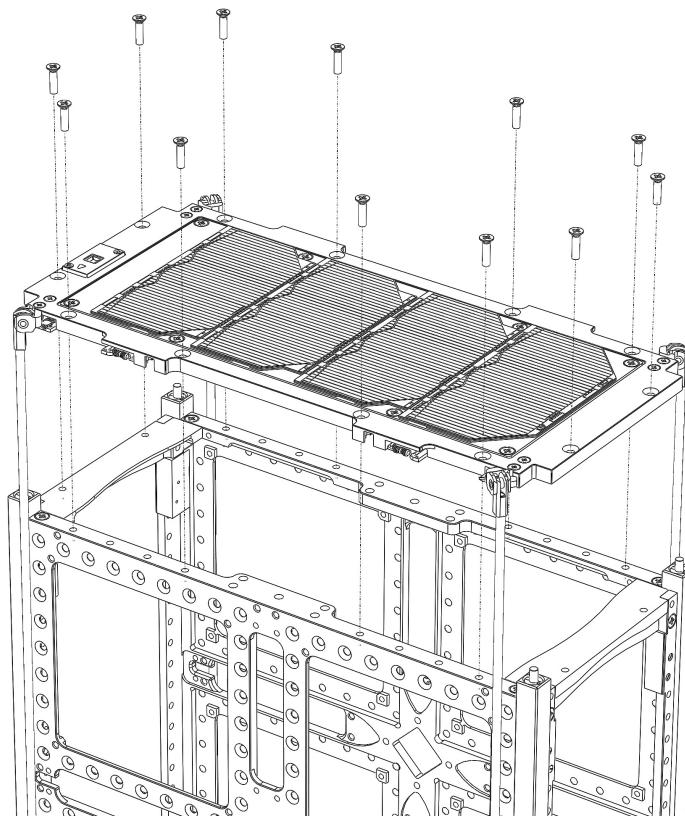


Fig. 3.2: How to mount ANT-6F VHF on NanoStructure 6U

Note: The ANT-6F VHF is used in combination with four NanoUtil AR6s. The mounting of the AR6s is described in the AR6 Manual and ANT-6F-VHF Datasheet: [gs-man-nanoutil-ar6-<version>.pdf](#) and [gs-ds-nanocom-ant6f-vhf-<version>.pdf](#).

3.2 Antenna Deployment System

The ANT-6F is equipped with four monopole antenna rods, with different lengths depending on UHF or VHF version. The antenna release mechanism is implemented differently in the two versions.

3.2.1 ANT-6F UHF Deployment

The four antennas are bound tight with burn wire over the thermal knives on each board. Each board is identified by its CSP address and its two antennas are identified by channel 0 and channel 1, respectively. Channel 1 is the one closest to the MCU - see [Fig.3.3](#).

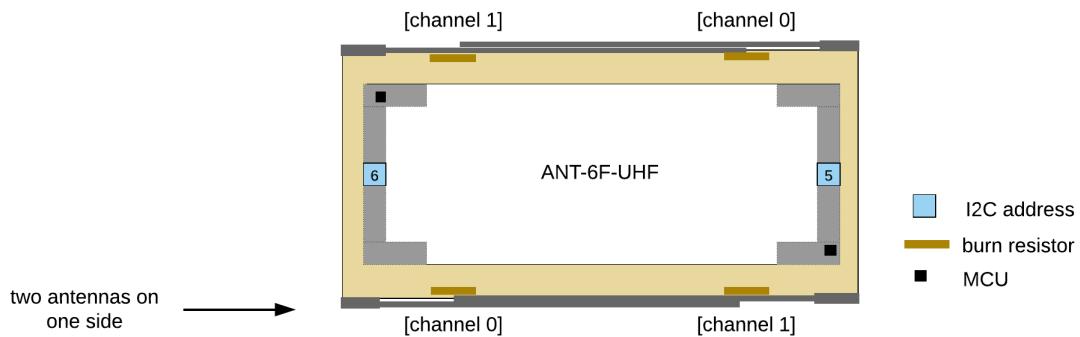


Fig. 3.3: Schematic overview of the ANT-6F UHF release mechanism

To release the two antennas on one side both boards has to activate the thermal knife in the respective side. The two boards has to be individually commanded to do this - see chapter [GOSH](#).

However, there is a backup deployment function which, if activated, will burn on each thermal knife on the respective board for a specific period of time. This is meant as a backup, so the antenna will deploy even if they do not get a command.

Note: This backup has to be disabled after successful deployment.

3.2.2 ANT-6F VHF Deployment

The four antennas on the ANT-6F VHF are held in place, along the sides of the 6U structure, with burn wire over the thermal knives on the NanoUtil AR6 boards. The release of the antennas is controlled by software related to the AR6 board - see the AR6 Manual: gs-man-nanoutil-ar6-<version>.pdf.

4. Software

This chapter describes software related to the ANT-6F UHF. The software related to the ANT-6F VHF is described in the AR6 Manual: gs-man-nanoutil-ar6-<version>.pdf.

The ANT-6F is intended to be used together with the libgssb software library. A part of libgssb comes with the ANT-6F. It implements the GSSB protocol and functions on top of a I^2C layer.

Note: If the ANT-6F is used with the NanoPower P60 Dock it comes with a I^2C layer implementation. Otherwise, an implementation of the I^2C layer is necessary.

In addition, GomSpace has developed a console-like interface called GomSpace Shell (GOSH), which provides a simple but extensive debug and configuration interface. GOSH is a general feature present on several of GomSpace's products.

Note: The ANT-6F itself, do not have GOSH running. To access the ANT-6F GOSH commands, one has to connect the antenna to a P60 Dock.

The console provides a text-interface to a given input/output stream such as a serial port. GOSH is described in detail in the GOSH manual. If this manual has not been provided upon purchase, please contact GomSpace support.

4.1 Getting Started

This section contains a brief description of how to establish a connection with ANT-6F UHF.

4.1.1 Connecting via the P60 Dock

Follow the steps below to command the ANT-6F UHF.

1) Connect the ANT-6F UHF to the P60 Dock

In order to gain access to GOSH, one has to connect the ANT-6F UHF to the P60 Dock - see the ANT-6F UHF datasheet.

2) Connect a serial cable to the P2 connector of the P60 Dock

3) The serial communication program (minicom/tio) should be configured as follows:

- baudrate: 500.000 baud, 8n1
- disable flow-control

4) Run the `gssb ant6` command to list all sub-commands:

```
p60-dock # gssb ant6

burn
stop
status
reset_count
settings
```

```
reboot
temp_internal
```

5) Run the `gssb ant6 status` command to get telemetry data:

```
p60-dock # gssb ant6 status
```

4.2 Interfaces

The ANT-6F supports GSRB over I^2C . The ANT-6F commands can be run directly via GOSH or the ANT-6F Client API can be used to develop custom mission code.

4.2.1 GSRB over I^2C

The ANT-6F-UHF is a GSRB/GSSB compatible device. It uses GSRB (GomSpace Release Bus) over I^2C , which is a variant of GSSB (GomSpace Sensor Bus). The bus carries the supply for ANT-6F UHF in form of GND and 3.3 V and the two I^2C signals SDA and SCL are used for communication.

Note: The ANT-6F UHF has been tested with standard mode (100 kbit/s) and full speed (400 kbit/s)

GomSpace has developed a library called 'libgssb', which makes it easy to communicate with GSSB compatible devices. The library implements commands for operating the ANT-6F UHF.

After having connected ANT-6F to the P60 Dock the commands can be run via GOSH.

4.2.2 GOSH

The GOSH commands are accessible through the GSSB client, which gives access to all ANT-6F UHF sub-commands:

```
p60-dock # gssb ant6

burn
stop
status
reset_count
settings
reboot
temp_internal
```

Below is a description of ANT-6F UHF sub-commands.

Antenna Deployment System (burn)

The antenna deployment system is described in chapter [Antenna Deployment System](#).

Each ANT-6F UHF board has to be commanded to burn on each of its two channels. The GOSH command to do so is:

```
p60-dock # gssb ant6 burn 0 3
```

The `burn` command takes two arguments. The two arguments are which channel to burn on and burn time. Channel 1 is describing the release mechanism closest to the MCU on the board. The burn time is in seconds.

Note: Remember to set the CSP address first by running the command `gssb setaddr <csp address of board>`.

Warning: The max burn time is 20 seconds for lab testing. Setting a wrong burn duration can damage the resistors used on the thermal knives, antenna PCBs or ICs on the PCB.

It is not recommended using the burn command multiple times before launching the satellite in to space.

Stop burning (stop)

Stops burning on both channels immediately.

Seeing the status (status)

It is possible to check the status of one of the ANT-6F UHF boards. There are three status functions, which each gives different telemetry. The telemetry data can be access by running the GOSH command:

```
p60-dock # gssb ant6 status
```

The normal response for this command looks like the following example.

| | |
|---------------------------|-------------------|
| Channel 0 | |
| Burn state: | Idle |
| Burn time left: | 0 |
| Antenna state: | Released |
| Burn attempts: | 0 |
| Channel 1 | |
| Burn state: | Idle |
| Burn time left: | 0 |
| Antenna state: | Released |
| Burn attempts: | 0 |
| Backup | |
| State: | Waiting to deploy |
| Seconds to backup deploy: | 3594 |
| Board | |
| Seconds since boot: | 6 |
| Number of reboots: | 2 |

The telemetry data is described in the table below.

Table 4.1: Telemetry

| Parameter | Description |
|--------------------|---|
| Channel | Channel 0 is the farthest from the MCU |
| Burn state | Can be <i>Idle</i> or <i>Burning</i> |
| Burn time left | Number of seconds until burning stops |
| Antenna state | Can be <i>Released</i> or <i>Not released</i> |
| Burn attempts | Number of deploy attempts |
| Backup | Backup deployment status |
| State | Can be 'Not active', 'Waiting to deploy', 'Deploying', 'Finished successful' or 'Finished unsuccessful' |
| Board | Board status |
| Seconds since boot | Number of seconds since powered on |
| Number of reboots | Number of reboots since last reboot count reset |

Reset (reset_count)

Resets burn counters and reboot counter on ANT-6F UHF.

Configuration (settings)

Warning: The max burn duration set for Ant6F-UHF, before launch, is recommended to be 60 sec. Setting a wrong max burn duration can damage the resistors used on the thermal knifes, antenna PCBs or ICs on the PCB. If in doubt, contact GomSpace Support.

The ANT-6F UHF is configured with the GOSH command:

```
p60-dock # gssb ant6 settings
```

If no arguments are supplied then the current configuration is fetched and displayed. Below is the normal response shown.

```
Backup deploy settings:  
Minutes until deploy: 60  
Backup deployment: Not active  
Max burn duration: 60
```

Note: The `Minutes until deploy` parameter should take the initial deploy time into account. As an example:

If the initial deploy time is 60 minutes, the `Minutes until deploy` should be at least 120 minutes.

In other words, the `Minutes until deploy` should be about twice as large as the initial first deploy.

The following command shows how to modify the settings:

```
p60-dock # gssb ant6 settings 60 1 60
```

The parameters are described in the following table:

Table 4.2: Settings

| Parameter | Description |
|----------------------|---|
| Minutes until deploy | Minutes until the board will burn for max burn duration on each thermal knife |
| Backup deployment | Can be 'Not active (0)' or 'Active (1)' |
| Max burn duration | The burn duration for backup deploy and also general max burn duration |

Soft reboot (reboot)

Resets burn counters and reboot counter on ANT-6F UHF.

Sensors (temp_internal)

The ANT-6F UHF comes with a temperature sensor placed in the MCU on each board. It is read on each board with the GOSH command below:

```
p60-dock # gssb ant6 temp_internal
```

The output is in degree Celcius.

4.2.3 ANT-6F UHF Client API

The ANT-6F UHF Client API can be used to control the ANT-6F UHF from custom mission code. The Client API can be found in `gssb_ant6.h`.

All the GOSH commands in chapter [GOSH](#) are based on the ANT-6F UHF Client API.

The ANT-6F UHF API header is shown below:

```
1 #ifndef _GS_GSSB_ANT6_H_
2 #define _GS_GSSB_ANT6_H_
3
4 #include <gs/gssb/gssb_dev.h>
5
6 /**
7     Get ant6 internal temperature.
8
9     Gets the ant6 internal temperature inside MCU.
10
11    @param[in] addr I2C address.
12    @param[in] timeout_ms Timeout in milliseconds.
13    @param[out] temp_out Pointer temperature.
14 */
15 gs_error_t gs_gssb_ant6_get_internal_temp(uint8_t addr, uint16_t timeout_ms, int16_t *temp_out);
16
17 /**
18     Burn ant6 channel.
19
20     Burns a specific duration on the specific channel (side).
21
22    @param[in] addr I2C address.
23    @param[in] timeout_ms Timeout in milliseconds.
24    @param[in] channel Channel to be burned on (1 = closest MCU, 0 = farthest from MCU).
25    @param[in] duration Duration in seconds to burn.
26 */
27 gs_error_t gs_gssb_ant6_burn_channel(uint8_t addr, uint16_t timeout_ms, uint8_t channel, uint8_t duration);
28
29 /**
30     Stop ant6 burn.
31
32     Stops burning on both channels immediately.
33
34    @param[in] addr I2C address.
35    @param[in] timeout_ms Timeout in milliseconds.
36 */
37 gs_error_t gs_gssb_ant6_stop_burn_all_channels(uint8_t addr, uint16_t timeout_ms);
38
39 /**
40     Get ant6 release status.
41
42     Gets the ant6 antenna release status.
43     It tells if either one of the channels are burning.
44     It tells burn time left.
45     It tells release state.
46
47    @param[in] addr I2C address.
48    @param[in] timeout_ms Timeout in milliseconds.
49    @param[out] channel_status Pointer to release status.
50 */
51 gs_error_t gs_gssb_ant6_get_release_status(uint8_t addr, uint16_t timeout_ms, gs_gssb_ant6_release_status_t *channel_status);
52
53 /**
54     Get ant6 backup status.
55
56     Gets the ant6 antenna backup status.
57     It tells backup deployment state.
58     It tells time left until backup deploy.
59
60    @param[in] addr I2C address.
```

```
61  @param[in] timeout_ms Timeout in milliseconds.
62  @param[out] channel_status Pointer to backup status.
63  */
64  gs_error_t gs_gssb_ant6_get_backup_status(uint8_t addr, uint16_t timeout_ms, gs_gssb_
65  ↪backup_status_t * backup_status);
66
67  /**
68   * Get ant6 board status.
69   *
70   * Gets the ant6 board status.
71   * It tells reboot count.
72   * It tells uptime.
73
74   * @param[in] addr I2C address.
75   * @param[in] timeout_ms Timeout in milliseconds.
76   * @param[out] board_status Pointer to board status.
77  */
78  gs_error_t gs_gssb_ant6_get_board_status(uint8_t addr, uint16_t timeout_ms, gs_gssb_board_
79  ↪status_t * board_status);
80
81  /**
82   * Get ant6 backup settings.
83   *
84   * Gets the ant6 backup settings.
85   * It tells backup deployment state.
86   * It tells backup deployment delay.
87   * It tells maximum burn time, for both backup burn and normal burn.
88
89   * @param[in] addr I2C address.
90   * @param[in] timeout_ms Timeout in milliseconds.
91   * @param[out] settings Pointer to settings.
92  */
93  gs_error_t gs_gssb_ant6_get_backup_settings(uint8_t addr, uint16_t timeout_ms, gs_gssb_
94  ↪backup_settings_t * settings);
95
96  /**
97   * Set ant6 backup settings.
98   *
99   * Sets the ant6 backup settings.
100  * It activates/deactivates backup deployment.
101  * It sets backup deployment delay.
102  * It sets maximum burn time, for both backup burn and normal burn.
103
104  * @param[in] addr I2C address.
105  * @param[in] timeout_ms Timeout in milliseconds.
106  * @param[in] settings Pointer to settings.
107  */
108  gs_error_t gs_gssb_ant6_set_backup_settings(uint8_t addr, uint16_t timeout_ms, gs_gssb_
109  ↪backup_settings_t settings);
110
111  /**
112   * Reset ant6 counters.
113   *
114   * Resets burn counters and reboot counter on ant6.
115   *
116   * @param[in] addr I2C address.
117   * @param[in] timeout_ms Timeout in milliseconds.
118   * @param[out] settings Pointer to settings.
119  */
120  gs_error_t gs_gssb_ant6_reset_count(uint8_t addr, uint16_t timeout_ms);
121
122  #endif
```

5. Disclaimer

The information in this document is subject to change without notice and should not be construed as a commitment by GomSpace. GomSpace assumes no responsibility for any errors that may appear in this document.

In no event shall GomSpace be liable for incidental or consequential damages arising from use of this document or the software and hardware described in this document.