Aerobits

Subsystems for the
UAS intergration into
the airspace

OEM TT-RG1

Data sheet & User manual

Introduction

TT-RG1 is designed to meet requirements of remote drone identification and localization in **ASTM/ASD-STAN standard**. Using the BLE broadcast and Beacon frames technology the device provides surveillance and drone operator identification capability based on any modern mobile devices such as smartphone or tablet.

It is equipped with a high quality multi-GNSS receiver and a barometric altitude sensor.

Applications

- SAA / DAA (Sense and Avoid / Detect and Avoid)
- UAS ground stations and high-density traffic surveillance
- UTM / U-Space construction (traffic surveillance network)
- RID devices that meet direct identification concept

Main features

- Capability to work with MAVLINK devices
- BLE broadcast technology compliant with ASTM and ASD-STAN
- Interfaces: UART, USB
- Supports Bluetooth versions from 4.0 to 5.3
- Free Android application available on Google Play OpenDroneID OSM
- Integrated GNSS source and pressure sensor

For more information please contact: support@aerobits.pl.

Contents

1 Technical parameters . . . **3**
1.1 Basic technical information . . . 3
1.2 Electrical specification . . . 3
1.2.1 Absolute maximum ratings . . . 3
1.2.2 Recommended operation conditions . . . 3
1.2.3 General electrical parameters . . . 3
1.2.4 Pin definition . . . 4
1.3 Mechanical specification . . . 6
1.3.1 Dimensions . . . 6
1.3.2 Recommended layout . . . 7

2 Principle of operation . . . **8**
2.1 States of operation . . . 8
2.1.1 BOOTLOADER state . . . 8
2.1.2 RUN state . . . 8
2.1.3 CONFIGURATION state . . . 8
2.2 Transitions between states . . . 8
2.2.1 BOOTLOADER to RUN transition . . . 8
2.2.2 RUN to CONFIGURATION transition . . . 9
2.2.3 CONFIGURATION to RUN transition . . . 9
2.2.4 CONFIGURATION to BOOTLOADER transition . . . 9

3 UART configuration . . . **10**

4 Settings . . . **11**
4.1 Write settings . . . 11
4.2 Read settings . . . 11
4.3 Settings description . . . 11
4.4 Errors . . . 11
4.5 Command endings . . . 11
4.6 Uppercase and lowercase . . . 12
4.7 Available settings . . . 12
4.8 Example . . . 14

5 Commands . . . **15**
5.1 Commands in BOOTLOADER and CONFIGURATION state . . . 15
5.1.1 AT+LOCK . . . 15
5.1.2 AT+BOOT . . . 15
5.2 Commands in CONFIGURATION state . . . 15
5.2.1 AT+CONFIG . . . 15
5.2.2 AT+SETTINGS? . . . 15
5.2.3 AT+HELP . . . 15
5.2.4 AT+SETTINGS_DEFAULT . . . 16
5.2.5 AT+SERIAL_NUMBER . . . 16
5.2.6 AT+FIRMWARE_VERSION . . . 16
5.2.7 AT+REBOOT . . . 16
5.2.8 AT+REBOOT_BOOTLOADER . . . 16
5.3 Commands in RUN state . . . 16

6 Protocols . . . **17**
6.1 CSV protocol (REMOTE) . . . 17
6.1.1 CRC . . . 17
6.1.2 RemoteID Aircraft message . . . 17
6.1.3 Statistics message . . . 19

1 Technical parameters

1.1 Basic technical information

Parameter	Description	Min.	Typ.	Max.	Unit
Carrier frequency ADS-B		-	1090	-	MHz
RX sensitivity GNSS		-	approx. -167	-	dBm
RX sensitivity Bluetooth		-	approx. -85	-	dBm
TX power Bluetooth		-	-	18	dBm
UART (baud)	AT commands	57600	115200	3000000	bps

Table 1: General technical parameters.

1.2 Electrical specification

1.2.1 Absolute maximum ratings

Parameter	Min.	Max.	Unit
Storage temperature	-5	+40	°C
Supply voltage (VCC)	3.0	3.6	DCV
Other pin voltage	-0.3	VCC + 0.3	DCV
RF input Bluetooth	-	+5	dBm

Table 2: Absolute maximum ratings.

1.2.2 Recommended operation conditions

Parameter	Min.	Max.	Unit
Operation temperature	-40	+85	°C
Supply voltage (VCC)	3.0	3.6	DCV

Table 3: Recommended operation conditions.

1.2.3 General electrical parameters

Parameter	Description	Min.	Typ.	Max.	Unit
Current consumption		-	70	-	mA
Input Low Voltage	RESET, UARTs, CAN, USB, SPI, I2C	-0.3	-	0.8	DCV
Input High Voltage	RESET, UARTs, CAN, USB, SPI, I2C, GPIO	VCC - 0.7	-	VCC + 0.3	DCV
Output Low Voltage	UARTs, CAN, USB, I2C, SPI, GPIO	-	-	0.4	DCV
Output High Voltage	UARTs, CAN, USB, I2C, SPI, GPIO	VCC - 0.4	-	-	DCV

Table 4: General electrical parameters.

1.2.4 Pin definition

Pin arrangement of OEM TT-RG1 is shown on the figure below (1).

Pin number	Pin Name	Pin Type	Description
1	VCC	Power	3.3V (digital supply)
2	TX1	CMOS Output	UART1 GNSS module NMEA data output
3	RX1	CMOS Input	UART1 GNSS module data input - not recommend sending anything due to possible malfunctioning main module
4	TX0	CMOS Output	UART0 data output
5	RX0	CMOS Input	UART0 data input
6	-	N/C	No commercial use (keep floating)
7	-	N/C	No commercial use (keep floating)
8	GND	GND	Common ground
9	I/O	N/C	General purpose input/output (keep floating)
10	-	N/C	No commercial use (keep floating)
11	GND	GND	Common ground
12	GNSS_RF	IN	GNSS RF Input (antenna)
13	GND	GND	Common ground
14	GND	GND	Common ground
15	BT_RF	RF Input	RF Input (antenna)
16	GND	GND	Common ground
17	BT_LED	CMOS Output	LED Output (digital)
18	PPS	CMOS Input	1PPS GNSS signal
19	GND	GND	Common ground
20	B/C	CMOS Input	Bootloader / Configuration mode
21	SDA	Bi-directional	I2C Data line
22	SCL	Bi-directional	I2C Clock line
23	RST_GNSS	CMOS Input	Reset input / active low
24	USB_DP	Bi-directional	USB+
25	USB_DM	Bi-directional	USB-
26	RESET	CMOS Input	Reset input / active low

Table 5: Pin definitions of OEM TT-RG1.

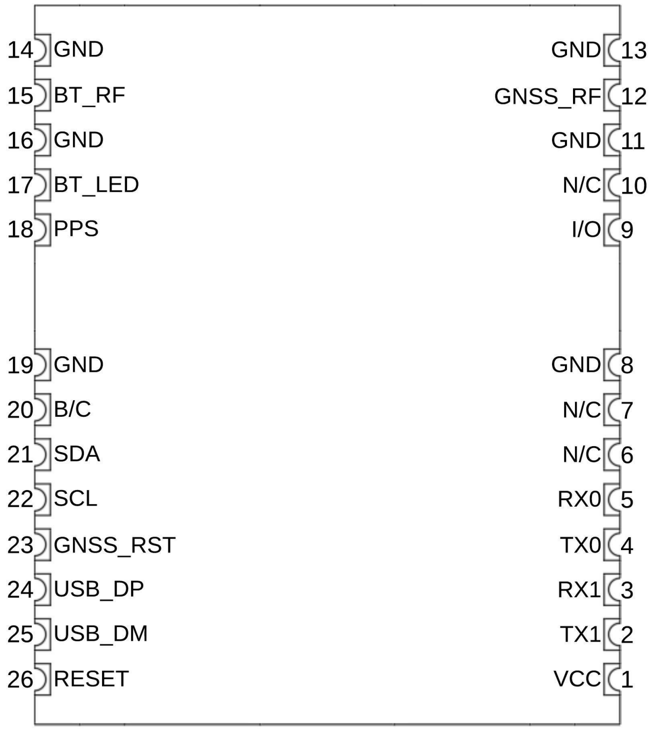

Figure 1: Pin arrangement of OEM TT-RG1.

1.3 Mechanical specification

1.3.1 Dimensions

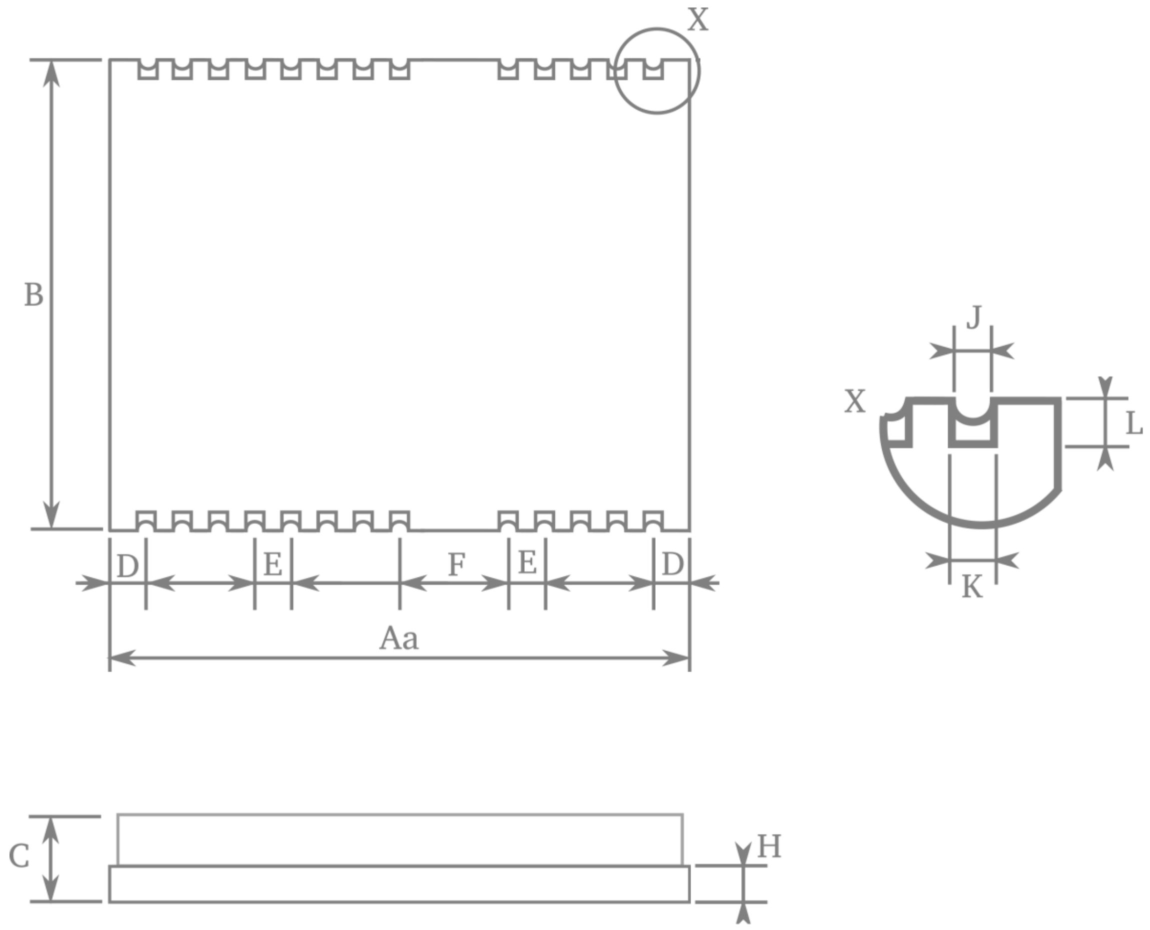

Figure 2: Mechanical drawing of OEM TT-RG1

Symbol	Min. (mm)	Typ. (mm)	Max. (mm)
A	15.9	16.0	16.1
B	12.9	13.0	13.1
C	2.6	2.7	2.8
D	0.9	1.0	1.1
E	0.9	1.0	1.1
F	2.9	3.0	3.1
H	0.6	0.7	0.8
J	0.4	0.5	0.6
K	0.6	0.7	0.8
L	0.7	0.8	0.9

Table 6: Dimensions and tolerances.

1.3.2 Recommended layout

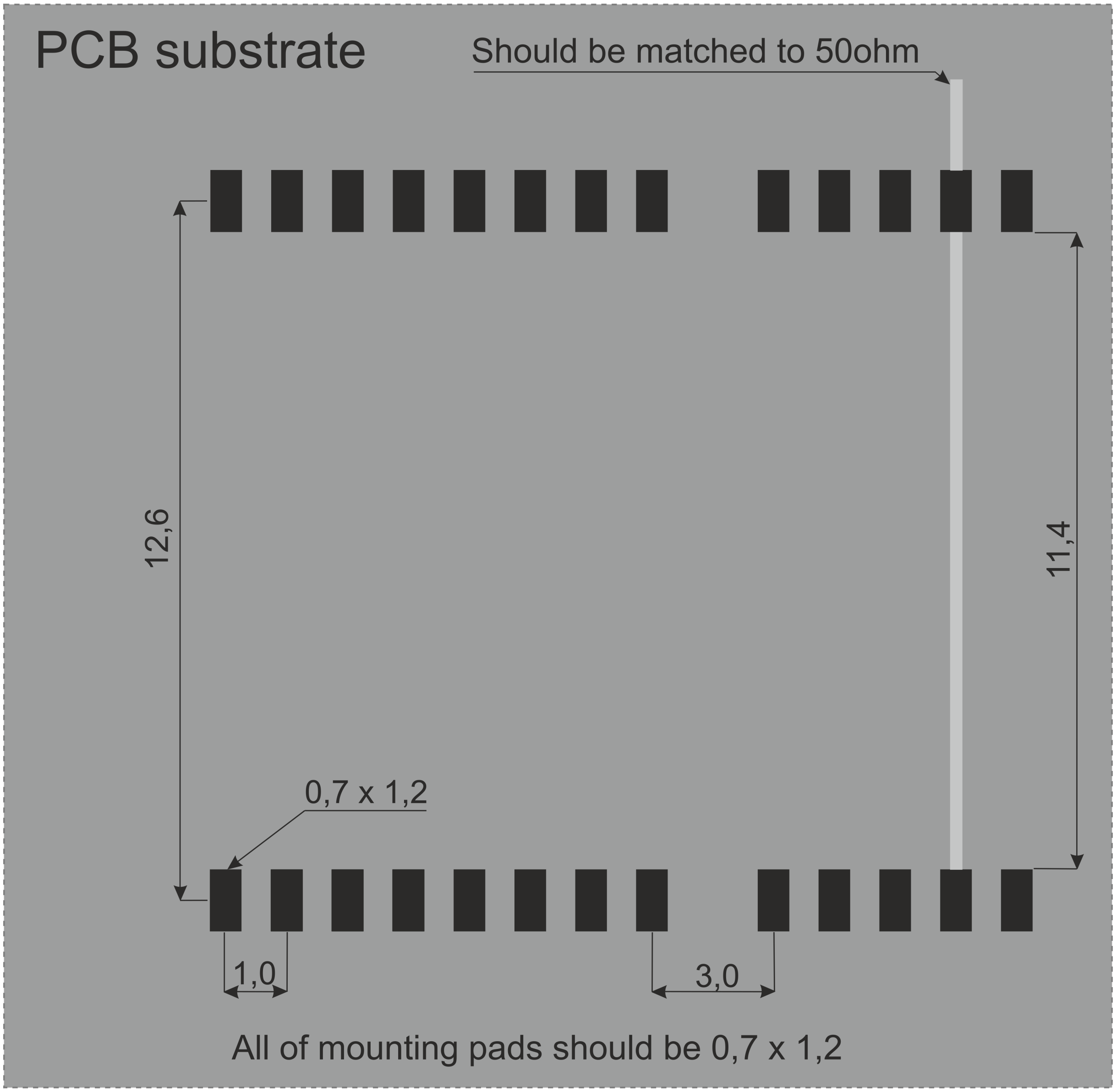

Figure 3: Footprint of OEM TT-RG1

NOTE: In case of OEM the RF inputs indicated in the footprint(3) should be matched to 50ohm.

2 Principle of operation

During work module goes through multiple states. In each state operation of the module is different. Each state and each transition is described in paragraphs below.

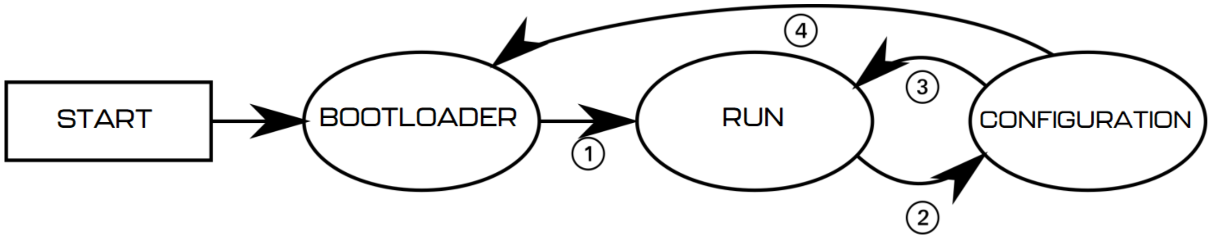

Figure 4: State machine of OEM TT-RG1

2.1 States of operation

2.1.1 BOOTLOADER state

This is an initial state of OEM TT-RG1 after restart. Firmware update is possible here. Typically module transits automatically to RUN state. It is possible to lock module in this state (prevent transition to RUN state) using one of BOOTLOADER triggers. UART baud is constant and is set to 115200bps. After powering up module, it stays in this state for up to 3 seconds. If no BOOTLOADER trigger is present, module will transit to RUN state. Firmware upgrade is possible using Micro ADS-B App software. For automated firmware upgrading scenarios, aerobits_updater software is available. To acquire this program please contact: support@aerobits.pl.

2.1.2 RUN state

In this state module is working and receiving the data from aircrafts. It uses selected protocol to transmit received and decoded data to the host system. In this state of operation module settings are loaded from non-volatile internal memory, including main UART interface's baud.

2.1.3 CONFIGURATION state

In this mode change of stored settings is possible. Operation of the module is stopped and baud is set to fixed 115200bps. Change of settings is done by using AT-commands. Changes to settings are stored in non-volatile memory on exiting this state. Additional set of commands is also available in this state, allowing to e.g. reboot module into BOOTLOADER state, check serial number and firmware version. It is possible to lock module in this state (similarly to BOOTLOADER) using suitable command.

2.2 Transitions between states

For each of state transitions, different conditions must be met, which are described below. Generally, the only stable state is RUN. Module always tends to transit into this state. Moving to other states requires host to take some action.

2.2.1 BOOTLOADER to RUN transition

BOOTLOADER state is semi-stable: the module requires additional action to stay in BOOTLOADER state. The transition to RUN state will occur automatically after short period of time if no action will be taken. To prevent transition from BOOTLOADER state, one of following actions must be processed:

- Pull BOOT/CONFIG pin low during start of module
- Send `AT+LOCK=1` command while device is in BOOTLOADER state (always after power on for up to 3s)
- Send `AT+REBOOT_BOOTLOADER` command in CONFIGURATION state. This will move to BOOTLOADER state and will lock module in this state.

If none of above conditions are met, the module will try to transit into RUN state. Firstly it will check firmware integrity. When firmware integrity is confirmed, module will transit into RUN state, if not, it will stay in BOOTLOADER state.

To transit into RUN state:

- Release or pull high BOOT/CONFIG pin
- If module is locked, send `AT+LOCK=0` command

When module enters RUN mode it will send `AT+RUN_START` command.

2.2.2 RUN to CONFIGURATION transition

To transit from RUN into CONFIGURATION state, host should do one of the following:

- Pull BOOT/CONFIG pin low
- Send `AT+CONFIG=1` (using current baud). This method is not recommended, because module will support multiple protocols in future and Aerobits Sp. z o.o. cannot ensure that this command will be present in all protocols.

When module leaves RUN state it sends `AT+RUN_END` message, then `AT+CONFIG_START` message on entering CONFIGURATION state. The former is sent using baud from settings, the latter always uses 115200bps baud.

2.2.3 CONFIGURATION to RUN transition

To transit from CONFIGURATION into RUN state, host should do one of the following:

- Release or pull high BOOT/CONFIG pin
- Send `AT+CONFIG=0` command.

When module leaves CONFIGURATION state it sends `AT+CONFIG_END` message, then `AT+RUN_START` message on entering RUN state. The former is always sent using 115200bps baud, the latter uses baud from settings.

2.2.4 CONFIGURATION to BOOTLOADER transition

To transit from CONFIGURATION into BOOTLOADER state, host should do one of the following:

- Send `AT+REBOOT_BOOTLOADER` command.
- Send `AT+REBOOT` and when module enters BOOTLOADER state, prevent transition to RUN state.

When entering the bootloader state, the module sends `AT+BOOTLOADER_START` .

3 UART configuration

Communication between module and host device is done using UART interface.

In CONFIGURATION and BOOTLOADER state transmission baud is fixed at 115200bps.

The UART interface uses settings as described in table 7.

UART Settings				
Parameter	Min.	Typ.	Max	Unit
Baud	57600	115200	3000000	bps
Stop Bits Number	-	1	-	-
Flow Control	-	None	-	-
Parity Bit	-	None	-	-

Table 7: UART settings.

4 Settings

In RUN state, operation of the module is determined based on stored settings. Settings can be changed in CONFIGURATION state using AT-commands. Settings can be written and read.

NOTE: New values of settings are saved in non-volatile memory when transitioning from CONFIGURATION to RUN state.

Settings are restored from non-volatile memory during transition from BOOT do RUN state. If settings become corrupted due to memory fault, power loss during save, or any other kind of failure, the settings restoration will fail, loading default values and displaying the `AT+ERROR (Settings missing, loaded default)` message as a result. This behavior will occur for each device boot until new settings are written by the user.

4.1 Write settings

After writing a new valid value to a setting, an `AT+OK` response is always sent.

`AT+SETTING=VALUE`
For example `AT+ADSB_MODE=3`
Response: `AT+OK`

4.2 Read settings

`AT+SETTING?`
For example: `AT+ADSB_MODE?`
Response: `AT+ADSB_MODE=3`

4.3 Settings description

`AT+SETTING=?`
For example: `AT+ADSB_MODE=?`
Response:

```
Setting: ADSB_MODE
Description: ADS-B receiver mode: 0 - Disabled, 1 - Raw only, 2 - Decoded only,
 3 - Raw and Decoded
Access: Read Write
Type: Integer decimal
Range (min.): 0
Range (max.): 3
Preserved: 1
Requires restart: 0
```

4.4 Errors

Errors are reported using following structure:
`AT+ERROR (DESCRIPTION)`
DESCRIPTION is optional and contains information about error.

4.5 Command endings

Every command must be ended with one of the following character sequences: “\n”, “\r” or “\r\n”. Commands without suitable ending will be ignored.

4.6 Uppercase and lowercase

All characters (except preceding `AT+`) used in command can be both uppercase and lowercase, so following commands are equal:

```
AT+ADSB_MODE?
AT+aDsB_mOdE?
```

NOTE: This statement is true in configuration state, not in bootloader state. in bootloader state all letters must be uppercase.

4.7 Available settings

Setting	Min	Max	Def	Comment
BAUDRATE	0	2	0	Baudrate in RUN state 0 - 115200bps 1 - 921600bps 2 - 3000000bps
GNSS_LOG	0	2	0	GNSS NMEA forwarding 0 - No forwarding 1 - RMC Messages only 2 - All
DRONE_ID_BASIC_BROADCAST_PERIOD	200	3000	2900	Basic frame broadcast period in [ms]
DRONE_ID_LOCALIZATION_BROADCAST_PERIOD	100	1000	900	Localization frame broadcast period in [ms]
DRONE_ID_ADVERTISING_ENABLE	0	1	1	Enable Bluetooth advertising
DRONE_ID_SCAN_ENABLE	0	1	0	Enable Bluetooth scan
DRONE_ID_HEIGHT_TYPE	0	1	0	Device Height type 0 - Above Takeoff 1 - AGL
DRONE_ID_MAVLINK_CONNECTION_TIMEOUT	0	30	5	Mavlink connection timeout in seconds
DRONE_ID_OPERATOR_ID	-	-	-	Operator ID (20 bytes)
DRONE_ID_OPERATOR_ID_TYPE	0	255	0	Operator ID type 0 - Operator ID 201-255 - Available for private use
DRONE_ID_OPERATIONAL_STATUS	0	2	0	Operation status 0 - Undeclared 1 - Ground 2 - Airborne
DRONE_ID_SELF_ID	-	-	-	Self ID (20 bytes)

DRONE_ID_SELF_ID_TYPE	0	255	0	Self ID type 0 - Text Description 201-255 - Available for private use
DRONE_ID_MODE	0	2	1	Determines mavlink reception 0 - Full mavlink support 1 - Ignore all mavlink messages 2 - Ignore only location messages
DRONE_ID_TYPE	0	3	1	UAS ID type 0 - None 1 - Serial Number 2 - CAA Assigned Registration ID 3 - UTM Assigned UUID
DRONE_ID_UAS_TYPE	0	15	0	Specification of the type of UAS 0 - None 1 - Aeroplane 2 - Helicopter or Multirotor 3 - Gyroplane 4 - Hybrid Lift 5 - Ornithopter 6 - Glider 7 - Kite 8 - Free Balloon 9 - Captive Balloon 10 - Airship 11 - Free Fall 12 - Rocket 13 - Tethered Powered Aircraft 14 - Ground Obstacle 15 - Other
DRONE_ID_UAS_ID	-	-	Device serial number	UAS ID (20 bytes)
DRONE_ID_DRONE_CATEGORY	0	3	0	Drone category class 0 - Undeclared 1 - Open 2 - Specific 3 - Certified

DRONE_ID_OPERATION_CATEGORY	0	7	0	Operation category 0 - Undeclared 1 - Class 0 2 - Class 1 3 - Class 2 4 - Class 3 5 - Class 4 6 - Class 5 7 - Class 6
PRESSURE_LOG	0	1	0	Show barometer log

Table 8: Settings

4.8 Example

As an example, to switch OEM TT-RG1 module to CSV protocol, one should send following commands. “<<” indicates command sent to module, “>>” is a response.

```
<< AT+CONFIG=1\r\n
>> AT+OK\r\n
<< AT+ADSB_MODE=2\r\n
>> AT+OK\r\n
<< AT+ADSB_DECODED_PROTOCOL=1\r\n
>> AT+OK\r\n
<< AT+CONFIG=0\r\n
>> AT+OK\r\n
```

5 Commands

Apart from settings, module supports set of additional commands. Format of this commands are similar to those used for settings, but they do not affect operation of module in RUN state.

5.1 Commands in BOOTLOADER and CONFIGURATION state

5.1.1 AT+LOCK

`AT+LOCK=1` - Set lock to enforce staying in BOOTLOADER or CONFIGURATION state
`AT+LOCK=0` - Remove lock
`AT+LOCK?` - Check if lock is set

5.1.2 AT+BOOT

`AT+BOOT?` - Check if module is in BOOTLOADER state

Response:

`AT+BOOT=0` - module in CONFIGURATION state
`AT+BOOT=1` - module in BOOTLOADER state

5.2 Commands in CONFIGURATION state

5.2.1 AT+CONFIG

`AT+CONFIG=0` - Transition to RUN state.
`AT+CONFIG?` - Check if module is in CONFIGURATION state.

Response:

`AT+CONFIG=0` - module in RUN state
`AT+CONFIG=1` - module in CONFIGURATION state (baudrate 115200)
`AT+CONFIG=2` - module in CONFIGURATION state (baudrate as set)

5.2.2 AT+SETTINGS?

`AT+SETTINGS?` - List all settings. Example output:

```
AT+ADSB_MODE=2
AT+ADSB_DECODED_PROTOCOL=1
AT+SUBPROTOCOL=0
AT+BAUDRATE=0
```

5.2.3 AT+HELP

`AT+HELP` - Show all settings and commands with descriptions. Example output:

```
SETTINGS:
AT+ADSB_MODE=2               [0 - Disabled,1 - Raw, 2 - Decoded, 3 - Raw and Decoded]
AT+ADSB_DECODED_PROTOCOL=1   [0 - None, 1 - CSV, 2 - Mavlink, 3 - JSON,
4 - GDL90, 5 - ASTERIX]
COMMANDS:
AT+HELP                [Show this help]
AT+TEST                [Responds "AT+OK"]
AT+SETTINGS_DEFAULT    [Load default settings]
AT+REBOOT              [Reboot system]
```

5.2.4 AT+SETTINGS_DEFAULT

`AT+SETTINGS_DEFAULT` - Set all settings to their default value.

5.2.5 AT+SERIAL_NUMBER

`AT+SERIAL_NUMBER?` - Read serial number of module.

Response:

`AT+SERIAL_NUMBER=07-0001337`

5.2.6 AT+FIRMWARE_VERSION

`AT+FIRMWARE_VERSION?` - Read firmware version of module.

Response:

`AT+FIRMWARE_VERSION=10101017(May 11 2018)`

5.2.7 AT+REBOOT

`AT+REBOOT` - Restart module.

5.2.8 AT+REBOOT_BOOTLOADER

`AT+REBOOT_BOOTLOADER` - Restart module to BOOTLOADER state.

NOTE: This command also sets lock.

5.3 Commands in RUN state

`AT+CONFIG=1` - transition to CONFIGURATION state (baudrate 115200).
`AT+CONFIG=2` - transition to CONFIGURATION state (baudrate as set).

NOTE: This command also sets lock.

6 Protocols

6.1 CSV protocol (REMOTE)

CSV protocol is simple text protocol, that allows fast integration and analysis of tracked aircrafts. CSV messages start with '#' character and ends with "\r\n" characters. There are following types of messages:

NOTE: In future versions, additional comma-separated fields may be introduced to any CSV protocol message, just before CRC field, which is guaranteed to be at the end of message. All prior fields are guaranteed to remain in same order.

6.1.1 CRC

Each CSV message includes CRC value for consistency check. CRC value is calculated using standard CRC16 algorithm and its value is based on every character in frame starting from '#' to last comma ',' (excluding last comma). After calculation, value is appended to frame using hexadecimal coding. Example function for calculating CRC is shown below.

```
uint16_t crc16(const uint8_t* data_p, uint32_t length){
    uint8_t x;
    uint16_t crc = 0xFFFF;
    while (length--){
        x = crc>>8 ^ *data_p++;
        x ^= x>>4;
         crc = (crc<<8) ^ ((uint16_t)(x<<12)) ^ ((uint16_t)(x<<5)) ^ ((uint16_t)x);
    }
    return swap16(crc);
}
```

6.1.2 RemoteID Aircraft message

This message describes state vector of aircraft determined from remoteID messages and is sent once per second. The message format is as follows:

```
#B4\B5:UAS_ID,ID_TYPE,UAS_TYPE,LAT,LON,HEIGHT,ALT_GEO,ALT_BARO,TRACK,VELH,
VELV,STATUS_FLAG,OPERATOR_ID,OPERATOR_ID_TYPE,OPERATOR_LAT,OPERATOR_LON
,OPERATOR_LOC_TYPE,TIMES,RSSI,CRC\r\n
```

#B4\B5	Aircraft message start indicator	Example value
UAS_ID	aircraft ID	18099300000132
ID_TYPE	Flags bitfield, see table 12	1
UAS_TYPE	Callsign of aircraft, see table 13	2
LAT	Latitude, in degrees, accuracy 0.6 degree	57.57634
LON	Longitude, in degrees, accuracy 0.6 degree	17.59554
HEIGHT	Height based on start up altitude, in meters	0.5
ALT_GEO	Geometric altitude, in meters	50
ALT_BARO	Barometric altitude, in meters	50
TRACK	Track of aircraft, in degrees [0,360)	35
VELH	Horizontal velocity of aircraft, in m/s, accuracy 0.1 m/s	464
VELV	Vertical velocity of aircraft, in m/s, accuracy 0.1 m/s	-1344
STATUS_FLAG	Operation status	0
OPERATOR_ID	The operator number from local FAA department	AAABBBBBBBBBBBBC-DDD
OPERATOR_ID_TYPE	Specific type of Operator ID	5
OPERATOR_LAT	The operator latitude in degrees, accuracy 0.6 degree	57.52614
OPERATOR_LON	The operator longitude in degrees, accuracy 0.6 degree	17.60154
OPERATOR_LOC_TYPE	The operator location type	0
TIMES	Timestamp of the sent frame expressed in seconds since current hour, accuracy 0.1 s-1.5 s	408.5
RSSI	Signal strength, in dBm	-92
SELF_ID_TYPE	Self id type 10	0
SELF_ID	Self id	
FTYPE_TYPE	Frame type 11	15
MAC	MAC address	df:a5:c3:84:78:66
CRC	CRC16 (described in CRC section)	2D3E

Table 9: Descriptions of RemoteID message fields.

Whereby the following prefixes mean:

- #B4 - Bluetooth 4.0(Legacy) frame
- #B5 - Bluetooth 5.0 frame

NOTE: Referring to the ASD-STAN prEN 4709-002 standard, our product displays all the required information (ASD-STAN prEN 4709-002 Table 1 - Data Dictionary), optional data is only available upon special request.

Below is a list off self Id types returned in Self Id value field.

Self Id Type value	Description
0	Text Description.
1	Emergency Description.
2	Extended Status Description.
3–200	Reserved.
201–255	Available for private use.

Table 10: RemoteID UAS Self Id type in CSV protocol.

Below is a list off frame types returned in Frame Type value field.

Id Type value	Description
0	Basic ID.
1	Location.
3	Self ID.
4	System.
5	Operator ID.
15	Packed all in one.

Table 11: RemoteID UAS ID Frame type in CSV protocol.

Below is a list of ID types returned in ID Type value field.

Id Type value	Description
0	None.
1	Serial Number.
2	CAA Assigned Registration ID.
3	UTM Assigned UUID.

Table 12: RemoteID UAS ID Type category values in CSV protocol.

Below is a list of emitter category values returned in ECAT value field.

ECAT value	Description
0	None.
1	Aeroplane.
2	Helicopter or Multirotor.
3	Gyroplane.
4	Hybrid Lift.
5	Ornithopter.
6	Glider.
7	Kite.
8	Free Balloon.
9	Captive Balloon.
10	Airship.
11	Free Fall.
12	Rocket.
13	Tethered Powered Aircraft.
14	Ground Obstacle.
15	Other.

Table 13: RemoteID ID Type category values in CSV protocol.

If data of any field of frame is not available, then it is transmitted as empty. For example:

```
#B5:18099300000170,1,0,53.3960175,14.6283543,-0.5,58.0,86.5,0,0.0,0.0,
0,,0,0.0000000,0.0000000,0,103.7,-46,0,,15,84:f7:03:28:e3:1a,420C\r\n
#B5:18099300000170,1,0,53.3960175,14.6283543,-0.5,58.0,86.5,0,0.0,0.0,
0,,0,0.0000000,0.0000000,0,103.7,-46,0,,15,84:f7:03:28:e3:1a,420C\r\n
```

NOTE: RSSI is measured based on analog RF signal.

6.1.3 Statistics message

This message contains some useful statistics about operation of module. Format of that frame is shown below:

```
#SR:FPS,FPB4_0S,FPB4_1S,FPB4_3S,FPB4_4S,FPB4_5S,FPB5S,RES,RES,CRC
```

#SR	Statistics message start indicator	Example
FPS	Number of frame received in last second %	1
FPB4_0S	Number of frame received in last second %	1
FPB4_1S	Number of legacy basic ID Bluetooth 4.0 frame send in last second	1
FPB4_3S	Number of legacy location Bluetooth 4.0 frame send in last second	1
FPB4_4S	Number of legacy self ID Bluetooth 4.0 frame send in last second	1
FPB4_5S	Number of legacy system Bluetooth 4.0 frame send in last second	1
FPB5S	Number of packed all in one Bluetooth 5.0 frame send in last second	1
RES	Reserved for future use	-
RES	Reserved for future use	-
CRC	CRC16 (described in CRC section)	2D3E

Table 14: Statistics message fields.

Please read carefully