USER MANUAL

UHF Transceiver Type II

UHF TRANSCEIVER TYPE II

USER MANUAL

This user manual details the applications, features and operation of EnduroSat's UHF Transceiver Type II module.

Please read this manual before unpacking and using the module to ensure safe and proper use.

Figure 1 – UHF transceiver type II module

1 CHANGE LOG

Date	Version	Note
02/11/2017	Rev 1.0	Initial revision
02/04/2018	Rev 1.2	Detailed antenna release connector pinout
24/04/2018	Rev 1.3	Changes in text
15/06/2018	Rev 1.4	Pinout changes
21/11/2018	Rev 1.5	Technical writing enhancements
13/08/2019	Rev 1.6	Changes in text and contents
17/01/2020	Rev 1.7	New commands, FW update and Bootloader description
27/01/2020	Rev 1.8	Technical writing enhancements

2 ACRONYMS LIST

AX.25	Amateur X.25 data link layer protocol
AFC	Automatic Frequency Correction
ASCII	American Standard Code for Information Interchange
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
ECSS	European Cooperation for Space Standardization
EMI	Electromagnetic interference
ESD	Electrostatic Discharge
ESTTC	EnduroSat's Telemetry and Telecommand
FRAM	Ferroelectric Random-Access Memory
FSK	Frequency-Shift Keying
GEVS	General Environmental Verification Standard
GFSK	Gaussian Frequency Shift Keying
GMSK	Gaussian Minimum Shift Keying
GND	Ground
IC	Integrated Circuit
I^2C	Inter-Integrated Circuit
LNA	Low-Noise Amplifier
MCU	Microcontroller Unit
MMCX	Micro-miniature Coaxial
NF	Noise Figure
OBC	On-Board Computer
OOK	On-Off Keying
PA	Power Amplifier
PCB	Printed Circuit Board
PER	Packet Error Ratio
RF	Radio Frequency
Rx	Receive
SCW	Status Control Word
SNR	Signal-to-noise ratio
SPI	Serial Peripheral Interface
SEL	Single Event Latch-up
Tx	Transmit
Tx/Rx	Transmit and Receive
UART	Universal Asynchronous Receiver/Transmitter
UHF	Ultra-High Frequency
USB	Universal Serial Bus
VCP	Virtual Com Port

3 SYSTEM OVERVIEW

EnduroSat's UHF Transceiver Type II module operates in the commercial frequency band 400 to 403 MHz (Tx/Rx), and the amateur frequency band 430 to 440 MHz (Tx/Rx). Furthermore, it features configurable data rates, which can be changed whilst the satellite is in orbit.

The output power can also be tuned in order to maximize the link budget depending on the orbital altitude, ground station performance and desired minimum elevation angle for communication. The typical output power is 1W (30dBm) with versions of the system allowing the power to be boosted up to 2W (33dBm).

The system has a USB (virtual COM port) which allows the connection of EnduroSat's PC Software or a third-party terminal program for monitoring and configuring. The module is designed to fit within a CubeSat, but a second module can also be integrated into a ground station to easily create a complete uplink and downlink communication solution. The module uses the popular AX.25 data protocol for periodical beacon messages.

The UHF transceiver is fully encapsulated in an aluminum box which is designed to dissipate the heat from the power amplifier, reduce EMI and EMC, and protects the electronics from particle radiation.

4 HIGHLIGHTED FEATURES

- Frequency range (Tx/Rx): 400 to 403 MHz, and 430 to 440 MHz
- Modulation: OOK, GMSK, 2FSK, 4FSK and 4GFSK are optional, 2GFSK (by default)
- Automatic Frequency Correction
- Configurable AX.25 telemetry beacon broadcast and automatic AX.25 telemetry decoding
- Morse code
- Audio beacon
- Protocols: ESTTC, transparent, AX.25
- Maximum transmit power: 1 W (customizable up to 2 W)
- Power supply: 3.3 V (customizable to 5 V)
- Ultra-low power MCU with FRAM
- External FRAM
- Typical current consumption during receive mode (idle mode) (Rx): 25mA @ 3.3V
- Frequency stability: +/- 2.5 ppm
- Data rate in the air: up to 19.2kbps (optional up to 100 kbps)
- Sensitivity: up to -121 dBm
- Communication interfaces: UART / I^2C / USB (VCP) / RS485 (opt.) / CAN (opt.)

- Local and remote (in-flight) secured application firmware update
- Antenna release connector for EnduroSat's UHF Antenna
- Type: Half-duplex
- Weight: 94g

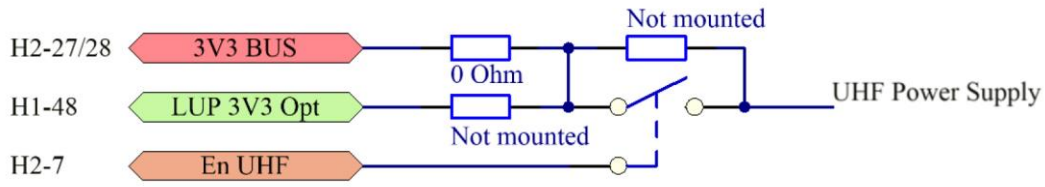
5 SYSTEM DESCRIPTION

The UHF Transceiver Type II works in the frequency range 400 to 403 MHz, and 430 to 440 MHz. Different modulation schemes are available, among them 2FSK, 4FSK, the spectrum efficient 2GFSK, 4GFSK, GMSK, and OOK as well.

The power supply block diagram is shown in Figure 2.

By default, the transceiver works in half-duplex mode with configurable data rate and modulation index (m). It has the option to transmit a beacon signal with predefined information. The default data and command interfaces are UART and I^2C. Optionally, RS485 and CAN could be used. Additionally, it has a USB (Virtual COM Port) for configuration by EnduroSat's PC Software or a third-party terminal program. The module can also be used in a UHF ground station so that the AX.25 modem, receiver and transmitter sections can be used for communication purposes. All communication interfaces have dedicated hardware buffers for protection and when the Enable UHF = OFF (see Fig. 2) then the interfaces of the module go high impedance.

The module can be powered either with 3.3V or 3.3V with latch-up protection with the corresponding enable pin pulled up in high state. Custom versions of the UHF transceiver can be powered from the 5V bus.



Enable UHF:

ON 1V – 5.5V

OFF 0V – 0.3V

Figure 2: Power Supply Diagram

At the core of the RF section there is a high-performance transceiver IC. In the transmitter part of the device, there is a high efficiency RF power amplifier which by default is powered with 3.3V giving an output power up to 1W. If a 5V bus is used the output power can be boosted up to 2W in specific versions of the device. In the receiver part, a low noise amplifier with a maximum Noise Figure (NF) of 0.9dB improves the overall receiver performance in terms of sensitivity which is specified down to -

121dBm. The device is equipped with an antenna release connector that facilitates connection and deployment of EnduroSat's UHF antenna.

The device uses a PC-104 connector which is suitable for stackable configurations of other satellite modules. The RF connector to the antenna is an MMCX.

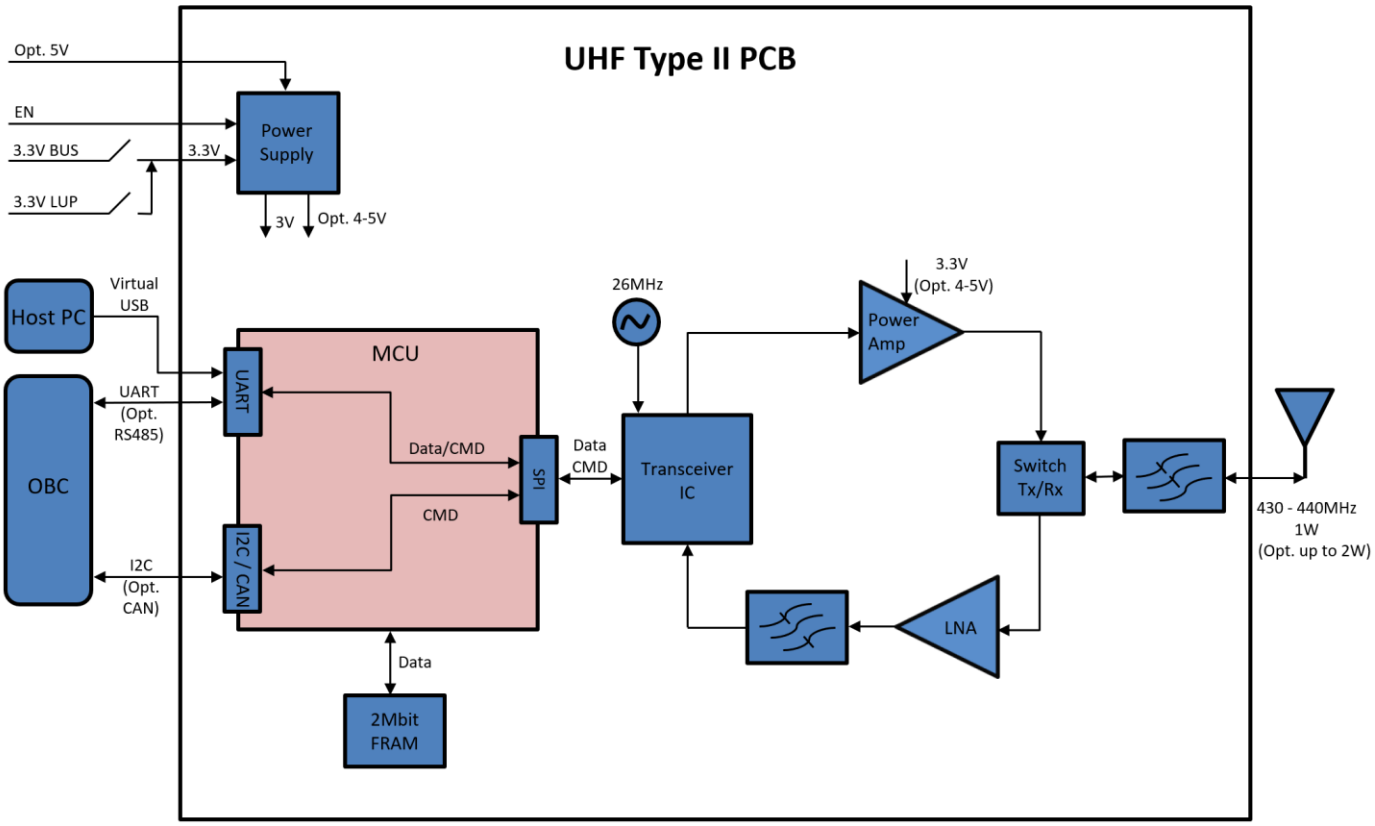


Figure 3: Functional Block Diagram of the UHF Transceiver Type II

6 ELECTRICAL CHARACTERISTICS

Parameter	Condition	Min	Typical value	Max
Supply Voltage [V]		3.2	3.3[1]	5V Opt.
Current Consumption [mA]	Receive mode (Idle mode)	20	25[2]	30
	Transmit mode		413[3]	
	Continuous wave mode	700	780[4]	800
Operating temperature [°C]		-35		80

[1] This voltage directly supplies the internal power amplifier. Changes in the supply voltage will reflect in the output transmit power.

[2] Typical current consumption at 3.3Vdc power supply using only the UART and I^2C interfaces (CAN and RS485 are turned off).

[3] Typical current consumption depends on the ratio of transmit vs receive mode duration. For 50% Tx / 50%, then the consumption would be as follows: 0.5*800 mA (Tx CW) + 0.5*25 mA (Rx) = 413 mA @ 3.3 V.

[4] Typical current consumption at 3.3Vdc and 435MHz working frequency

7 RF CHARACTERISTICS

7.1 Transmitter

Parameter	Unit	Min	Typical value	Max
Freq. Range	MHz		400÷403 435÷438	
Output Power	W		1	2
Spurious level	dBc	60		
Baud Rate in The Air	bps	1200		19200

Table 1. RF characteristics in transmit mode

7.2 Receiver

Parameter	Condition	Unit	Min	Typical value	Max
Freq. Range		MHz		400÷403 435÷438	
Baud Rate In The Air		bps	1200		19200
SNR	PER<1% @9600bps	dB		14	
Input Power		dBm			10

Table 2. RF characteristics in receive mode

8 CONNECTOR PINOUT

8.1 Connectors Locations

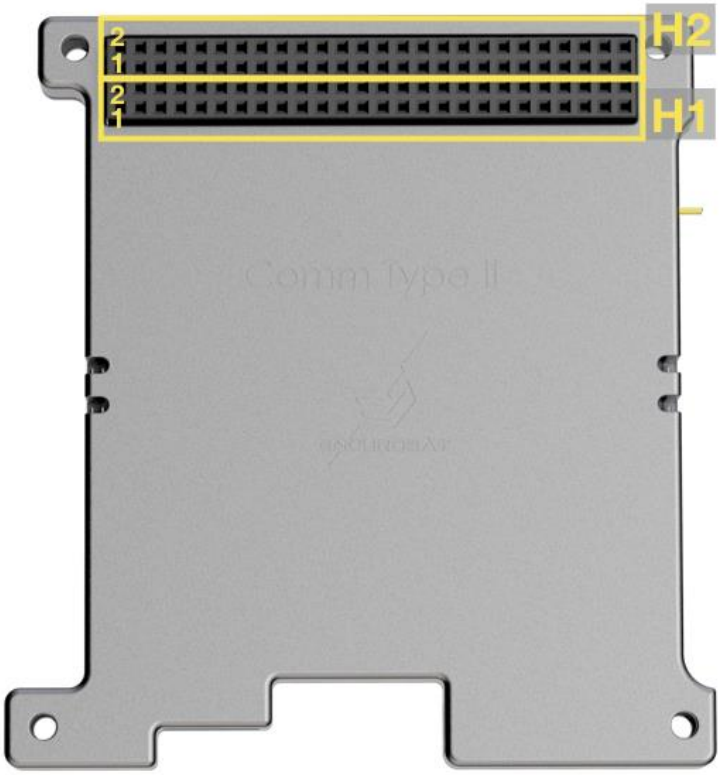


Figure 4: Main Stack Connector Location

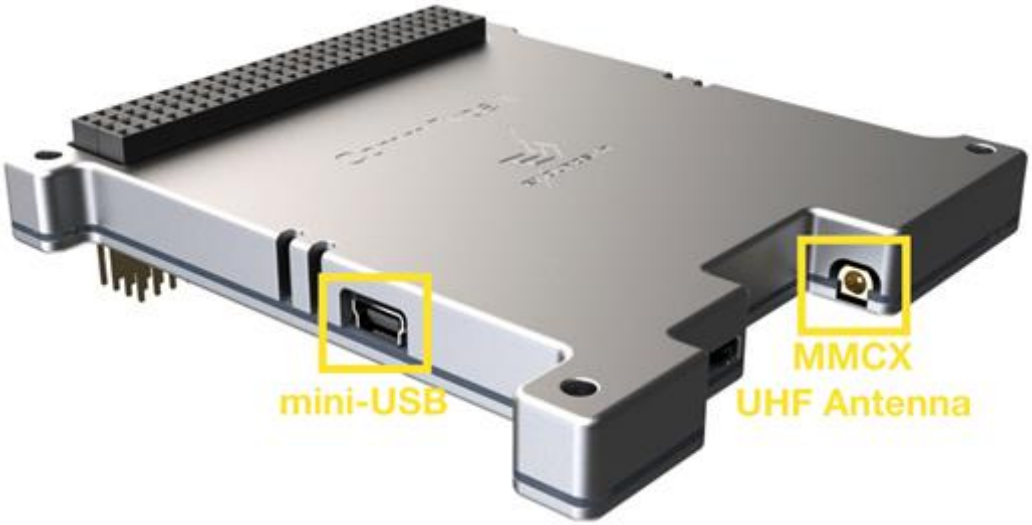


Figure 5: MMCX and Mini-USB Connector Location

Figure 6: Antenna Release Connector and Jumper Location

8.2 H1 – Stack Connector

Pin	Mnemonic	Description
H1-1	CAN L	CAN communication Low (3.3V)
H1-3	CAN H	CAN communication High (3.3V)
H1-22	RS485A	RS-485 Driver output or receiver input (complementary to B)
H1-24	RS485B	RS-485 Driver output or receiver input (complementary to A)
H1-33	RxD	UART receive data
H1-35	TxD	UART transmit data
H1-41	I^2C SDA	I2C data
H1-43	I^2C SCL	I2C clock
H1-48	LUP 3V3	Latch-up protected 3.3V power bus (input)

Table 3. H1 Stack Connector pinout

8.3 H2 – Stack Connector

Pin	Mnemonic	Description
H2-5	En UHF Opt.	UHF power enable pin (optional)[1]
H2-7	En UHF	UHF power enable pin
H2-25	+5V	+5V BUS Power supply (optional)
H2-26	+5V	+5V BUS Power supply (optional)
H2-27	+3V3	+3.3V BUS Power supply
H2-28	+3V3	+3.3V BUS Power supply
H2-29	GND	Ground
H2-30	GND	Ground
H2-31	GND	Ground
H2-32	GND	Ground

Table 4. H2 Stack Connector pinout

[1] can be used to control a secondary UHF Module

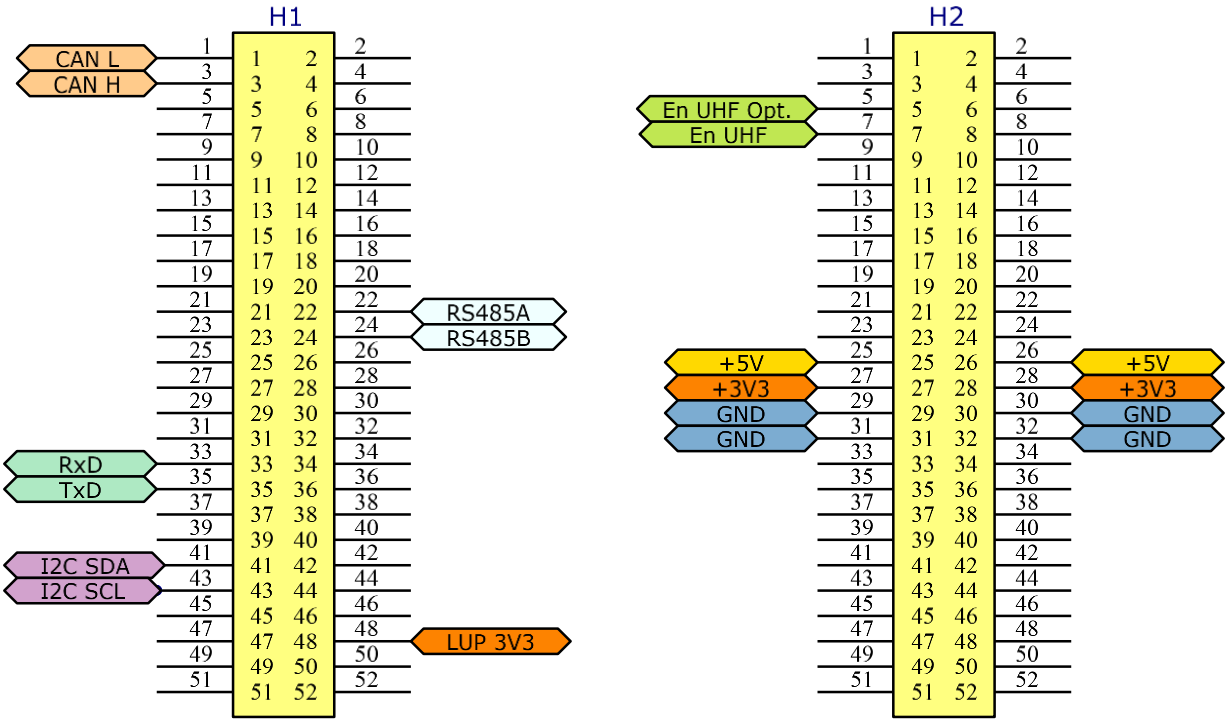


Figure 7: Headers Pinout

8.4 Antenna Release Connector

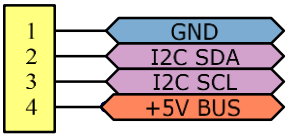


Figure 8: Antenna release connector pinout

Pin	Mnemonic	Description
1	Ground	Ground
2	I2C SDA	I²C data pin
3	I2C SCL	I²C clock pin
4	+5V	+5V BUS

Table 5. Antenna connector pinout

Note: All pins of the antenna release connector are directly connected to the H1&H2 stack connectors.

8.5 Jumper

As shown in Figure 6, there is a 2-pin jumper on the side of the module. When the jumper is mounted, the UHF module starts immediately to transmit Morse code followed by the Audio beacon.

8.6 Mini USB

The Mini-B USB shown in Figure 5 enables the device to be monitored and configured by EnduroSat's software or third-party software. The module can be used as an AX.25 receiver.

9 LED INDICATORS

As shown in Figure 9, the system has LED indicators to give information about its status.

- Blinking of the blue LEDs (left side) – USB communication;
- Blinking of the green LED (right side) – transmitting;
- Blinking of the orange LED (right side) – receiving;

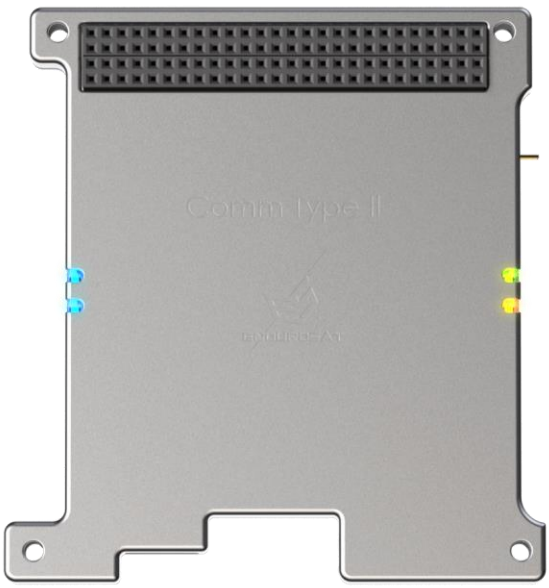


Figure 9: LED indicators

10 BOOTLOADER

Each UHF Transceiver is equipped with a bootloader and an application. The bootloader allows for secure in-flight firmware updates but can also be used for standard (low throughput) command and data handling. It only partially supports the ESTTC protocol and can operate only with some of the RF modes. If the version of the bootloader is below **v.2.0.0**, then all RF modes except **19200 – 9600** and **19200 – 19200** bauds are supported. If one of these two RF modes is chosen, the Status Control Word(SCW) will display the chosen value, but the actual RF mode will be the default one, **9600 – 2400** bauds. If the bootloader version is **v.2.0.0** or above only RF mode **9600 – 2400** bauds is supported. **Again, the SCW will display the chosen RF mode, but the actual RF mode will be the default one**. In either bootloader version case, If the user switches back to application mode, the SCW displayed RF mode will be loaded and used. For example, the user chooses **19200 – 4800** bauds while in application mode and then switches to bootloader mode. The displayed SCW will be **19200 – 4800**, but the actual RF mode will be **9600 – 2400** (assuming bootloader version **v.2.0.0**). If the transceiver switches back to application, the displayed and actual RF modes will be again **19200 – 4800** bauds.

The main bootloader supported features are:

- Secure firmware updates
- AX.25 beacon messages (but no AX.25 automatic decoding of incoming messages)
- Transparent mode
- FRAM read/write

The ESTTC commands supported by the bootloader are marked for each command (see section 13. Command Types and Description) and span the basic configuration of the UHF module.

As of bootloader version **v.2.0.0**, all commands support **CRC32** as described in 12.5 ESTTC Protocol Packet CRC32 Description and Information.

11 UPDATING FIRMWARE

11.1 Overview

The UHF transceiver is designed to allow for secure application firmware updates. An update can be executed via radio or UART while the device is in bootloader or application mode. The update files are provided by EnduroSat and are in a proprietary .SCRM format.

11.2 Firmware Update Procedure Diagram

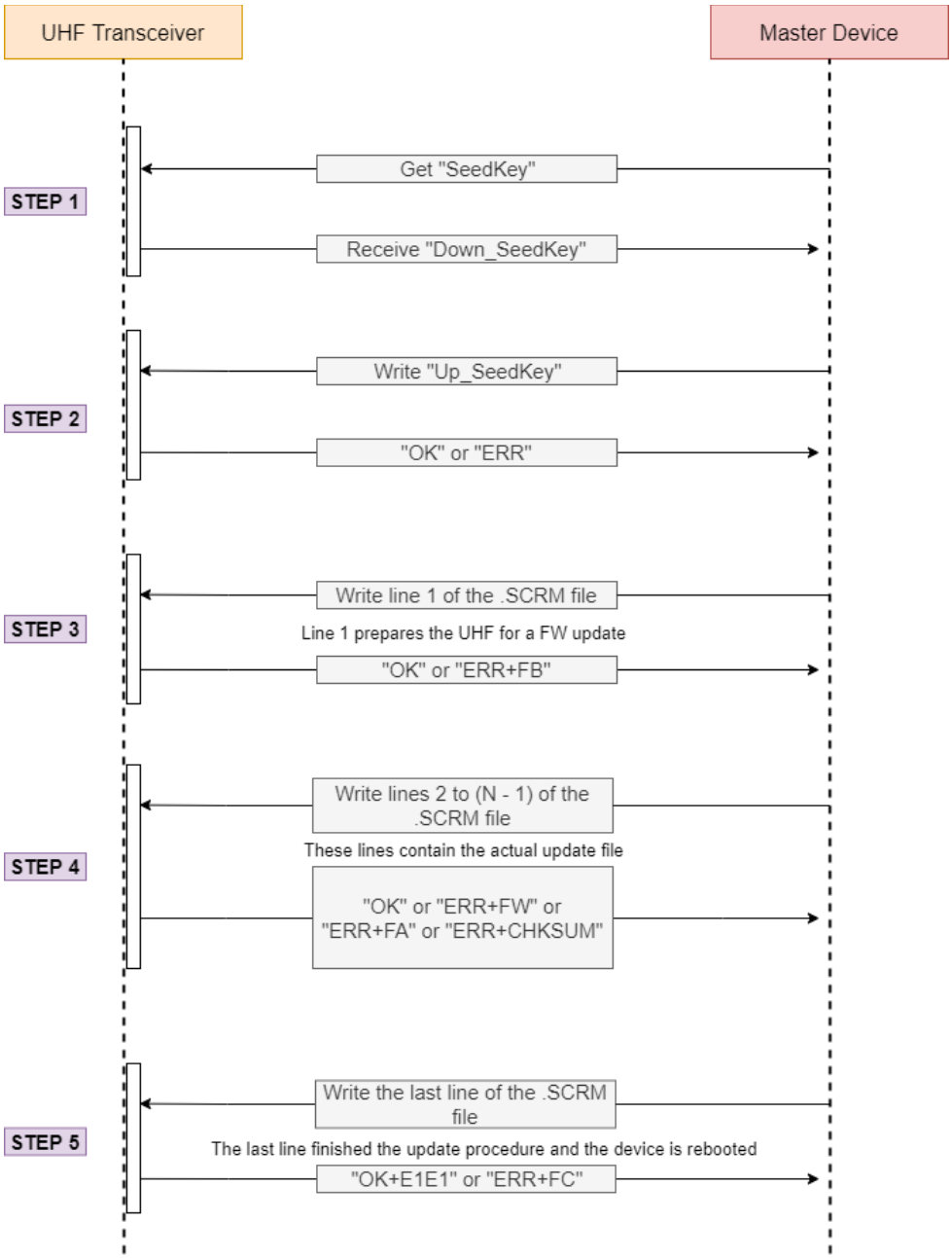


Figure 10: Update Flow of the UHF Transceiver Application Firmware

11.3 Update Step 1

- The **Master** requests the ***SeedKey*** from the **UHF** by sending command **Secure mode** in read format.
- The **UHF** returns the ***Down_SeedKey***, which is: ***SeedKey*** ^ DOWNLINK_XOR

11.4 Update Step 2

- The **Master** XORs the received ***Down_SeedKey*** to get the ***SeedKey***, or:
  SeedKey = ***Down_SeedKey*** ^ DOWNLINK_XOR

- The **Master** writes the *Up_SeedKey* by issuing a write **Secure mode** command, where: *Up_SeedKey* = *SeedKey* ^ UPLINK_XOR
- The **UHF** XOR's the *UP_SeedKey* to get the original *SeedKey* or: *SeedKey* = *UP_SeedKey* ^ UPLINK_XOR
- The UHF replies with ***"OK"*** if the original key was restored or with ***"ERR"*** if it was not

11.5 Update Step 3

- The **Master** sends the first line of the .SCRM file by issuing a **FW update** command
- The **UHF** replies with ***"OK"*** or with ***"ERR+FB"*** if there is a memory error. The Master device should start a new update procedure.

11.6 Update Step 4

- The **Master** sends all but the last line of the .SCRM file using the **FW update** command.
- The **UHF** replies after each received line with ***"OK"*** if reception is successful, with ***"ERR+CHKSUM"*** if the line was received corrupted, with ***"ERR+FW"*** if an error during storing the line occurred. If the **Master** receives one of the above errors, he must re-send the last line until an ***"OK"*** reply is received. If the **Master** receives no reply, he must repeat the line as it is unclear whether it was received or not.

11.7 Update Step 5

- The **Master** sends the last line of the .SCRM file using the **FW update** command.
- The **UHF** replies with ***"OK+F1F1"*** if the update was successful and reboots, loading the new application firmware. If the **UHF** replies with ***"ERR+FC"*** then the update was unsuccessful, and the **Master** device must start the entire update procedure from the start.

11.8 DOWNLINK_XOR and UPLINK_XOR Values

- DOWNLINK_XOR = 0xAB7563CD
- UPLINK_XOR = 0x6ACD3B57

11.9 Other Possible Replies

In the unlikely event that an ***"ERR"*** or ***"ERR+FA"*** message is received during the update procedure please contact EnduroSat for support.

11.10 Updating at Different RF Modes

Firmware updates can be successfully performed under any RF mode supported by the UHF transceiver. However, it should be noted that at the low speed RF modes (such as 1200 – 600 bauds) the reply after each line can take a longer time (due to the low speed) and should be considered. Otherwise, the update will not be possible as both transmit and receiver radios (updater and under-update) will transmit at the same time.

12 PROTOCOL DESCRIPTION

12.1 Features

EnduroSat's Telemetry and Telecommand (ESTTC) protocol highlighted features are:

- Supported interface UART TTL full duplex w/o flow control, default 115200, 8, n, 1
- Module configuration, control and status
- Transmission power adjustable via command (please contact EnduroSat)
- Direct access to radio transceiver properties
- Local and remote (in-flight) secured application firmware update
- ESTTC mode for handling UHF transceiver read/write commands (default mode)
- Transparent mode for passing TX/RX data from the UART to the radio and vice versa
- Beacon transmission in AX.25 UI-frame in disconnected mode
- Beacon transmission in MIDI audio format and Morse code
- All parameters and configurations are stored in non-volatile FRAM memory
- User-accessible non-volatile 256KB FRAM memory

12.2 Radio Packet Structure Description

The radio packet structure utilized in Endurosat's UHF module contains a preamble, a sync word, payload (separated in two data fields) and CRC16 (depicted in Figure 11). These fields are described in detail in Table 6.

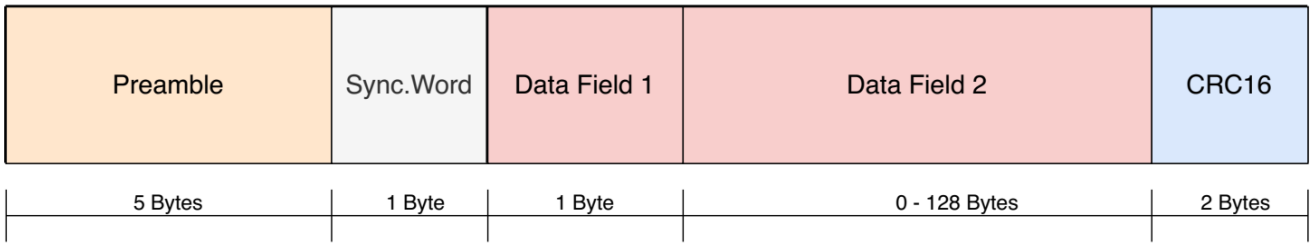


Figure 11: UHF TYPE II Packet Structure (uplink/downlink)

Field	Length [Bytes]	Description
Preamble	5	Value: **0xAAAAAAAAAA**
Sync Word	1	Value: **0x7E**
Data Field 1	1	**Length** of "Data Field 2"
Data Field 2	0-128	**Payload** (variable number of bytes)
CRC	2	**CRC-16/CCITT-FALSE** Polynomial: **$x^{16} + x^{12} + x^5 + 1$**, Seed (initial value): **all 1's** Transmitted **MSB first** (i.e. big-endian) The CRC is applied to Data Field 1 + Data Field 2. For testing purposes, please use https://crccalc.com/ with settings Calc CRC-16 and Algorithm CRC-16/CCITT-FALSE, or an alternative.

Table 6: Packet Fields Description

12.3 AX.25 Protocol UI Frame Structure Description

The AX.25 protocol UI frame used in the UHF transceiver is depicted in Figure 12. Notice that it is fits inside **Data Field 2** and is surrounded by a Preamble and postamble.

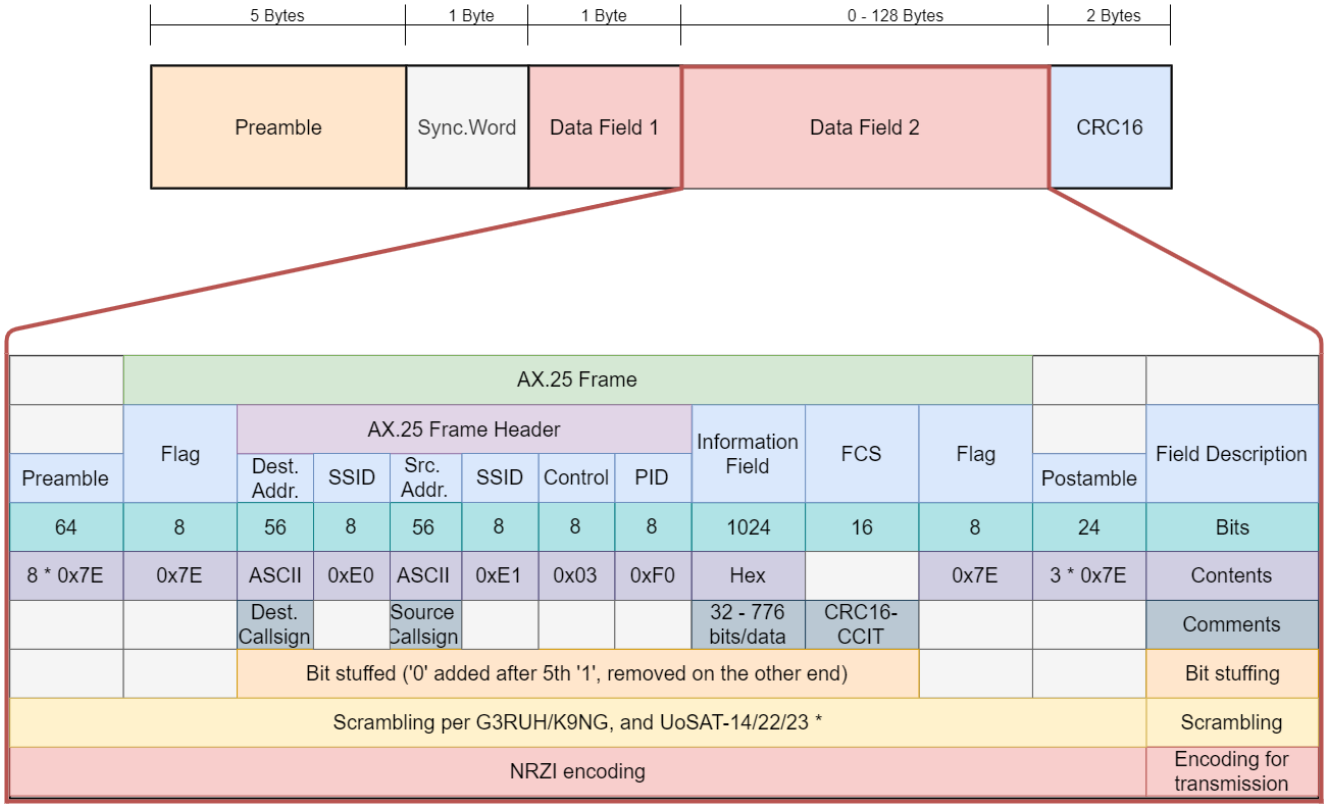


Figure 12: AX.25 protocol UI frame structure

The used scrambling polynomial is $1 + X^{12} + X^{17}$. This means the currently transmitted bit is the EXOR of the current data bit, plus the bits that were transmitted 12 and 17 bits earlier. Likewise, the unscrambling operation simply EXORs the bit received now with those sent 12 and 17 bits earlier. The unscrambler perforce requires 17 bits to synchronise.

12.4 ESTTC Protocol Packet Structure Description

The ESTTC protocol packet utilizes the 128 bytes of payload (Data Field 2 from Figure 11/12) for read/write commands and firmware update when the UHF transceiver is in ESTTC mode. Each command consists of header, command type, address and command code, followed by a different number of command specific data, blank (optional), CRC32 (optional) and ends with <CR>, as depicted in Figure 13 and Table 7. All commands are in ASCII format.

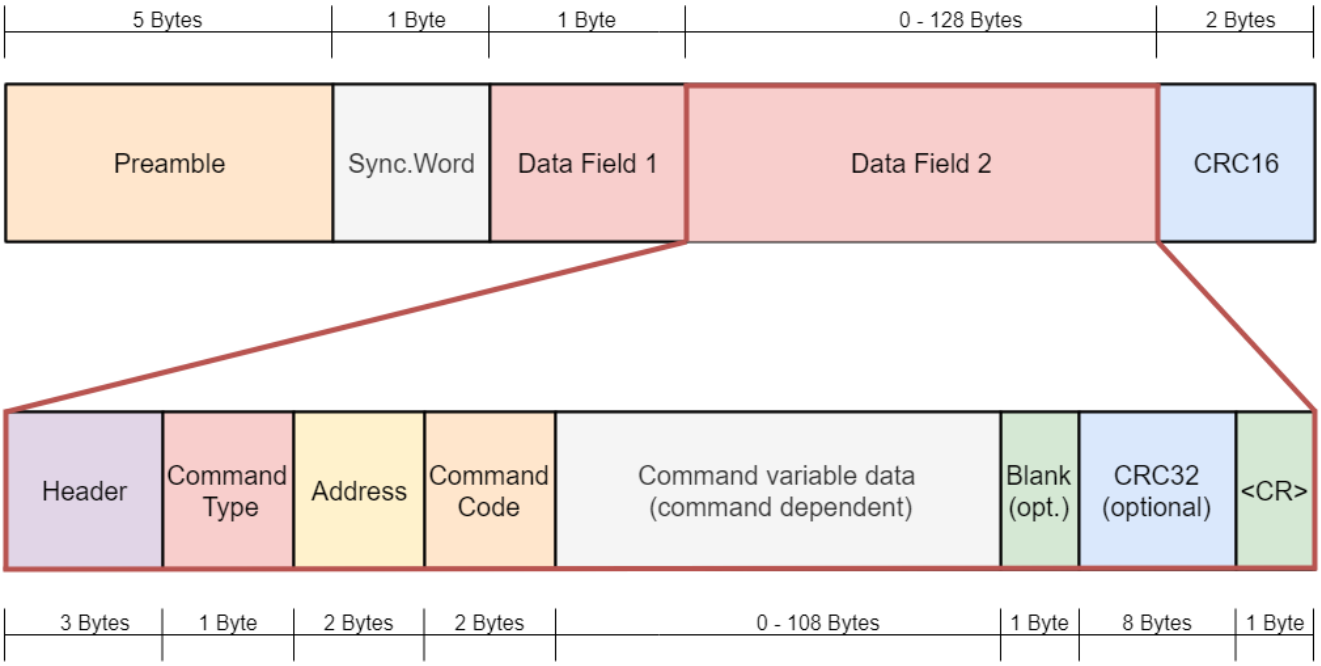


Figure 13: ESTTC Protocol Packet Structure as part of the Radio Packet

Field	Length [Bytes]	Description
Header	3	Value: **ES+**[1]
Command Type	1	Values: **R, W**
Address	2	Values: **22, 23**
Command Code	2	Values: **00 – FF**
Command variable data	0 – 108	Depending on the command type this data can be FRAM addresses, registers, values, etc. Consult command description for further information.
Blank (optional)	1	Value: **0x20** hex or **SPACE** as ASCII
CRC32 (optional)	8	Calculated CRC32 value as ASCII symbols.
<CR>	1	Value: **0x0D** hex or **CR** as ASCII

Table 7: ESTTC Protocol Packet Fields Description

[1] Unless otherwise stated all values are given as ASCII symbols

12.5 ESTTC Protocol Packet CRC32 Description and Information

12.5.1 CRC32 Implementation Description

The different parameters used in EnduroSat's CRC32 implementation are presented in Table 8.

CRC parameter	Value
Polynomial	**0x04C11DB7** or $(x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1)$
Initial value	0xFFFFFFFF
Final XOR	0xFFFFFFFF
Input reflection	TRUE
Output reflection	TRUE

Table 8: CRC32 Implementation Description

Following the above implementation, the calculated value for the standard sequence of ASCII characters **"123456789"** would be: **0xCBF43926**

12.5.2 Sending ESTTC Commands with and without CRC

All ESTTC commands can be sent with or without CRC as of Application/Bootloader FW version 2.00. If commands without CRC are preferred the answer to these commands will be without CRC and the command structure described in **Chapter 13** can be followed by simply dropping the blank space and the following eight ASCII symbols, which represent the calculated CRC.

For example, for command **13.1 – SCW**, the typical read command containing CRC would be (ASCII):

ES+R2200 BD888E1F<CR>

and the typical read command without CRC would be (ASCII):

ES+R2200<CR>

12.5.3 Calculating the CRC with the "xHF and OBC Configurator" Desktop Application

A simple CRC calculator is embedded in the "xHF and OBC Configurator" Application, which is being distributed with each UHF module. The calculator can be located under Tools and is useful if third-party terminal program or script is used for communicating with the transceiver.

NOTE: EnduroSat strongly recommends sending commands with CRC as this ensures proper command reception and subsequent execution.

12.6 Transparent (PIPE) Mode Description

In this mode the UHF transceiver will transmit data received via UART over the radio and vice versa, data received by the radio will be transmitted over UART. This requires no additional commands, but the activation of the transparent mode (see section 13.1 Status Control Word (SCW) for description).

To achieve optimal transmission speeds, the user data must be split into 128 byte chunks before being sent to the UHF Transceiver as the maximum length of the radio packet Data Field 2 is 128 bytes (Figure 11). There must also be a reasonable time gap between two consecutive packets as the UHF

transceiver takes a certain amount of time to send each packet. If more data is sent to the UHF, the data is buffered, however the buffers have a finite size and data may be lost if data transmission rate is too high. To prevent this case, the user can consult Table 9, which relates RF modes and baudrates to the minimum required time between two consecutive packets of size 128 bytes for reliable and optimal communication. Of course, if the packets are shorter, then the time depicted in Table 9 will be different.

Baudrate [bps] / RF Mode [bps – Hz]	115200	19200	9600
1200 – 600	> 920[ms]	> 920 [ms]	> 920 [ms]
2400 – 600	> 460[ms]	> 460 [ms]	> 460 [ms]
4800 – 1200	> 240 [ms]	> 240 [ms]	> 240 [ms]
9600 – 2400	> 120 [ms]	> 120 [ms]	> 3[1] [ms]
9600 – 4800	> 120 [ms]	> 120 [ms]	> 3[1] [ms]
19200 – 4800	> 60 [ms]	> 3[1] [ms]	> 3[1] [ms]
19200 – 9600	> 60 [ms]	> 3[1] [ms]	> 3[1] [ms]
19200 – 19200	> 60 [ms]	> 3[1] [ms]	> 3[1] [ms]

Table 9: RF Mode and Baudrate (relation to minimum time between two consecutive packets).

The device stays in transparent mode for as long as data is received from the radio or the UART. If no data is received for a certain amount of time, which is user-configurable (see section 13.7 Transparent (PIPE) Mode Timeout Period Configuration), the UHF transceiver will exit transparent mode and enter normal mode, where it listens for valid ESTTC commands or AX.25 UI-frames.

[1]See description in text.

12.7 Automatic AX.25 UI-Frame Decoding

When the device is not in Transparent mode, it listens actively for valid ESTTC commands or AX.25 UI-frames following the implementation described in 12.3 AX.25 Protocol UI Frame Structure Description. If a valid frame is detected, it is decoded, and the message transmitted over the UART interface.

12.8 Communicating Via I²C

- The UHF Transceiver is a slave device, but can switch to master mode if commanded so by a UART or Radio command
- 7-bit address in the range **0x22** ÷ **0x23**
- Clock speed is 400kHz
- Pull-up 10k Ohm resistors are mounted on the UHF I²C interface

The ESTTC commands format is the same when the command is sent via I²C, UART or Radio. However, there are some commands that are not available via I²C (see Chapter 13). To send a command to the UHF and to read the result, the I²C Master must first initiate a write and then a read following the standard I²C flow (Figure 14):

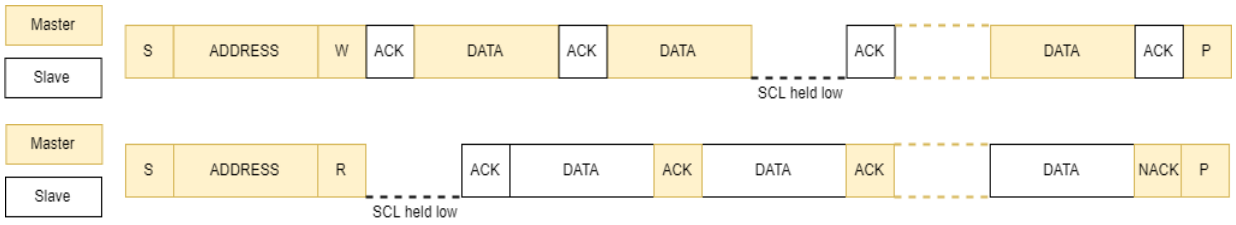


Figure 14: Slave Receiver (top) and Slave Transmitter (bottom) Modes on the I2C Bus.

To read the entire command response the master is expected to generate the necessary number of clocks. If it fails to do so, the response is lost prior to the next I^2C command. If the master reads more data than the reply message length, the UHF replies with **0x00**.

After each read request there is some delay before the UHF is ready to send a response. During this time the SCL line is held low. **This delay may be as large as 150ms**, and during this time the SCL line is held low by the UHF.

12.9 Common Answers Intrinsic to All ESTTC Commands

Answer	Meaning
E_CRC_ERR	Command was sent with wrong CRC attached, command is received with an error or received command CRC has an error
E_CRC_ERR_LEN	Command length does not comply with ESTTC protocol commands description with or without CRC
I2C_NACK	Typically received if the UHF antenna is not attached and command F3 is executed

Table 10: Common command answers

13 COMMAND TYPES AND DESCRIPTION

13.1 Status Control Word (SCW) [Application / Bootloader]

SCW format:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		UartBaud		Reset	RfMode			Echo	BCN	Pipe	Boot	CTS	SEC	FRAM	RFTS
0	0	r/w-3		r/w-0	r/w-3			r/w-0	r/w-0	r/w-0	r-X	r-X	r-X	r-X	r-X

Bit field	Meaning
[15]	**Reserved:** For future use (default value is 0)
[14]	**HFXT:** High frequency oscillator status, 0 – oscillator OK, 1 – oscillator error
[13, 12]	**UartBaud:** Speed of the UART interface. Possible values 00-9600, 01 - reserved, 10-19200, 11-115200 (default). **NOTE:** 230.400 kbps UART interface speed supported (optional)
[11]	**Reset:** Write 1 to reset the device, 0 – no effect (default)
[10, 9 ,8]	**RFMode:** see Table 9
[7]	**Echo**: Local UART echo of the transmitted symbols over the radio when a valid ESTTC command is received via radio; 1 – Echo on, 0 – Echo off (default)
[6]	**BCN**: Beacon message control; 1 – enabled, 0 – disabled (default)
[5]	**Pipe**: Transparent mode communication control; 1 – Pipe mode on, 0 – Pipe mode off (default)
[4]	**Boot**: Indicates whether the device is in bootloader or application mode 1 – Bootloader, 0 – Application
[3]	**CTS**: Reserved for future use (default value is 0)
[2]	**SEC**: Reserved (default value is 0)
[1]	**FRAM**: Indicates whether FRAM is initialized correctly after reset; 1 – OK, 0 – FRAM Error
[0]	**RFTS**: Indicates whether radio transceiver is initialized correctly after reset; 1 – OK, 0 – Radio Error

To switch back and forth between bootloader and application the user will have to set both **Bit 4** and **Bit 11** bearing in mind their effect as described in the above table. If only **Bit 4** is set/cleared the device will stay in its current mode (bootloader or application).

RF Mode #	Modulation	Data rate, [bps]	Fdev, [Hz]	ModInd
0 [000]	2GFSK	1200	600	1
1 [001]	2GFSK	2400	600	0.5
2 [010]	2GFSK	4800	1200	0.5
3 [011] (default)	**2GFSK**	**9600**	**2400**	**0.5**
4 [100]	2GFSK	9600	4800	1
5 [101]	2GFSK	19200	4800	0.5
6 [110]	2GFSK	19200	9600	1
7 [111]	2GFSK	19200	19200	2

Table 11: Available RF modes

NOTE: For other modulations schemes please contact EnduroSat for support. Up to 16 different RF modes (modulation, data rate, frequency deviation) can be preconfigured with one more bit (optional).

Status Control Word	
Write	Answer
ES+W[AA]00[WWWW][B][C..C]<CR>	1) OK+[WWWW][B][C..C]<CR> : successful configuration 2) +ESTTC[B][C..C]<CR>: response at exit of PIPE mode (not valid if command is sent via I2C)
Read	Answer
ES+R[AA]00[B][C..C]<CR>	OK+[RR][AA][BB][WWWW][B][C..C]<CR>

[AA] is the device address in HEX format;

[WWWW] is a 16-bit value in HEX format use to modify the SCW bit fields in "Write" command and current SCW content as a result of "Write" and "Read" command execution;

[RR] is the last received signal strength indicator (RSSI);

[BB] is the reset counter. It counts times device has undergone reset (power down, reset via command, new SW via bootloader);

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R2200 BD888E1F<CR>
- ES+W22003323 589B0F83<CR>

13.2 Radio Frequency Configuration [Application / Bootloader]

Radio frequency	
Write	Answer
ES+W[AA]01[FFFFFF][NN][B][C..C]<CR>	OK[B][C..C]<CR> : successful configuration ERR[B][C..C]<CR>: configuration failed
Read	Answer
ES+R[AA]01[B][C..C]<CR>	OK+[RR][FFFFFF][NN][B][C..C]<CR>

[AA] is the device address in HEX format;

[FFFFFF] is the fractional part and [NN] is the integer divider of the radio PLL synthesizer in HEX format;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The values can be calculated according to SILICON LABS Si4463/4 revB1B Command/Property API Documentation FREQ_CONTROL Grp 0x40 or Si4463 datasheet, Ch.5.3. Synthesizer using the formula: $\mathrm{RF_{channel}, [Hz]} = (\mathrm{F_{cint}} + \mathrm{F_{cfrac}}/2^{19}) \times \mathrm{NPRESC} \times \mathrm{F_{req\_xo}}/\mathrm{outdiv},$

where $\mathrm{NPRESC} = 2$, $\mathrm{F_{req\_xo}} = 26000000, \mathrm{outdiv} = 8$.

The entire $\mathrm{F_{cfrac}}$ word is 20-bits in length, but the most significant bit should always be set to 1, and thus the term $\mathrm{F_{cfrac}}/2^{19}$ will always be between 1 and 2 in value. As a result, the integer term $\mathrm{F_{cint}}$ should be reduced by 1. The default value is 76620F41 = 435000000 Hz.

Example: A total desired divide ratio of $(\mathrm{F_{cint}} + \mathrm{F_{cfrac}}/2^{19}) = 60.135$ should be implemented as $\mathrm{F_{cint}} = 59$, $\mathrm{F_{cfrac}}/2^{19} = 1.135$.

Example commands:

- ES+R2200 BD888E1F<CR>
- ES+W220150E90942 36F6ADAB<CR>

13.3 Read Uptime [Application / Bootloader]

Uptime	
Read	Answer
ES+R[AA]02[B][C..C]<CR>	OK+[RR][PPPPPPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPPPPPP] is the uptime value in seconds in HEX format;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R2202 5386EF33<CR>

13.4 Read Number of Transmitted Packets [Application / Bootloader]

Transmitted packet	
Read	Answer
ES+R[AA]03[B][C..C]<CR>	OK+[RR][PPPPPPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPPPPPP] is the value in HEX format;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R2203 2481DFA5<CR>

13.5 Read Number of Received Packets [Application / Bootloader]

Received packet	
Read	Answer
ES+R[AA]04[B][C..C]<CR>	OK+[RR][PPPPPPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPPPPPP] is the value in HEX format;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R2204 BAE54A06 <CR>

13.6 Read Number of Received Packets with CRC Error [Application / Bootloader]

Received packet	
Read	Answer
ES+R[AA]05[B][C..C]<CR>	OK+[RR][PPPPPPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPPPPPP] is the value in HEX format;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The read value indicates the number of packets that were received, but the CRC of the radio packet was found to be wrong. This value is not to be mixed with the CRC32 that is part of the ESTTC protocol packet structure.

Example commands:

- ES+R2205 CDE27A90<CR>

13.7 Transparent (PIPE) Mode Timeout Period Configuration [Application / Bootloader]

Transparent mode	
Write	Answer
ES+W[AA]06000000[TT][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR>: unsuccessful configuration
Read	Answer
ES+R[AA]06[B][C..C]<CR>	OK+[RR]000000[TT][B][C..C]<CR>

[AA] is the device address in HEX format;

[TT] is the period in HEX format in seconds;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

This timeout defines the time the device stays in transparent mode after the last transfer of incoming UART data to the radio or vice-versa. If there is no incoming data for longer than the timeout period, then the module returns to ESTTC (default) mode. Minimum value is 0x01 = 1 second. Maximum value is 0xFF=255 seconds. The default value is 10 seconds.

Example commands:

- ES+R2206 54EB2B2A<CR>
- ES+W220600000060 9F610824<CR>

13.8 Beacon Message Transmission Period Configuration [Application / Bootloader]

Beacon transmission	
Write	Answer
ES+W[AA]070000[TTTT][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR>: try to configure zero period.
Read	Answer
ES+R[AA]07[B][C..C]<CR>	OK+[RR]0000[TTTT][B][C..C]<CR>

[AA] is the device address in HEX format;

[TTTT] is the period in HEX format in seconds;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

When beacon transmission is enabled, the device is transmitting the beacon message at a preconfigured 2-byte period of 1-65535 seconds. The default value is 60 seconds.

Example commands:

- ES+R2207 23EC1BBC<CR>
- ES+W220700000060 881A1C67<CR>

13.9 Audio Beacon Period Between the Transmissions [Application / Bootloader]

Audio beacon period	
Write	Answer
ES+W[AA]080000[TTTT][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR>: [TTTT] = [1..30]
Read	Answer
ES+R[AA]08[B][C..C]<CR>	OK+[RR]0000[TTTT][B][C..C]<CR>

[AA] is the device address in HEX format;

[TTTT] is the period in HEX format in seconds;

[RR] is the last RSSI;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

To enable the transmission [TTTT] > 30, to disable the transmission [TTTT] = 0. Any other values are incorrect and lead to error. **By default, the audio beacon is disabled.**

NOTE: Period at least 30 seconds between the consecutive transmissions is guaranteed to reduce the power consumption and to enable the temperature dissipation.

Example commands:

- ES+R2208 B353062D<CR>
- ES+W220800000060 57A3D3B6<CR>

13.10 Restore Default Values [Application / Bootloader]

Default values	
Write	Answer
ES+W[AA]09[B][C..C]<CR>	OK[B][C..C]<CR>: successful configuration

[AA] is the device address in HEX format;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

By using this command, the user can restore initial values for:

- Destination/Source/Morse code call sign;
- Audio beacon period and message;
- Text beacon period
- Pipe timeout period.

Example commands:

- ES+W2209 0CB4B9CB<CR>

13.11 Internal Temperature Sensor Measurement Value [Application]

Temperature measurement	
Read	Answer
ES+R[AA]0A[B][C..C]<CR>	OK[B][TTTT][B][C..C]<CR>

[AA] is the device address in HEX format;

[TTTT] is the signed measured temperature in DEC format with 0.1 degrees Celsius accuracy;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R220A 9A8ACFB5<CR>

13.12 Generic Write And/Or Read From an I2C Device [Application]

Write and/or read from an I2C device	
Write	Answer
ES+W[AA]F1[C][XX][WW][D…D][RR][B][C..C]<CR>	1) OK+[RD..RD][B][C..C]<CR>: data read successfully 2) OK+SENT[B][C..C]<CR>: successful write on the I2C bus (no read requested) 3) ERR+I2C_NOINIT[B][C..C]<CR>: I2C not successfully initialized 4) ERR+WRT_LEN[B][C..C]<CR>: write length exceeded 5) ERR+READ_LEN[B][C..C]<CR>: read length exceeded 6) ERR+NA[B][C..C]<CR>: cmd not available via I2C 6) Consult Table 10

[AA] is the device address in HEX format;

[C] indicates the way the data is sent. There are three options:

- 'S' – sends the data written in [D…D] and then attaches a blank space and the CRC32 checksum (calculating according to the implementation described in ESTTC protocol packet CRC32 description and information)
- 'C' – sends the data and attaches a <CR> at the end as the termination symbol
- 'D' – sends the data as is with nothing additional

[XX] is the I2C slave device address in HEX format;

[WW] is the length of the data in HEX format. This data is limited to 32 bytes (64 ASCII symbols);

[D…D] is the data to be placed on the I2C bus in HEX format;

[RR] is the number of bytes to read from the slave device in HEX format;

[RD…RD] is the read data in HEX format. This read length is limited to 32 bytes (64 ASCII symbols);

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

This is a generic command allowing access via the UHF to all devices connected to the I2C bus. When this command is sent, the UHF transceiver switches to master mode. The above command can be sent via UART or Radio and supports write only, write and read and read only modes.

To send a write only command, the read length should be set to 0x00. Likewise, to send a read only command, the write length should be set to 0x00.

NOTE: This command is not available via I2C

Example commands:

- ES+W22F1S230845532b523232303316 71E32F28<CR>

13.13 UHF Antenna Release Configuration [Application]

Ant. release config.	
Write	Answer
ES+W[AA]F2[FFFF][B][C..C]<CR>	1) OK+[FFFF][B][C..C]<CR>: successful configuration 2) ERR<CR>: invalid command value
Read	Answer
ES+R[AA]F2[B][C..C]<CR>	OK+[PPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPP] indicates the current UHF antenna release configuration. The first byte holds the enable automatic antenna deployment flag and the enable robust deployment flag. By default, these flags are not set (deployment is disabled). The second byte specifies the time in minutes (in HEX) after power-up after which the deployment sequence should be executed.

[FFFF]:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			First	Reserved			EN	Time							
0	0	0	r/w-0	0	0	0	r/w-0	r/w-0	r/w-0	r/w-0	r/w-0	r/w-0	r/w-0	r/w-0	r/w-0

The upper byte can be used to enable/disable the automatic release sequence by setting/clearing the EN bit. The First bit indicates if the robust automatic release sequence is to be executed. If set, the release logic will check (after power-up/reset and first antenna connection) if any rods are opened. If such are found, the logic will consider that some issue has occurred and will try to deploy them first. If this algorithm is successfully executed, the First flag will be cleared automatically, otherwise it will stay set and the UHF transceiver will try to deploy all open rods again at the next power-up/reset cycle.

The lower byte specifies the time in minutes after device power-up when antenna deployment should happen. This value is in HEX and can be anything between 0x0A and 0xFF including.

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Bear in mind that if a release command is given to the device and the device has been powered up longer than the time indicated by the lower byte, the antenna release sequence will begin immediately! In case the command is sent via I2C the user may not get an appropriate reply as the UHF Transceiver will switch to I2C master mode.

To prepare the UHF to release the antenna after the next power-up, power-up the UHF module, set the appropriate command values and power-off the device. At the next power-up the UHF will wait until the set time has elapsed and will release the antenna. If the device is not powered-off after the release command is set, the release will happen as soon as the internal UHF timer reaches the set time.

If connection to the antenna was not established or if all rods were not deployed, the **EN** flag will not be cleared and at next power-up/reset the release logic will be restarted (release after the set time). Only once the antenna has returned a status that all rods are opened the **EN** flag will be cleared.

The release algorithm embedded in the UHF transceiver will analyse the antenna rods statuses and will try to release all of them by turning on the different algorithms intrinsic to the UHF antenna. For further information on the antenna please consult EnduroSat's UHF antenna documentation.

Example commands:

- ES+R22F2 2AE33143<CR>
- ES+W22F201FF 852DF0FE<CR>

13.14 UHF Antenna Read/Write [Application]

UHF antenna	
Write	**Answer**
ES+W[AA]F3[CC][B][C..C]<CR>	1) OK[B][C..C]<CR>: command accepted 2) ERR[B][C..C]<CR>: invalid command value
Read	
ES+R[AA]F3[B][C..C]<CR>	1) OK+[PPPPPPPP] [B][C..C]<CR> 2) ERR+I2C_NOINIT[B][C..C]<CR>: I2C not init. 3) ERR+NA[B][C..C]<CR>: cmd not available via I2C 4) Consult Table 10

[AA] is the device address in HEX format;

[PPPPPPPP] is the current antenna status;

[CC] is the antenna deployment algorithm command;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Consult EnduroSat's UHF Antenna documentation or contact EnduroSat for a list of commands and current antenna status replies.

NOTE: This command is not available via I2C

Example commands:

- ES+R22F3 5DE401D5<CR>
- ES+W22F31F C14B8267<CR>

13.15 Low Power Mode [Application]

Low power mode	
Write	Answer
ES+W[AA]F4[B][C..C]<CR>	1) OK[B][C..C]<CR>: entered low power mode 2) ERR[B][C..C]<CR>: already in low power mode
Read	Answer
ES+R[AA]F4[B][C..C]<CR>	OK+[KK][B][C..C]<CR>

[AA] is the device address in HEX format;

[KK] indicates low power mode status. When KK = 01 the device is in low power mode and when KK = 00 the device is in normal mode. **By default, the device is in normal mode**;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

This command places the device in low power (sleep) mode. **The command can be sent via UART or I2C only and care must be taken to implement the necessary wake-up procedure as otherwise the transceiver will remain in sleep mode indefinitely!** Upon entering low power mode, the current UHF configuration is saved and will be used once the device is brought back from sleep. To exit low power mode any ESTTC command, except the current one, can be used but it must be issued via UART or I2C as radio communication is off.

Power consumption of the UHF transceiver in sleep mode is **8.37 mA** and **24.61 mA** in normal mode.

Example commands:

- ES+R22F4 C3809476<CR>
- ES+W22F4 0B601B06<CR>

13.16 Destination Call Sign [Application / Bootloader]

Destination call sign	
Write	Answer
ES+W[AA]F5[DDDDDD][B][C..C]<CR>	OK[B][C..C]<CR>: successful configuration
Read	Answer
ES+R[AA]F5[B][C..C]<CR>	OK+DDDDDD[B][C..C]<CR>

[AA] is the device address in HEX format;

[DDDDDD] is the call sign in ASCII format (the first 6 symbols are used only);

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The Destination Call Sign is used in the AX.25 protocol format. By default, the content is = **"CQ "**.

Example commands:

- ES+R22F5 B487A4E0<CR>

- ES+W22F5ABCDEF EB1BA8B1<CR>

13.17 Source Call Sign [Application / Bootloader]

Source call sign	
Write	Answer
ES+W[AA]F6[DDDDDD][B][C..C]<CR>	OK[B][C..C]<CR>: successful configuration
Read	Answer
ES+R[AA]F6[B][C..C]<CR>	OK+DDDDDD[B][C..C]<CR>

[AA] is the device address in HEX format;

[DDDDDD] is the call sign in ASCII format (the first 6 symbols are used only);

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The Source Call Sign is used in the AX.25 protocol format. By default, the content is = **"XX0UHF"**.

Example commands:

- ES+R22F6 2D8EF55A<CR>
- ES+W22F6FEDCBA 3C6B73DC<CR>

13.18 Morse Code Call Sign [Application]

Morse code call sign	
Write	Answer
ES+W[AA]F7[LL][C…C][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR>: incorrect string length
Write	Answer
ES+R[AA]F7[B][C..C]<CR>	OK+[LL][C…C][B][C..C]<CR>

[AA] is the device address in HEX format;

[LL] is the number of symbols in DEC format (<37);

[C…C] is the symbol string, which consist of DASH (HEX 0x2D, ASCII -), DOT (HEX 0x2E, ASCII) and SPACE (HEX 0x20, ASCII). Any different symbol will break further string parsing;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

By default, the Morse Code is = "-..- -..- ----- ..- …. ..-. " X-ray X-ray Zero Uniform Hotel Foxtrot.

Example commands:

- ES+R22F7 5A89C5CC<CR>
- ES+W22F705...-- C2FFF8E7<CR>

13.19 MIDI Audio Beacon [Application]

MIDI audio beacon	
Write	Answer
ES+W[AA]F8[LL][NND]…[NND][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR+MIDI[B][C..C]<CR>: incorrect midi number 3) ERR[B][C..C]<CR>: incorrect string length
Read	Answer
ES+R[AA]F8[B][C..C]<CR>	OK+[LL][NND]…[NND][B][C..C]<CR>

[AA] is the device address in HEX format;

[LL] is the number of notes in DEC format (<37);

[NND] is a note in ASCII format as NN is the number of the MIDI (Musical Instrument Digital Interface) piano key for that note/octave and D is its duration. The allowed MIDI numbers are [12 – 99]. The duration code table is:

Symbol	Duration
'w'	Whole note, no gap
'W'	Whole note
'h'	$^1/_2$ Note, no gap
'H'	$^1/_2$ Note
'D'	$^1/_4$ Note + $^1/_8$ Note ('D' = 'q' + 'o')
'q'	$^1/_4$ Note, no gap
'Q'	$^1/_4$ Note
'o'	$^1/_8$ Note, no gap
'O'	$^1/_8$ Note
'-'	$^1/_8$ + $^1/_{16}$ Dash delay
'x'	$^1/_{16}$ Note, no gap
'X'	$^1/_{16}$ Note
'.'	Dot delay
Any other symbol	$^1/_{32}$ Note

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

NOTE: Default = "Ode to Joy": 1571Q71Q72Q74Q74Q72Q71Q69Q67Q67Q69Q71Q71D69O69H

NOTE: No answer will be received if 37 or more notes are sent via radio frequency channel.

NOTE: In Bootloader mode Morse code and MIDI audio beacon are not supported.

Example commands:

- ES+R22F8 CA36D85D<CR>
- ES+W22F80371Q71Q71Q 78DEE43E<CR>

13.20 Read Software Version Build [Application / Bootloader]

Software version	
Read	Answer
ES+R[AA]F9[B][C..C]<CR>	OK+[V.vv]<date, time>[B][C..C]<CR>

[AA] is the device address in HEX format;

In [V.vv] the first digit is related to the version and the second/third ones to the increment;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R22F9 BD31E8CB<CR>

13.21 Read Device Payload Size [Application / Bootloader]

Payload size	
Read	Answer
ES+R[AA]FA[B][C..C]<CR>	OK+[PPPP][B][C..C]<CR>

[AA] is the device address in HEX format;

[PPPP] is the payload size in HEX format (refer to Data Field 2, Figure 11);

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R22FA E3EF11C5<CR>

13.22 Beacon Message Content Configuration [Application / Bootloader]

Beacon content	
Write	Answer
ES+W[AA]FB[LL][B…B][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR> - [LL] is greater than the beacon size, which is 128 bytes.
Read	Answer

ES+R[AA]FB[B][C..C]<CR>	1) OK+[BB][HH…HH][B][C..C]<CR> 2) ERR+REMOTE<CR>

[AA] is the device address in HEX format;

[LL] is the number of bytes in [B…B] in HEX format, where [LL] < 0x62;

[B…B] is the beacon message variable size content in ASCII format;

[BB] is beacon message size in HEX;

[HH … HH] is AX.25 encoded message content in HEX format;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The ESTTC protocol is in ASCII format and is using <CR> (0D-HEX/13-DEC) as the termination symbol. It is therefore necessary for the users to avoid the byte value 13-DEC in "write" commands, due to possible command parser errors. Any other byte values are generally allowed, but ASCII readable symbols are recommended for an easy decoding by any terminal application.

NOTE: The default content of the beacon message is = "Hello, world!"

NOTE: The read message size is always greater than the write message size due to AX.25 encoding.

NOTE: In case of a remote read request, the beacon content cannot be sent via the radio due to its unknown length.

NOTE: After reset the beacon message content defaults to "Hello, world!".

Example commands:

- ES+R22FB 7AE6407F<CR>
- ES+W22FB0BHello Earth E361E6C8<CR>

13.23 Device Address Configuration [Application / Bootloader]

Address	
Write	Answer
ES+W[CurrAddr]FC[NewAddr][B][C..C]<CR>	OK+[NewAddr][B][C..C]<CR>:successful configuration

[CurrAddr] is the address hex value and can be 0x22 or 0x23;

[NewAddr] is the address hex value and can be 0x22 or 0x23;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

The 1-byte address is used to determine the recipient of the command via UART according to the ESTTC protocol as well as the slave 7-bit address via I2C interface. It is used when two modules are integrated in one system. The default value is 0x22.

NOTE: If this command is sent via I2C there will be no reply as the device will reconfigure the internal I2C periphery to align with the new address

Example commands:

- ES+W22FC23 9EF83C47<CR>

13.24 FRAM Memory Read/Write [Application / Bootloader]

FRAM memory	
Write	Answer
ES+W[AA]FD[AAAAAAAA][D0…D15][B][C..C]<CR>	1) OK[B][C..C]<CR>: data is stored in FRAM successfully. 2) ERR[B][C..C]<CR>: fail writing to FRAM.
Read	Answer
ES+R[AA]FD[AAAAAAAA][B][C..C]<CR>	1) OK+[D0…D15][B][C..C]<CR>: data read from FRAM in HEX format. 2) ERR[B][C..C]<CR>: fail reading from FRAM location.

[AA] is the device address in HEX format;

[AAAAAAAA] is FRAM 32-bit address location in HEX;

[D0…D15] is the 16 bytes (HEX) of data to be written. The FRAM memory has flat structure and linear addressing space from 0x00 to 0x3FFF0.

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

NOTE: FRAM addresses 0x8000-0x83A4, 0x83FE-0x24000 are reserved and cannot be used by user.

Example commands:

- ES+R22FD00024001 73B3F69C<CR>
- ES+W22FD00024001ABABABABDDDDDDDDFFFFFFFFAAAAAAAA 149E3006<CR>

13.25 Radio Transceiver Property Configuration [Application / Bootloader]

Radio property	
Write	Answer
ES+W[AA]FE[GG][NN][SS][D…D][B][C..C]<CR>	1) OK[B][C..C]<CR>: successful configuration 2) ERR[B][C..C]<CR>: fail to set property
Read	Answer
ES+R[AA]FE[GG][NN][SS][B][C..C]<CR>	1) OK+[Y…Y][B][C..C]<CR> 2) ERR[B][C..C]<CR>: fail to read property

[AA] is the device address in HEX format;

[GG] is the property group in HEX;

[NN] is the number of bytes in HEX;

[SS] is the start offset of the property in HEX;

[D…D] is the variable size data to be written in HEX format;

[Y…Y] is the property content in HEX format;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

NOTE: Do not use this command without EnduroSat's approval!

Example commands:

- ES+R22FE220101 B5D9879D<CR>
- ES+W22FE2201017F E47DB270<CR>

13.26 Secure Mode [Application / Bootloader]

Secure mode	
Write	Answer
ES+W[AA]FF[B][C..C]<CR>	1) OK[B][C..C]<CR>: successfully entered secure mode 2) ERR[B][C..C]<CR>: authentication failed
Read	Answer
ES+R[AA]FF[B][C..C]<CR>	1) OK+[K…K][B][C..C]<CR>

[AA] is the device address in HEX format;

[K…K] are the four bytes comprising the secure key in HEX format;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+R22FF 7D8B8466<CR>

13.27 Firmware Update [Application / Bootloader]

Firmware update	
Update command format	Answer
ES+D[AA][D…D][B][C..C]<CR>	Consult Chapter 11. Updating Firmware for information on the possible answers

[AA] is the device address in HEX format;

[D…D] is the variable length data which comprises a single line of the .SCRM file;

[B] is the blank space ASCII character;

[C..C] are the 8 ASCII characters representing the calculated CRC32;

Example commands:

- ES+D22S3220D119E261D2309A92D9DAA6677494576CD02556F794EE76B49F175EDE2B615858DAD F2AC3BA3<CR>

14 MECHANICAL DRAWING

The following pictures show the external dimensions of the UHF Transceiver Type II module.

STEP files can be provided upon request. All dimensions are in mm.

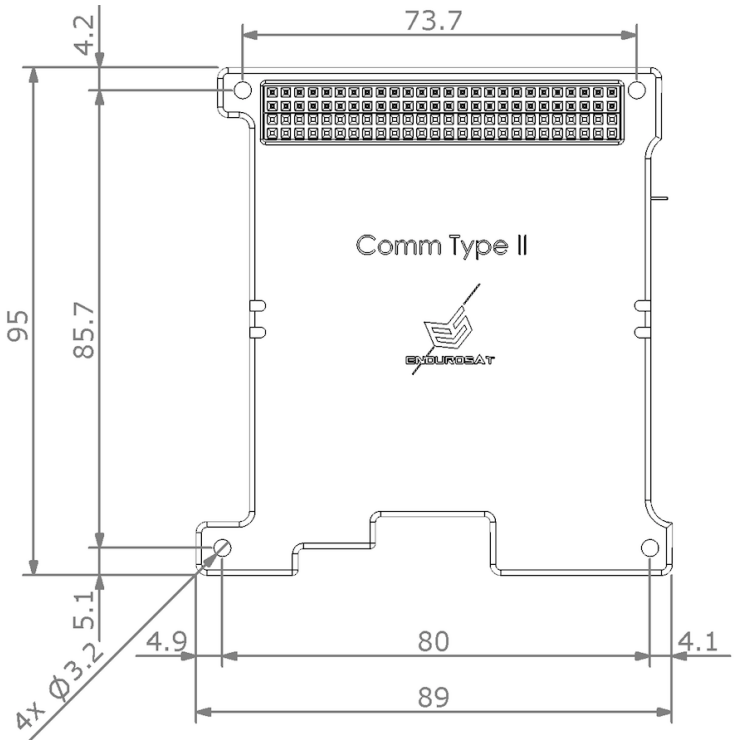


Figure 14: UHF transceiver Type II - Top View

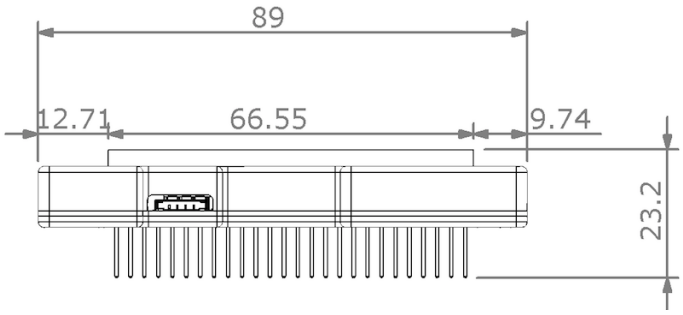


Figure 15: UHF Transceiver Type II - Side View

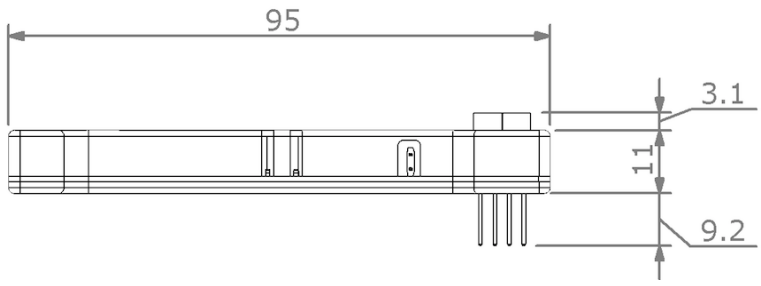


Figure 16: UHF Transceiver Type II - Side View

15 ENVIRONMENTAL AND MECHANICAL TESTING

A full campaign of tests at qualification level was performed on the qualification engineering model. Qualification tests level and duration follow the GEVS standard: GSFC-STD-7000A. Test performed:

- Random Vibration
- Sinusoidal Vibration
- Pyroshock Test
- Thermal Cycling
- Thermal Vacuum
- Total Ionizing Dose

16 MATERIALS AND PROCESSES

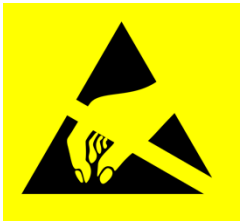
- Surface mount technology component placement
- Standard: IPC-A-610E Class 3
- Aluminum 6061 T651 box
- Visually inspected
- Functionally verified

17 HANDLING AND STORAGE

Particular attention shall be paid to the avoidance of damage of the communication module during handling, storage and preservation. The handling of the communication module should be performed in compliance with the following instructions:

- Handle using PVC, latex, cotton (lint free) or nylon gloves.
- The environment where the device will be handled shall meet the requirements for a class environment 100,000, free of contaminants such dust, oil, grease, fumes and smoke from any source.
- Store in such a manner as to preclude stress and prevent damage.
- To prevent the deterioration of the module, then the module shall be stored in a controlled environment (i.e. the temperature and humidity levels shall be maintained within the proper ranges):
    - Ideal storage temperature range: 15ºC to 27ºC
    - Ideal storage humidity range: 30% to 60% relative humidity (RH)

18 WARNINGS



This product uses semiconductors that can be damaged by electrostatic discharge (ESD). Observe precautions for handling.

Sensitive Electronic device. Do not ship or store near strong electrostatic, electromagnetic, magnetic or radioactive fields.

Communication module. Do not transmit without antenna or attenuator. Be mindful of RF interference issues.