LM-130 EVB II
Operation Manual

Version: 1.1

GlobalSat WorldCom Corporation

16F., No. 186, Jian 1st Rd, Zhonghe Dist.,

New Taipei City 23553, Taiwan

Tel: 886.2.8226.3799/ Fax: 886.2.8226.3899
service@globalsat.com.tw
www.globalsat.com.tw

USGlobalSat Incorporated

14740 Yorba Court Chino, CA 91710

Tel: 888.323.8720 / Fax: 909.597.8532
sales@usglobalsat.com
www.usglobalsat.com

Revision History

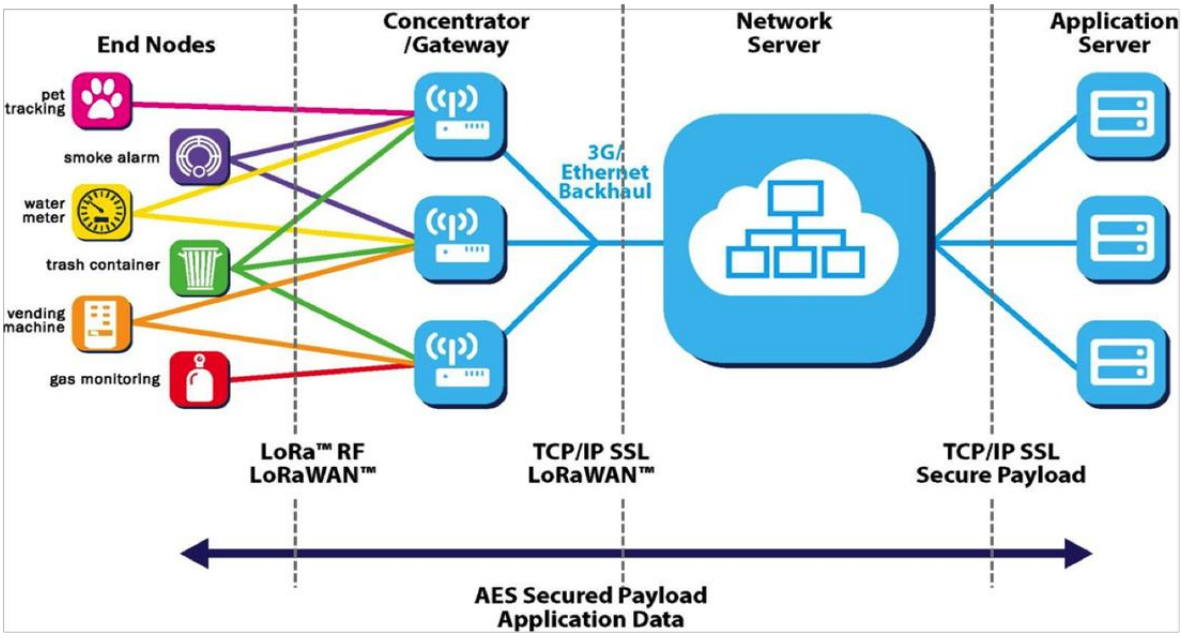
Rev. No.	Change History	Issue Date	Remark
1.0	Formal release	2016.6.8	Peter
1.1	[4.2 Firmware Behavior] 1. Add the description of converting payload to the proper format. 2. The self-defined payload by sending AT command would be disabled when active report mode set to on.	2016.6.20	Peter

CONTENT

1. Introduction

The LM-130 EVB II LoRa® is a LoRaWAN™ Class A end-node device based on the GlobalSat LM-130 module which is a LoRaWAN™ certified module. The LM-130 EVB II is a standalone battery powered node with power management, it includes humidity and temperature sensors to generate data, which are transmitted either on a regular schedule (can be configured) or initialed by a button-press. The LM-130 EVB II provides a convenient platform to quickly demonstrate the long-range and low power consumption capabilities of the modem, as well as interoperability when connected to LoRaWAN™ v1.0 compliant gateways and the infrastructure.

The LM-130 EVB II also provides a standard USB interface for connection to a host computer, providing a bridge to the UART interface of the LM-130 module. It can also help the developers to develop the applications, including hardware and software design, before the end product is ready. The LM-130 EVB II is everything you need to get started with your design.

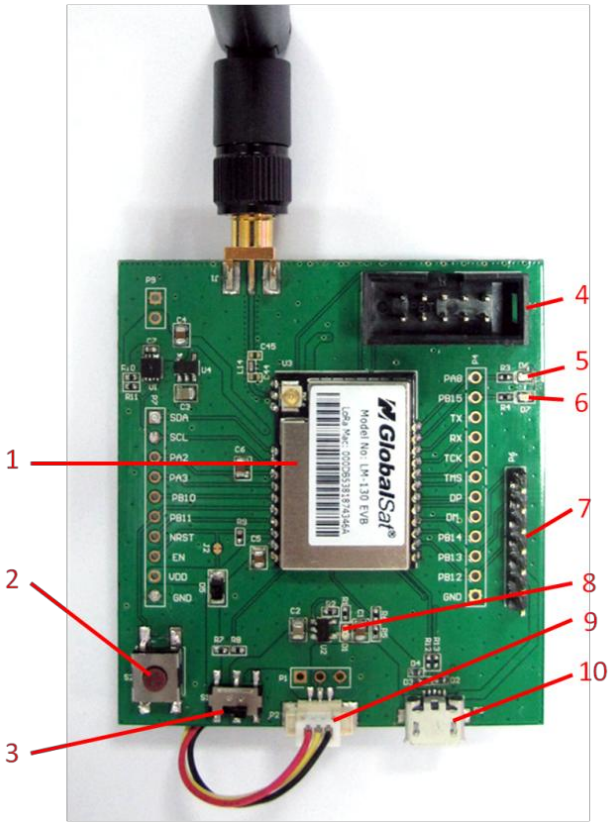


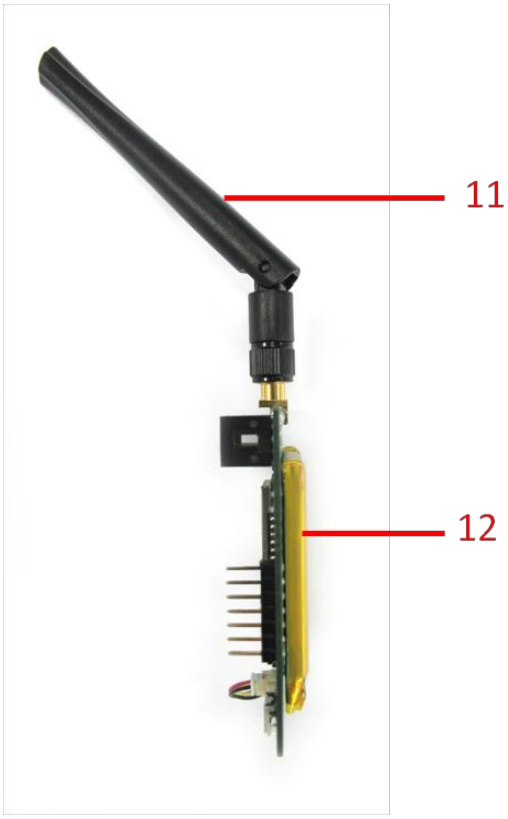
Features:

- **LoRaWAN™ Compliant Evaluation Board.**
- **RF ISM band, Supports 868/915 MHz**
- **Long range transmission (1km to 10km)**
- **Push button (trigger event)**
- **Configurable payload**
- **Configurable report interval**
- **Active report mode**
- **Built-in humidity/Temperature sensor**

2. Setup EVB

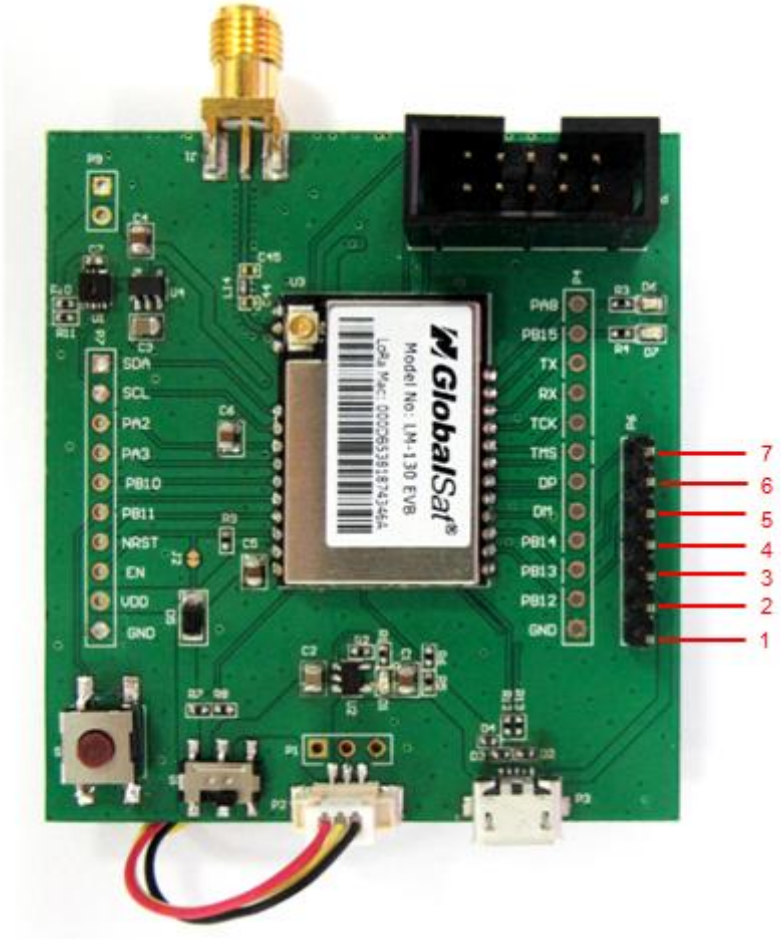
LM-130 EVB's component introduction is as below:





1	LM-130 Module	7	Converting board connector
2	Push Button	8	Charging Status LED
3	Power Switch	9	Battery connector
4	No Fuction	10	Micro USB
5	LoRa Status LED	11	LoRa Antenna
6	Power Status LED	12	Battery

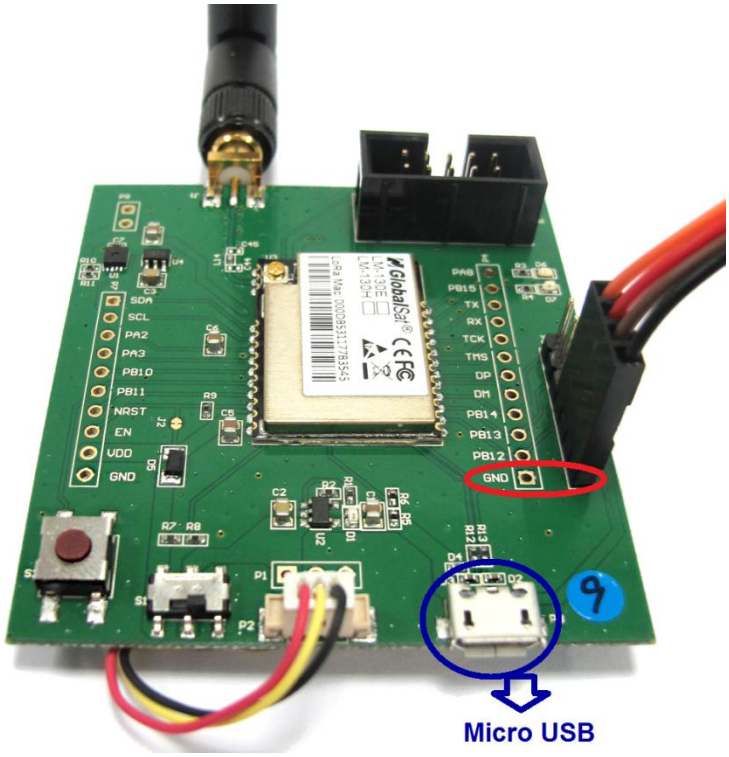
Please see pin definition of converting board connector as below:

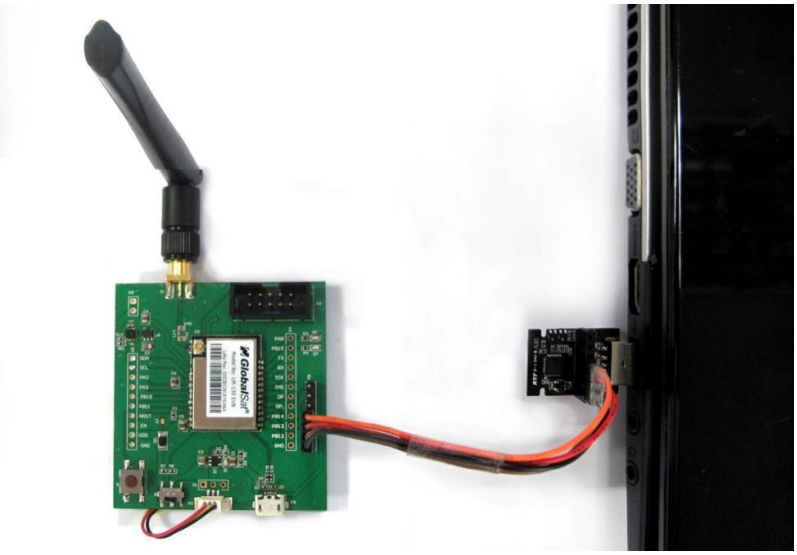


Pin No.	Definition
7	None Connect
6	None Connect
5	None Connect
4	LoRa's TX
3	LoRa's RX
2	V_USB
1	GND

1) Please connect the converting board as below. The black wire must be connected to the pin as below, and the USB side should be connected to PC. Charging status LED (Red) would be on.

You could also connect with LM-130 EVB II with your PC by micro USB.





2) Please turn on LM-130 EVB II by pushing power switch to the left side. You could check LM-130 EVB II is on when the LoRa status LED (Blue) flashes.

3) Please refer to “3. Access into network server” to access into network server.

3. Access into network server

End-device can be activated by “Over-The-Air Activation” (OTAA) or by “Activation By Personalization” (ABP). You might need to send AT commands to activate OTAA or ABP by serial terminal tools. We recommend “RealTerm”. You could download it at the following link http://realterm.sourceforge.net/.

Please follow the steps below for activating LM-130 EVB II to access into the network server.

1. **For ABP Mode**
   1) Make sure the gateway already connect to the internet.
   2) Make sure LM-130 EVB II’s information (DevAddr, AppSKey, NwkSKey) has been successfully registered to the network server.
   3) Use AT command (AAT2 JoinMode=0) to switch LM-130 EVB II from OTAA to ABP mode. You would see “ok” when LM-130 EVB II is switched to ABP mode. Use AT command (AAT1 Save) to save the setting to the flash. You would see “ok” when the settings are saved. Use AT command (AAT1 Reset) to run the new settings.
   4) Use AT command (AAT2 Tx=2,cnf,aabbccdd) to perform the interop testing. The payload here is “aabbccdd”. You could set other payload uplink to server.
   5) Check if the payload on the network server is correct.
2. **For OTAA Mode**
   1) Make sure the gateway already connect to the internet.
   2) Make sure LM-130 EVB II’s information (DevEUI, AppEUI, AppKey) has been successfully registered to the network server.
   3) Use AT command (AAT2 JoinMode=1) to switch LM-130 EVB II from ABP to OTAA mode. You would see “ok” when LM-130 EVB II is switched to ABP mode. Use AT command (AAT1 Save) to save the setting to the flash. You would see “ok” when the settings are saved. Use AT command (AAT1 Reset) to run the new settings.
   4) Use AT command (AAT2 Tx=2,cnf,aabbccdd) to perform the interop testing. The payload here is “aabbccdd”. You could set other payload uplink to server.
   5) Check if the payload on the network server is correct.

Note:

The example of our DevEUI, DevAddr and Default key is as following table. **DevEUI and DevAddr would be unique for each device.** DevEUI and DevAddr can be retrieved by

following AT commands. (AAT2 DevEui=?, AAT2 DevAddr=?)

	US ISM Band	**EU ISM Band**
DevEUI	000DB533188A3572	000DB5331760356B
DevAddr	188A3572	1760356B
Default Key	**1. ABP:** NwkSKey=28AED22B7E1516A609CFABF715884F3C (Hexadecimal number) AppSKey=1628AE2B7E15D2A6ABF7CF4F3C158809 (Hexadecimal number) **2. OTAA** AppEUI=0000000000010203 (Hexadecimal number) AppKey=0123456789ABCDEFEFCDAB8967452301 (Hexadecimal number)	

4. Specifications

4.1 Hardware Specifications

Item	Parameters
LoRa Module	Globalsat LM-130 LoRaWAN™ module
RF Band	868MHz/915 MHz
RF Output Power	Max. 20dBm
Receiving Sensitivity	-132dBm @ DR0
Dimensions	71 x 55 x 15 mm (not including antenna)
Interface	UART
Battery	Re-chargeable Li-polymer battery 820mAh Over consumption protection
Sensor	Temperature/humidity SHT3x-DIS
LED Indicator	Power on: Yellow LED light on. Power off: Yellow LED light off. Charging: Charging Status LED (Red) On. Full battery: Charging Status LED (Red) Off. Data transmitting: LoRa Status LED (Blue) blinking. End data transmitting: LoRa Status LED (Blue) Off.
Operation Conditions	Temperature -20℃ ~ 60℃; Humidity 5% ~ 95%
Micro USB	Charging @ 500mA & Sending AT commands
Button	Power Switch : On/Off Push button: send report

4.2 Firmware Behavior

Active report mode: On (Default)

LM-130 EVB II reports default data to gateway **by interval automatically**.

The push button is disabled when the active report mode is on.

- Configurable report interval. (default = 10 seconds)
- Default Payload includes “GlobalSat”, “LM-130_EVK”, battery voltage, temperature & humidity sensor data. **(Hexadecimal number, 24 Bytes)**.
- **The self-defined payload by sending AT command would be disabled.**

Active report mode: Off

LM-130 EVB II reports the default data to gateway once the **push button is pressed**.

- Default Payload includes “GlobalSat”, “LM-130_EVK”, battery voltage, temperature & humidity sensor data. **(Hexadecimal number, 24 Bytes)**.
- You could send the self-defined payload by sending AT command. Please refer to “5.AT Command” concerning **AAT2 Tx= [parameter1], [parameter2],[parameter3]**.

P.S. For converting payload to the proper format, please see the picture below.

EX: Payload Data=476c6f62616c5361744c4d3133305f45564b4610610a8c0f (24 Bytes, Hex format)

																			Battery Voltage		Temperature		Humidity	
	Hex -> ASCII																		Hex -> Decimal					
																			Hex(Low)	Hex(High)	Hex(Low)	Hex(High)	Hex(Low)	Hex(High)
Payload Data	47	6C	6f	62	61	6C	53	61	74	4C	4D	31	33	30	5F	45	56	4B	46	10	61	0a	8C	0f
Transformed Data	G	l	o	b	a	l	S	a	t	L	M	1	3	0	_	E	V	K	4166		2657		3980	

Value Description	
Battery Voltage	4166 / 100 = 4.166V
Temperature	2657 / 100 = 26.57℃
Humidity	3980 / 100 = 39.8 %RH

5. AT Commands

All of the LM-130 module's settings and commands are transmitted over UART using the ASCII interface. All commands need to be terminated with <CR><LF> and any replies they generate will also be terminated by the same sequence.

After setting LM-130 EVB II, you must use AT command (AAT1 Save) to save the settings to the flash. You would see "ok" when the settings are saved. Then use AT command (AAT1 Reset) to run the new settings.

The settings for the UART interface are 57600 bps, 8 bits, no parity, 1 Stop bit, no flow control.

Noted: All AT command are case sensitive.

Command	Description
AAT1 UpdateFW	Upgrade the LM-130 module firmware. Response ***ok*** after entering the command
AAT1 Save	All parameters are saved. Response ***ok*** after parameters are saved.
AAT1 FwVersion	Show up firmware version.
AAT1 Reset	Resets and restarts the LM-130 module. Response ***ok*** after entering the command
AAT1 SLEEP	Put LM-130 into sleep mode. To leave sleep mode, just Input 0xFF by UART to wake up LM-130. Response ***ok*** after entering the command
AAT1 Restore	Restore the defaults of FW. Response ***ok*** after entering the command.
AAT1 TestMode=[parameter1]	[parameter1]: 0: Disable **(Active Report Mode: Off)** 1: Enable **(Active Report Mode: On)** Response: ***ok*** if value is valid ***invalid_param*** if parameter1 is not valid **This command sets the state of the active report**

Command	Description
	mode for the module
AAT1 TestMode=?	Response: 0: disable (Active Report Mode: Off) 1: enable (Active Report Mode: On) **This command will return the state of the active report mode.**
AAT2 DevAddr=[parameter1]	[parameter1]: 4-byte hexadecimal number representing the device address, from 00000000 – FFFFFFFF Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command configures the module with a 4-byte unique network device address [parameter1]. The [parameter1] must be unique to the current network. This must be directly set solely for activation by personalization devices. This parameter must not be set before attempting to join using over-the-air activation because it will be overwritten once the join process is over.
AAT2 DevAddr=?	Response: 4-byte hexadecimal number representing the device address, from 00000000 to FFFFFFFF. This command will return present end-device address of the module.
AAT2 DevEui=[parameter1]	[parameter1]: 8-byte hexadecimal number representing the device EUI Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets the globally unique device identifier for the module. The identifier must be set by the host MCU. The module contains a pre-programmed unique EUI and can be retrieved using user provided EUI can be configured using the

Command	Description
	AAT2 DevEui command.
AAT2 DevEui=?	Response: 8-byte hexadecimal number representing the device EUI. This command returns the globally unique end-device identifier, as set in the module.
AAT2 AppEui=[parameter1]	[parameter1]: 8-byte hexadecimal number representing the application EUI Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets the application identifier for the module.
AAT2 AppEui=?	Response: 8-byte hexadecimal number representing the application EUI. This command will return the application identifier for the module. The application identifier is a value given to the device by the network.
AAT2 NwkSKey=[parameter1]	[parameter1]: 16-byte hexadecimal number representing the network session key Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets the network session key for the module. This key is 16 bytes in length, and should be modified with each session between the module and network. The key should remain the same until the communication session between devices is terminated.
AAT2 NwkSKey=?	Reponse: [parameter1]: 16-byte hexadecimal number representing the network session key This command sets the network session key for the module.
AAT2 AppSKey=[parameter1]	[parameter1]: 16-byte hexadecimal number representing the application session key Response: ***ok*** if address is valid

Command	Description
	invalid_param if parameter1 is not valid This command sets the application session key for the module. This key is unique, created for each occurrence of communication, when the network requests an action taken by the application.
AAT2 AppSKey=?	Response: [parameter1]: 16-byte hexadecimal number representing the application session key This command sets the application session key for the module.
AAT2 AppKey=[parameter1]	[parameter1]: 16-byte hexadecimal number representing the application key Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets the application key for the module. The application key is used to identify a grouping over module units which perform the same or similar task.
AAT2 AppKey=?	Response: [parameter1]: 16-byte hexadecimal number representing the application key This command sets the application key for the module.
AAT2 ADR=[parameter1]	[parameter1]: 0: disable 1: enable Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets if the adaptive data rate (ADR) is to be enabled, or disabled. The server is informed about the status of the module's ADR in every uplink frame it receives from the ADR field in uplink data

Command	Description
	packet. If ADR is enabled, the server will optimize the data rate and the transmission power of the module based on the information collected from the network.
AAT2 ADR=?	Response: 0: disable 1: enable This command will return the state of the adaptive data rate mechanism.
AAT1 EVK_TxCycle=[parameter1]	[parameter1]: decimal number representing the report interval in seconds, from 1 to 254. This command will only take effect when "TestMode"=1. Response: ***ok*** if parameter1 is valid ***invalid_param*** if parameter1 is not valid This command sets the report interval for the module.
AAT1 EVK_TxCycle=?	Response: decimal number representing the interval, in seconds, for EVK_TxCycle, from 1 to 254. This command will return the interval, in seconds, for EVK_TxCycle.
AAT2 JoinMode=[parameter1]	[parameter1]: 0: ABP mode 1: OTAA mode Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command informs the ***module activation type.***
AAT2 JoinMode=?	Response: 0: ABP mode 1: OTAA mode

Command	Description
	This command will return the ***activation type*** of module.
AAT2 reTx=[parameter1]	[parameter1]: decimal number representing the number of retransmissions for an uplink confirmed packet, from 0 to 10. Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command sets the number of retransmissions to be used for an uplink confirmed packet, if no downlink acknowledgment is received from the server.
AAT2 reTx=?	Response: decimal number representing the number of retransmissions, from 0 to 10. This command will return the currently configured number of retransmissions which are attempted for a confirmed uplink communication when no downlink response has been received.
AAT2 RxDelay1=[parameter1]	[parameter1]:decimal number representing the delay between the transmission and the first reception window in microseconds, from 100000 to 10000000. Response: ***ok*** if address is valid ***invalid_param*** if parameter1 is not valid This command will set the delay between the transmission and the first reception window to the [parameter1] in microseconds. The delay between the transmission and the second Reception window is calculated in software as the delay between the transmission and the first Reception window + 1000000 (μs).
AAT2 RxDelay1=?	Response: decimal number representing the interval, in microseconds, for RxDelay1.

Command	Description
	This command will return the interval, in microseconds, for RxDelay1.
AAT2 Tx=[parameter1], [parameter2], [parameter3]	[parameter1]: decimal number representing the port number, from ***1*** to ***223***. [parameter2]: string representing the uplink payload type, either ***"cnf"*** or ***"uncnf"*** (cnf = confirmed, uncnf = unconfirmed) [parameter3]: Hexadecimal number representing the payload value. The length of [parameter3] bytes capable of being transmitted are dependent upon the set data rate (please refer to the LoRaWAN™ Specification for further details). Response: this command may reply with two responses. The first response will be received immediately is valid (ok reply received), a second reply will be received after the end of the uplink transmission. Please refer to the the LoRaWAN™ Specification for further details. Response after entering the command: ● ok - If parameters and configurations are valid. ● Invalid_param – if parameters ([parameter1],[parameter2],[parameter3]) are not valid. ● Tx_ok - if "cnf" radio Tx return with ack ● Tx_ok - if "uncnf" radio Tx return. ● Tx_noACK – if "cnf" radio Tx return without ack. ● Rx < parameter1> < parameter2>– if transmission was successful, [parameter1] port number, from 1 to 223; [parameter2] Hexadecimal number that was received from the server.