

## Lora Radio Modem AT Command

Lora Transparent Transmission Protocol

**RC-RICK-868** is a radio modem with UART interface usable with AT commands for long distance communication with LoRa modulation (LoRa Transparent Transmission Protocol).

Working into European bandwidth : 868MHz

The hardware is based on the module RC-WLE5-868 designed by RadioControlli (STM32WLE5JC based).

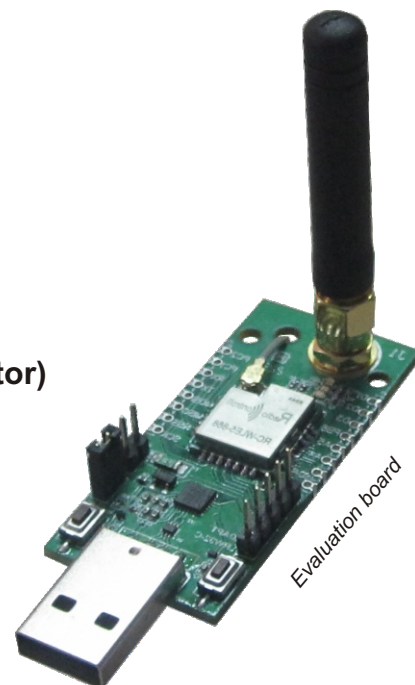
For more information and details, please refer to :

- RC-WLE5-868 datasheet ([www.radiocontrolli.com](http://www.radiocontrolli.com))
- STM32WLE5JC datasheet ([www.st.com](http://www.st.com))

Module Information :

**RC-RICK-868** → Standard Version (UFL Connector)

**RC-RICK-868-HA** → Helical Antenna Version



The module is designed to be easily integrated into your hardware application, we provide 2 versions:

### 1) Standard version with UFL connector

You can use the UFL connector where you can directly connect the antenna. The RF output signal is also present on a 50 ohm pin (pin 15) in case you want to create your own antenna project on your pcb.

### 2) Version with helical antenna (full version)

#### Applications:

- Smart meters
- Wireless security systems
- Home and Building automation
- 6LoWPAN systems
- Automatic Measure Reading
- Low-Power Wireless Systems
- Wireless Sensor Networks
- Remote Control
- Street Lights System
- Parking Sensors
- Environmental Sensors
- Smart Grid and Automatic Meter Reading

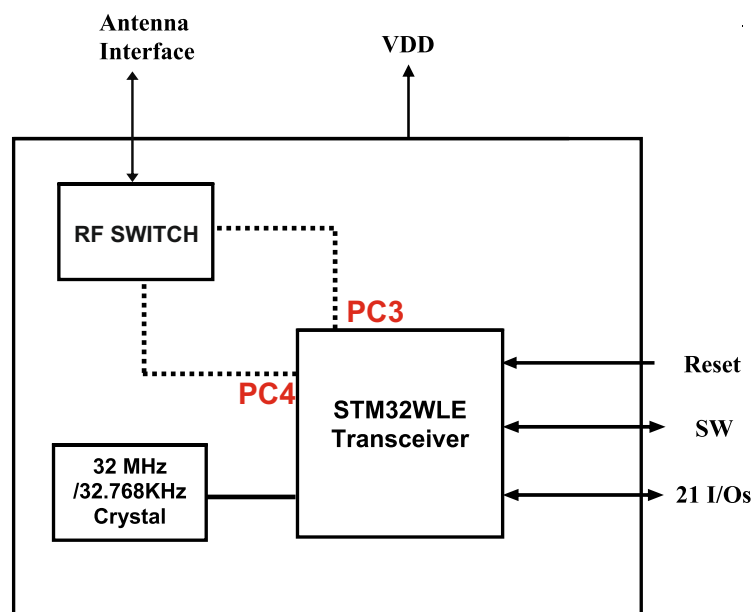
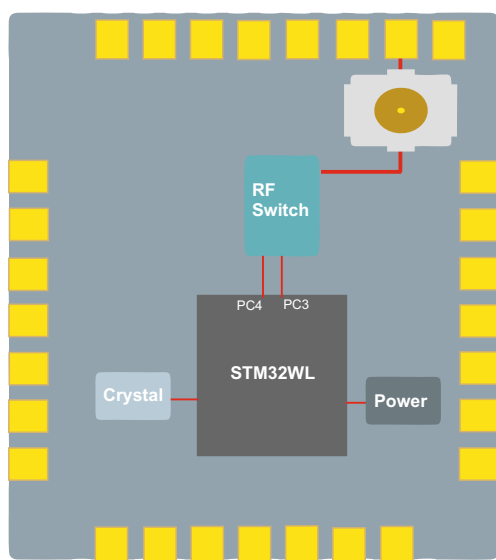
## TECHNICAL CHARACTERISTICS

Characteristics	MIN	TYP	MAX	UNIT
V <sub>CC</sub> Supply Voltage	2.5	3.3	3.7	Vdc
I <sub>s</sub> Supply Current ( Sleep mode )		2.0		uA
I <sub>s1</sub> Supply Current (Receive mode)		5.0		mA
I <sub>s2</sub> Supply Current (Transmit mode)		120.0		mA
F Frequency Band		868.0		Mhz
T RF TX Power		18.5	22.0	dBm
S RX Sensitivity		140		dBm
TE Operating Temperature Range	-20		+70	°C

## BLOCK DIAGRAM

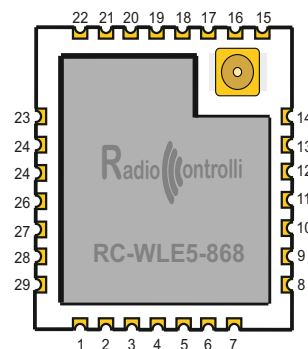
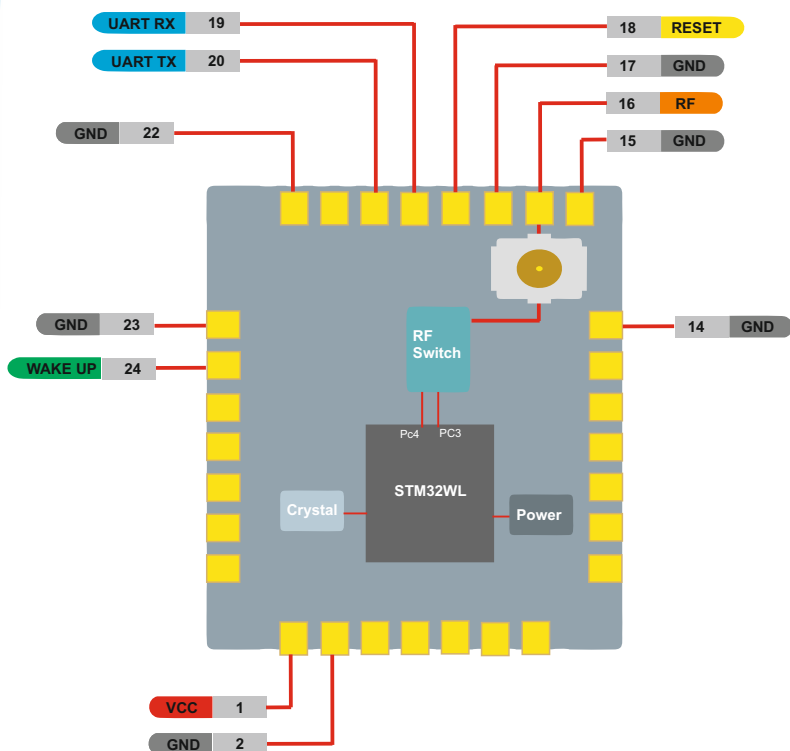
The module is based on STM32WL5Jx device which integrates :

- ARM Cortex M4 processor
- Sx126x LoRa transceiver on the same chip.

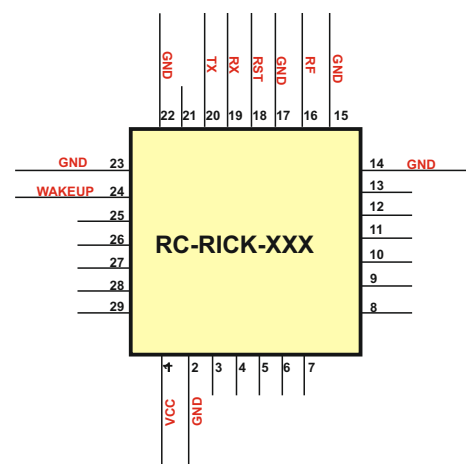


	PC3	PC4	RF Mode
RF Switch Control	1	0	TX
	0	1	RX

## PINOUT



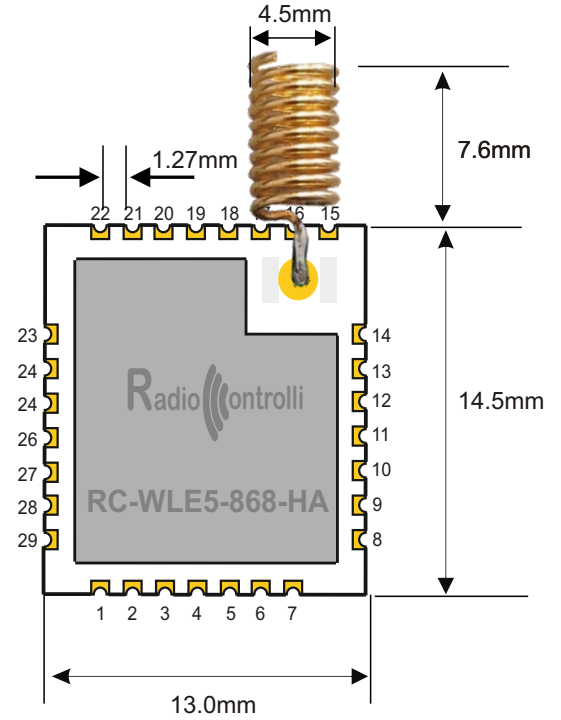
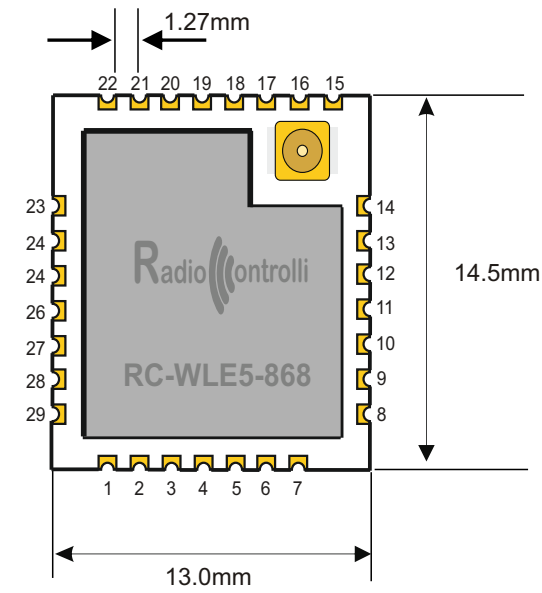
Standard Version



### Pin Descriptions

Pin Number	Name	Description
01	VCC	Supply Voltage
2,14,15,17,22,23	GND	Ground
16	RF	Antenna RF OUT
18	RST	Reset Trigger Input
19	RX	UART RX Input
20	TX	UART TX Input
24	WakeUp	When the module is in sleep mode, it's possible to pull this pin in high status to wake up the module.

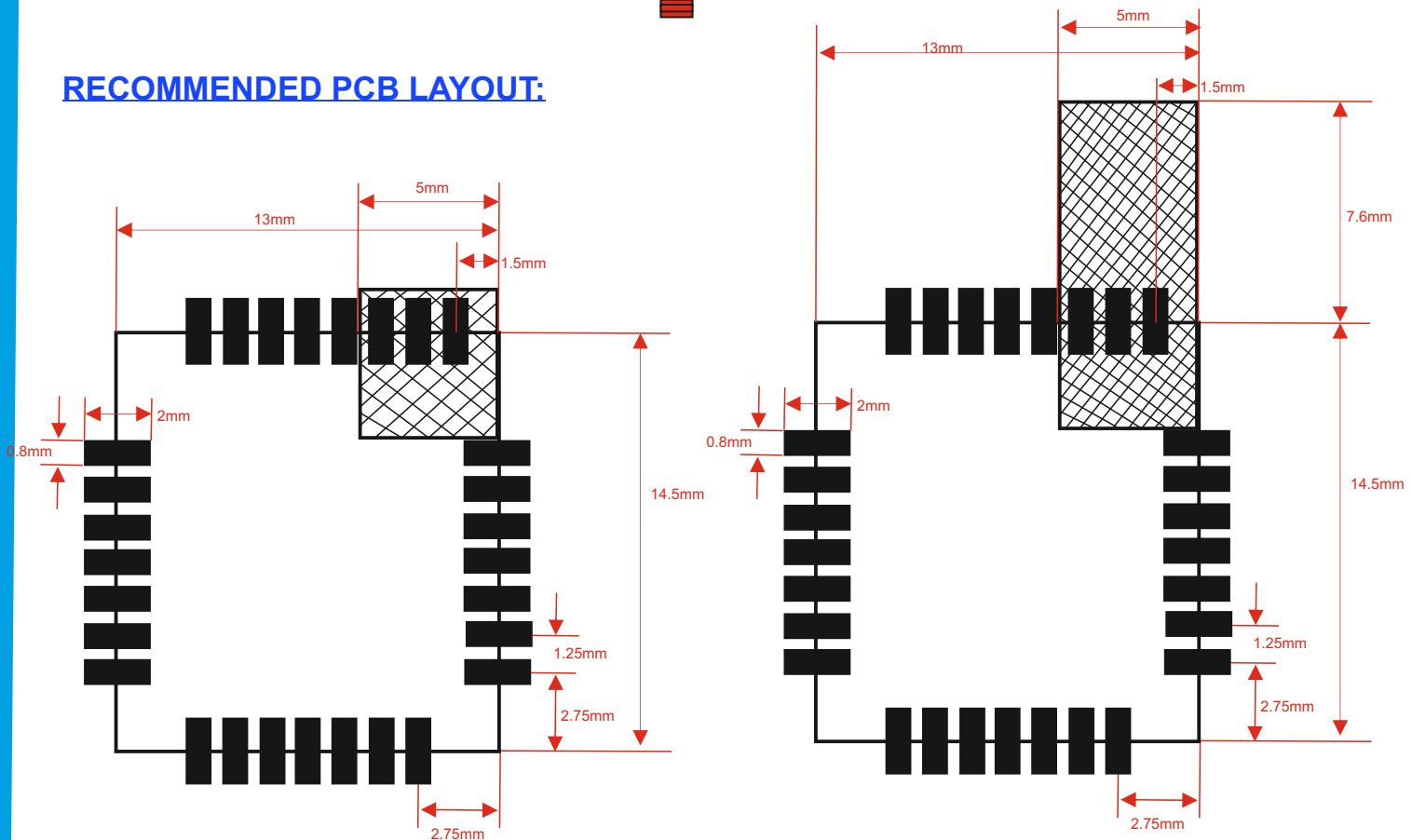
## MECHANICAL DIMENSIONS:



Tolerance  $\pm 0.5\text{mm}$



## RECOMMENDED PCB LAYOUT:



Note :  
Make sure in the shadow area shall without any wiring or ground.

## AT Command instruction

The AT commands are used to drive the LoRa module and to send data. AT commands are sent through the UART peripheral.

Serial Port parameters:

- Baud rate: 9600
- Data: 8 bit
- Parity: none
- Stop: 1 bit
- Flow control: none

AT command is based on ASCII code.

Request message: AT+<CMD>[OP][parameter-1,parameter-2,...,parameter-n]<CR><LF>

token	Description
AT+	prefix
CMD	command name
OP code	operational character, use OP code : [OP1]"=" : setting [OP2]"?" : inquiry current command instruction [OP3]"'" : execution [OP4]"=?" : inquiry current command content
Parameter-1, parameter-n	Parameters setting
<CR><LF>	

Response format : <CR><LF>[+CMD:] [parameter-1,parameter-n]<CR><LF>  
Or <CR><LF><STATUS><CR><LF>

token	Description
+CMD:	command
Parameter-1 ... parameter-n	Back to current character
STATUS	execution status "OK" : exe success "ERROR": exe failure

## AT Command (**Basic Command**)

Description	<b>Read Firmware version</b>
Syntax	+VER=?<CR><LF>
Format	AT+VER=? <CR><LF>
Response	<CR><LF>+VER:[parameter-1]<CR><LF>
ResultCode	<CR><LF>OK<CR><LF>

```
12:32]>>>AT+VER=?
[12:32]<<<
+VER:v1.4.2_X0.1
OK
```

Description	<b>Open and Close LOG</b>
Syntax	+LOG=[parameter-1] <CR><LF>
Format Command	AT+LOG=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+LOG=? <CR><LF>
Response Query	<CR><LF>+LOG:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*parameter-1 = 0 close*

*parameter-1 = 1 open*

```
[14:03]>>>AT+LOG=1
[14:03]<<<
OK
```

```
[14:12]>>>AT+LOG=?
[14:12]<<<
+LOG:1
OK
```

Description	<b>Sleeping Mode</b>
Syntax	+Sleep=[parameter-1] <CR><LF>
Format Command	AT+Sleep<CR><LF>
Response Command	<CR><LF>
Format Query	AT+Sleep? <CR><LF>
Response Query	<CR><LF>+Sleep<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*In Sleeping mode will be necessary to use the wake-up pin to wake up*

```
[10:34]>>>AT+Sleep
[10:34]<<<
OK
```

```
[10:35]>>>AT+Sleep?
[10:35]<<<
AT+Sleep entry low power mode
```

## AT Command (Basic Command)

Description	<b>Open and Close automatic saving command</b>
Syntax	+AutoSave=[parameter-1] <CR><LF>
Format Command	AT+AutoSave=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+AutoSave=? <CR><LF>
Response Query	<CR><LF>+AutoSave:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 0=close 1=open

```
[10:55]>>>AT+AutoSave=1
[10:55]<<<
OK
```

```
[10:55]>>>AT+AutoSave=?
[10:55]<<<
+AutoSave:1
OK
```

Note :

With AutoSave=1 the value of the parameters below are saved in flash memory , ready to be used when the device is turned on again.

With AutoSave=0 the value of the parameters below are not saved in flash memory.

Command AutoSave : AT-Freq, AT+BW ,AT+SW , AT+CW, AT+RX, AT+ACKRSSI  
AT+PreambleLen, AT+LPRX , AT+LOG , AT+CadParams  
AT+AESKEY, AT+ENAES , AT+TXPW

Description	<b>Set &amp; and get UART baudrate (default 9600bps)</b>
Syntax	+Baud=[parameter-1] <CR><LF>
Format Command	AT+Baud=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+Baud? <CR><LF>
Response Query	<CR><LF>+Baud:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 0=115200 1=57600 2=38400 3=19200 4=9600 5=4800 6=2400value

```
[10:48]>>>AT+Baud=4
[10:48]<<<
OK
```

```
[10:45]>>>AT+Baud=?
[10:45]<<<
+Baud:4
OK
```

## AT Command (Basic Command)

Description	<b>Receiving data setting RSSI SNR</b>
Syntax	+ACKRSSI=[parameter-1] <CR><LF>
Format Command	AT+ACKRSSI=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+ACKRSSI=? <CR><LF>
Response Query	<CR><LF>+ACKRSSI:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*parameter-1 = 1 mean enable RSSI and SNR 0 mean disable RSSI and SNR*

```
[10:04]>>>AT+ACKRSSI=1
[10:04]<<<
OK
```

```
[10:05]>>>AT+ACKRSSI=?
[10:05]<<<
+ACKRSSI:1
OK
```

```
[15:51]<<<Hello world
RSSI=-20 dBm,SNR=10
OK
16126s508:OnRxDone
+EVT:RxDone
```

examples of the packet received with ACKRSSI=1

Description	<b>Read current RSSI</b>
Syntax	+ED=?<CR><LF>
Format	AT+ED=? <CR><LF>
Response	<CR><LF>+ED:[parameter-1]<CR><LF>
ResultCode	<CR><LF>OK<CR><LF>

```
[16:29]>>>AT+ED=?
[16:29]<<<
+ED=-108
OK
```



## AT Command (Radiofrequency Command)

Description	<b>Set LoRa Frequency</b>
Syntax	+Freq=[parameter-1] <CR><LF>
Format Command	AT+Freq=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+Freq=? <CR><LF>
Response Query	<CR><LF>+Freq:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 868000000-900000000

```
[15:03]>>>AT+Freq=868100000
[15:03]<<<
OK
```

```
[15:04]>>>AT+Freq=?
[15:04]<<<
+Freq:868100000
OK
```

Description	<b>Set LoRa Bandwidth</b>
Syntax	+BW=[parameter-1] <CR><LF>
Format Command	AT+BW=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+BW? <CR><LF>
Response Query	<CR><LF>+BW:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 0 : 125KHz    1 : 250KHz    2 : 500KHz    3 : 62KHz    4 : 41KHz    5 : 31KHz  
 6 : 20KHz    7 : 15KHz    8 : 10KHz    9 : 7KHz

```
[15:03]>>>AT+BW=3
[15:03]<<<
OK
```

```
[15:18]>>>AT+BW=?
[15:18]<<<
+BW:3
OK
```

## AT Command (Radiofrequency Command)

Description	<b>Set LoRa Spreading Factor</b>
Syntax	+SF=[parameter-1] <CR><LF>
Format Command	AT+SF=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+SF? <CR><LF>
Response Query	<CR><LF>+SF:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = value between 6-12

```
[15:03]>>>AT+SF=9
[15:03]<<<
OK
```

```
[15:24]>>>AT+SF=?
[15:24]<<<
+SF:9
OK
```

Description	<b>Set LoRa Coding Rate</b>
Syntax	+CR=[parameter-1] <CR><LF>
Format Command	AT+CR=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+CR? <CR><LF>
Response Query	<CR><LF>+CR:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = value between 1-4

```
[15:03]>>>AT+CR=4
[15:03]<<<
OK
```

```
[15:27]>>>AT+CR=?
[15:27]<<<
+CR:4
OK
```

## AT Command (Radiofrequency Command)

Description	<b><i>RX Mode Time-out</i></b>
Syntax	+RX=[parameter-1] <CR><LF>
Format Command	AT+RX=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+RX? <CR><LF>
Response Query	<CR><LF>+RX:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*parameter-1 = RX time-out value units is ms, 0 means no time-out value*

```
[15:03]>>>AT+RX=0
[15:03]<<<
OK
```

```
[09:32]>>>AT+RX=?
[09:32]<<<
+RX:0
OK
```

Description	<b><i>Change RF to be wait mode</i></b>
Syntax	+Standby<CR><LF>
Format Command	AT+Standby<CR><LF>
Response Command	<CR><LF>
Format Query	AT+Standby? <CR><LF>
Response Query	<CR><LF>+AT+Standby switch RF to standby mode<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

```
[15:03]>>>AT+Standby
[15:03]<<<
OK
```

```
[09:42]>>>AT+Standby?
[09:42]<<<
AT+Standby switch RF to standby mode
```

## AT Command (Radiofrequency Command)

Description	<b><i>Set and get TX power</i></b>
Syntax	+TXPW=[parameter-1] <CR><LF>
Format Command	AT+TXPW=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+TXPW=? <CR><LF>
Response Query	<CR><LF>+TXPW:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*parameter-1 = value between 0-22*

```
[10:13]>>>AT+TXPW=22
[10:13]<<<
OK
```

```
[10:14]>>>AT+TXPW=?
[10:14]<<<
+TXPW:22
```

```
OK
```

Description	<b><i>Transfer the text and specify a lenght</i></b>
Syntax	+Send[parameter-1],[parameter-2] <CR><LF>
Format Command	AT+Send=[parameter-1],[parameter-2] <CR><LF>
Response Command	<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

*parameter-1 = data len*

*parameter-2 = data*

## AT Command **LowPower RX Mode Command**)

Description	<b><i>Set &amp; and get lenght</i></b>
Syntax	+PreambleLen=[parameter-1] <CR><LF>
Format Command	AT+PreambleLen=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+PreambleLen? <CR><LF>
Response Query	<CR><LF>+PreambleLen:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = value from 1 to 65535

```
[11:52]>>>AT+PreambleLen=8
[11:52]<<<
OK
```

```
[11:52]>>>AT+PreambleLen=?
[11:52]<<<
+PreambleLen:8
OK
```

Note : TX & RX Preamble Lem must be the same.

Description	<b><i>Low power model</i></b>
Syntax	+LPRX=[parameter-1] <CR><LF>
Format Command	AT+LPRX=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+LPRX? <CR><LF>
Response Query	<CR><LF>+LPRX:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = value from 100 to 4000 ms

```
[11:58]>>>AT+LPRX=1000
[11:58]<<<
OK
```

```
[11:58]>>>AT+LPRX=?
[11:58]<<<
+LPRX:1000
OK
```

Note : TX & RX Preamble Lem must be the same.

## AT Command ( LowPower RX Mode Command)

Description	<b>Set wake up character by AIR</b>
Syntax	+CadParams[parameter-1],[parameter-2],[parameter-3]<CR><LF>
Format Command	AT+CadParams=[parameter-1],[parameter-2][parameter-3] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+CadParams=? <CR><LF>
Response Query	<CR><LF>+LPRX:[parameter-1],[parameter-2][parameter-3]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 0 = CAD\_01\_SYMBOL  
 1= CAD\_02\_SYMBOL  
 2= CAD\_04\_SYMBOL  
 3= CAD\_08\_SYMBOL  
 4= CAD\_16\_SYMBOL  
 DEFAULT = 2

```
[16:26]>>>AT+CadParams=?
[16:26]<<<
+CadParams:2,28,10
OK
```

parameter-2 check CAD peak (data:0-255)  
 DEFAULT=28

parameter-3 check CAD Min. (data:0-255)  
 DEFAULT=10

## AT Command ( RF Testing Command for laboratoring use)

Description	<b>Set transmission continually</b>
Syntax	+DutyTest[parameter-1],[parameter-2],[parameter-3]<CR><LF>
Format Command	AT+SutyTest=[parameter-1],[parameter-2][parameter-3] <CR><LF>
Response Command	<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = packages quantity ( 0 to illimitate)

parameter-2 delay second x package

parameter-3 transmission

Example: AT+DutyTest=0,200,AAAAAAAAAA  
 Transmission unlimited, AAAAAAAAAA per 200ms

## AT Command (Encryption transmission Command)

Description	<b>Set transmission key using AES256 for encryption</b>
Syntax	+AESKEY[parameter-1]<CR><LF>
Format Command	AT+AESKEY=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+AESKEY=? <CR><LF>
Response Query	<CR><LF>+AESKEY:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter-1 = 64pc character key in hexadecimal.

Ex:

```
AT+AESKEY=2121212122222222232323232424242425252525262626262727272728282828
```

2121212122222222232323232424242425252525262626262727272728282828 is the key, range is :0-9, A-F,a-f.

```
[[16:40]>>>AT+AESKEY=?
[16:40]<<<
+AESKEY:41414141424242424343434444444445454546464647474748484848
OK
OK
```

Description	<b>Open &amp; Close encryption transmission</b>
Syntax	+ENAES[parameter-1]<CR><LF>
Format Command	AT+ENAES=[parameter-1] <CR><LF>
Response Command	<CR><LF>
Format Query	AT+ENAES=? <CR><LF>
Response Query	<CR><LF>+ENAES:[parameter-1]<CR><LF>
Result Code	<CR><LF>OK<CR><LF>

parameter -1 = 0:close 1: open

When open, TX& RX terminal must set with same key code and must transmission or receiving code by this command.

```
[16:46]>>>AT+ENAES=?
[16:46]<<<
+ENAES:0
OK
```

## Quick Start Guide

To proceed with this test, use a RC-RICK-868-EV evaluation board which is composed as follows:

- RC-RICK-868 module
- 868MHz Antenna
- CP2102 USB/serial interface

More information about this evaluation board can be found on the last page of this document.

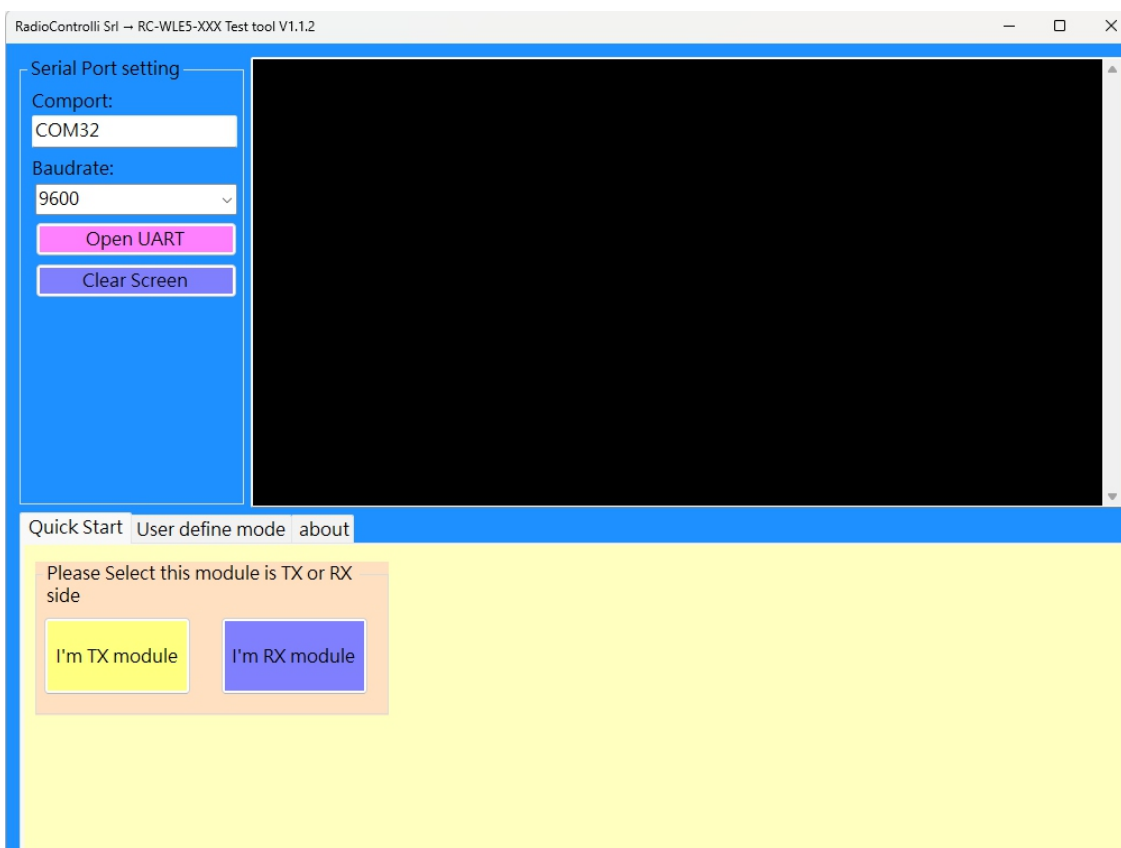
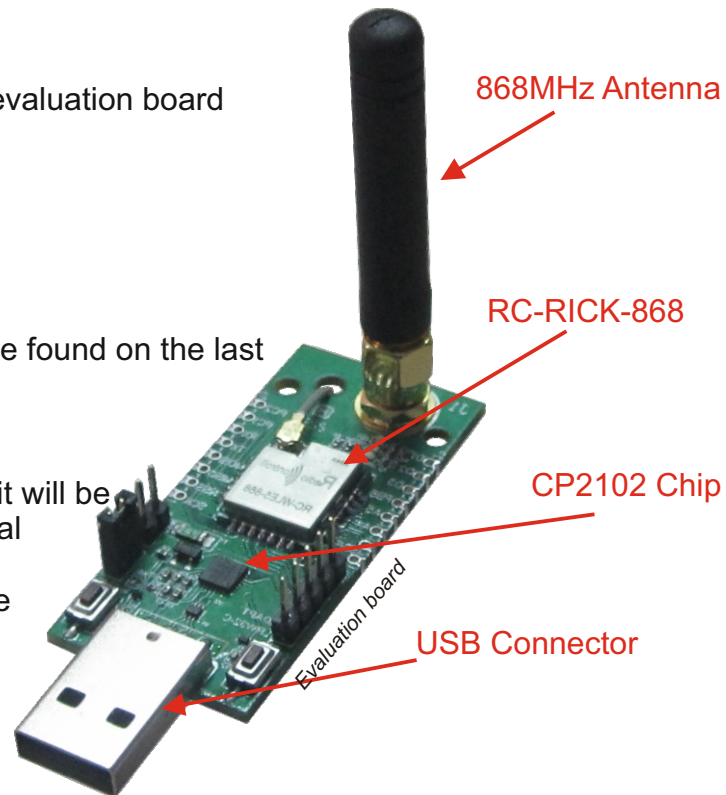
For user to start quickly to use the LORA module, it will be necessary to use a standard windows serial terminal (for example Putty, Real Term ect) or you can use a RadioControlli application downloadable from the following link :

**[DOWNLOAD AT COMMAND](#)**

Connect the LORA module through UART dongle to a Personal Computer.

The test will be done using two modules : one as RX, the other as TX.

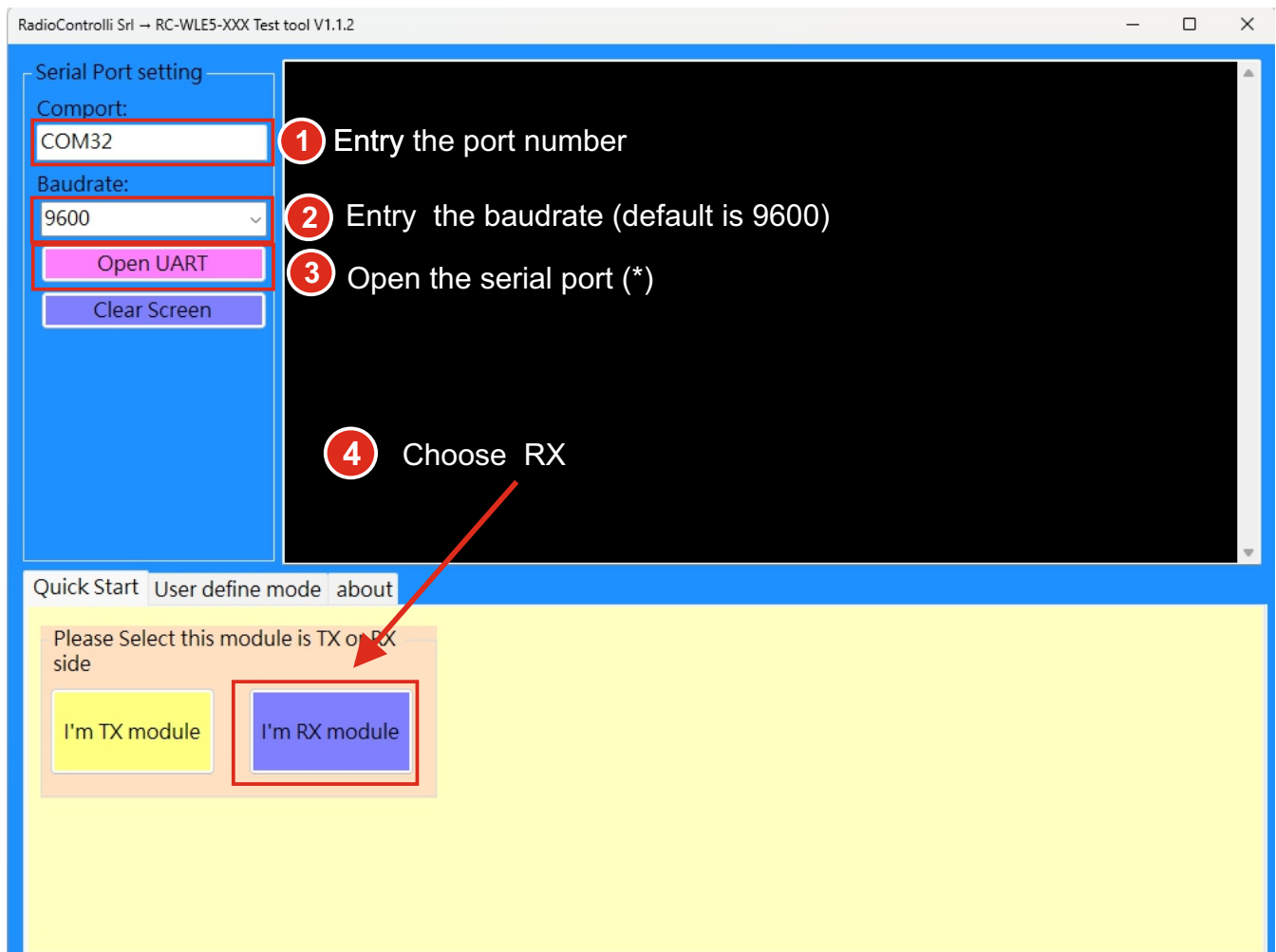
Launch the application RC-WLE5-XXX.exe.





### Quick Start : Receiving Station-1

- 1) Input the Comport and set the baudrate (9600 by default).
- 2) Click «Open UART»
- 3) Choose TX or RX Module to go on the corresponding setting function.
- 4) Choose RX



(\*) The USB Dongle RadioControlli has inside the Silicon Labs CP2102 chip as serial/usb interface, to open the serial port will be necessary to use the Silicon Labs driver.

## Quick Start : Receiving Station-2

The following figure shows the setting page of Rx mode, and send all the 01-06 instruction in sequence once, that is to complete the module setting of RX.

- AT+Freq=868000000      set frequency to 868MHz
- AT+BW=0                set Lora bandwidth to 125MHz
- AT+SF=9                set Lora spreading factor to 9
- AT+CR=4                set Lora coding rate to 4
- AT+PreambleLen=8    set PreambleLen to 8
- AT+RX

When the last instruction AT+RX will be executed, the module will always be in the receiving mode.

The screenshot shows the 'RadioControlli Srl - RC-WLE5-XXX Test tool V1.1.3' interface. On the left, the 'Serial Port setting' panel is visible with 'COM4' selected for the port and '9600' for the baudrate. The main terminal window displays the following sequence of commands and responses:

```
[12:12]>>>AT+Freq=868000000
[12:12]<<<
OK
[12:12]>>>AT+BW=0
[12:12]<<<
OK
[12:12]>>>AT+SF=9
[12:12]<<<
OK
[12:12]>>>AT+CR=4
[12:12]<<<
OK
[12:12]>>>AT+PreambleLen=8
[12:12]<<<
OK
[12:12]>>>AT+RX
[12:12]<<<
OK
```

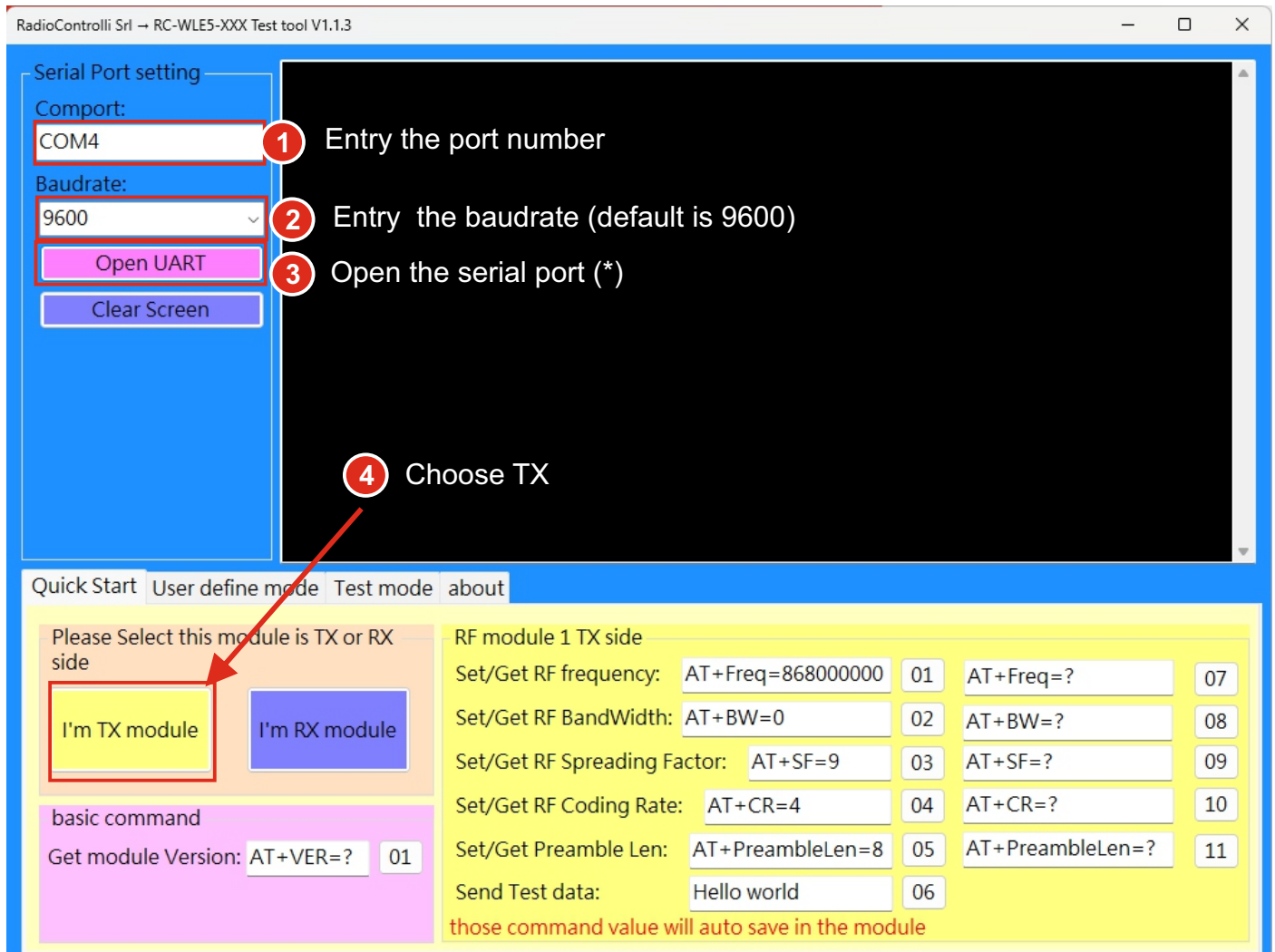
Annotations on the right side of the terminal window identify each command:

- first command set frequency
- second command set bandwidth
- third command set SF
- fourth command set coding rate
- fifth command set preamble len
- sixth command rx mode

At the bottom of the interface, the 'Quick Start' tab is active. It features a 'Please Select this module is TX or RX side' section with 'I'm TX module' and 'I'm RX module' buttons. The 'I'm RX module' button is highlighted with a red box. Below this, a 'basic command' section shows 'Get module Version: AT+VER=? 01'. The 'RF module 2 RX side' section contains a list of commands with corresponding input fields and numeric IDs (01-11). A red box highlights the numeric IDs 01 through 06, and a red circle with the number '1' is placed above this box. A note at the bottom of this section states: 'those command value will auto save in the module'.

### Quick Start : Transmission Station-1

- 1) Input the Comport and set the baudrate (9600 by default).
- 2) Click «Open UART»
- 3) Choose TX or RX Module to go on the corresponding setting function.
- 4) Choose TX



RadioControlli Srl → RC-WLE5-XXX Test tool V1.1.3

Serial Port setting

Comport: COM4

Baudrate: 9600

Open UART

Clear Screen

1 Enter the port number

2 Entry the baudrate (default is 9600)

3 Open the serial port (\*)

4 Choose TX

Quick Start User define mode Test mode about

Please Select this module is TX or RX side

I'm TX module I'm RX module

basic command

Get module Version: AT+VER=? 01

RF module 1 TX side

Set/Get RF frequency: AT+Freq=868000000 01 AT+Freq=? 07

Set/Get RF BandWidth: AT+BW=0 02 AT+BW=? 08

Set/Get RF Spreading Factor: AT+SF=9 03 AT+SF=? 09

Set/Get RF Coding Rate: AT+CR=4 04 AT+CR=? 10

Set/Get Preamble Len: AT+PreambleLen=8 05 AT+PreambleLen=? 11

Send Test data: Hello world 06

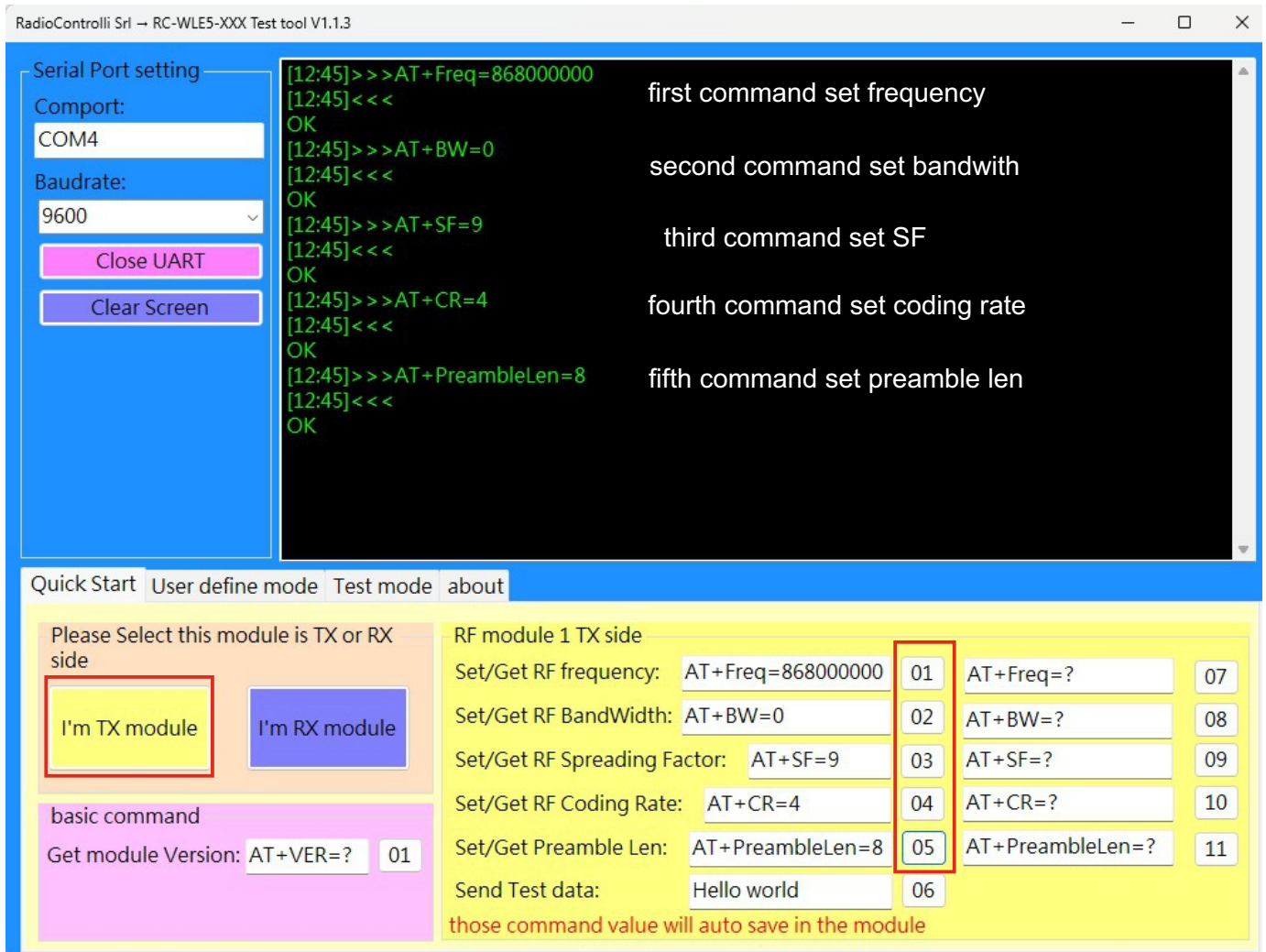
those command value will auto save in the module

(\*) The USB Dongle RadioControlli has inside the Silicon Labs CP2102 chip as serial/usb interface, to open the serial port will be necessary to use the Silicon Labs driver.

## Quick Start : Transmission Station-2

The following figure shows the setting page of Tx mode, and send all the 01-05 instruction in sequence once, that is to complete the module setting of TX .

- AT+Freq=868000000      set frequency to 868MHz
- AT+BW=0                    set Lora bandwidth to 125MHz
- AT+SF=9                    set Lora spreading factor to 9
- AT+CR=4                    set Lora coding rate to 4
- AT+PreambleLen=8      set PreambleLen to 8



RadioControlli S.r.l. → RC-WLE5-XXX Test tool V1.1.3

Serial Port setting  
 Comport: COM4  
 Baudrate: 9600  
 Close UART  
 Clear Screen

```
[12:45]>>>AT+Freq=868000000
[12:45]<<<
OK
[12:45]>>>AT+BW=0
[12:45]<<<
OK
[12:45]>>>AT+SF=9
[12:45]<<<
OK
[12:45]>>>AT+CR=4
[12:45]<<<
OK
[12:45]>>>AT+PreambleLen=8
[12:45]<<<
OK
```

first command set frequency  
 second command set bandwidth  
 third command set SF  
 fourth command set coding rate  
 fifth command set preamble len

Quick Start | User define mode | Test mode | about

Please Select this module is TX or RX side  
 I'm TX module | I'm RX module

basic command  
 Get module Version: AT+VER=? 01

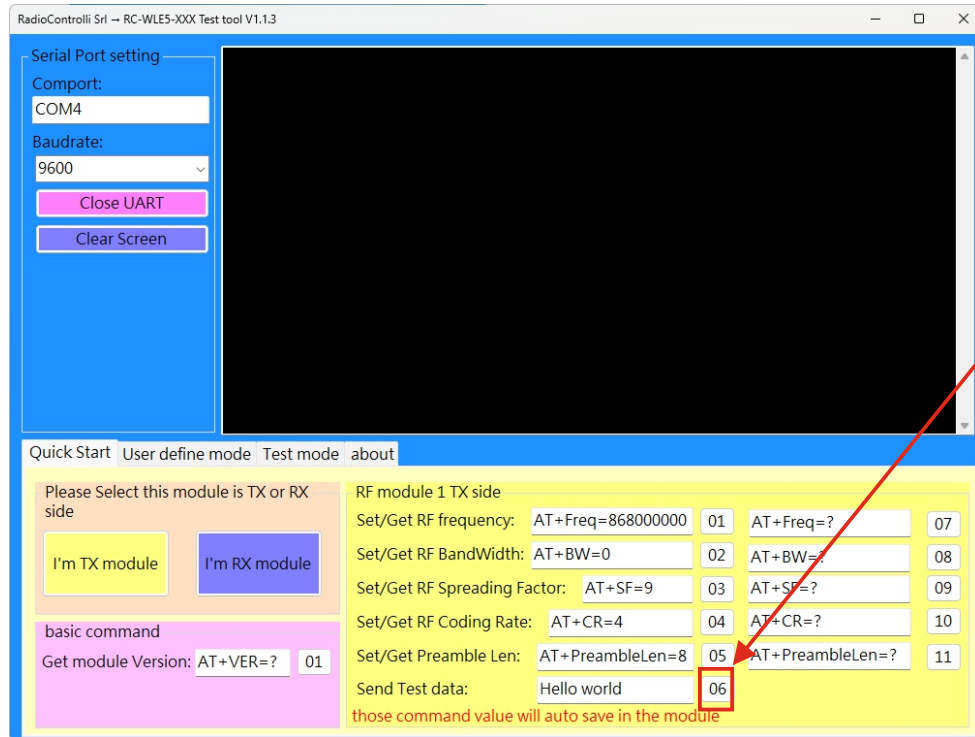
RF module 1 TX side

Set/Get RF frequency: AT+Freq=868000000	01	AT+Freq=?	07
Set/Get RF BandWidth: AT+BW=0	02	AT+BW=?	08
Set/Get RF Spreading Factor: AT+SF=9	03	AT+SF=?	09
Set/Get RF Coding Rate: AT+CR=4	04	AT+CR=?	10
Set/Get Preamble Len: AT+PreambleLen=8	05	AT+PreambleLen=?	11
Send Test data: Hello world	06		

those command value will auto save in the module

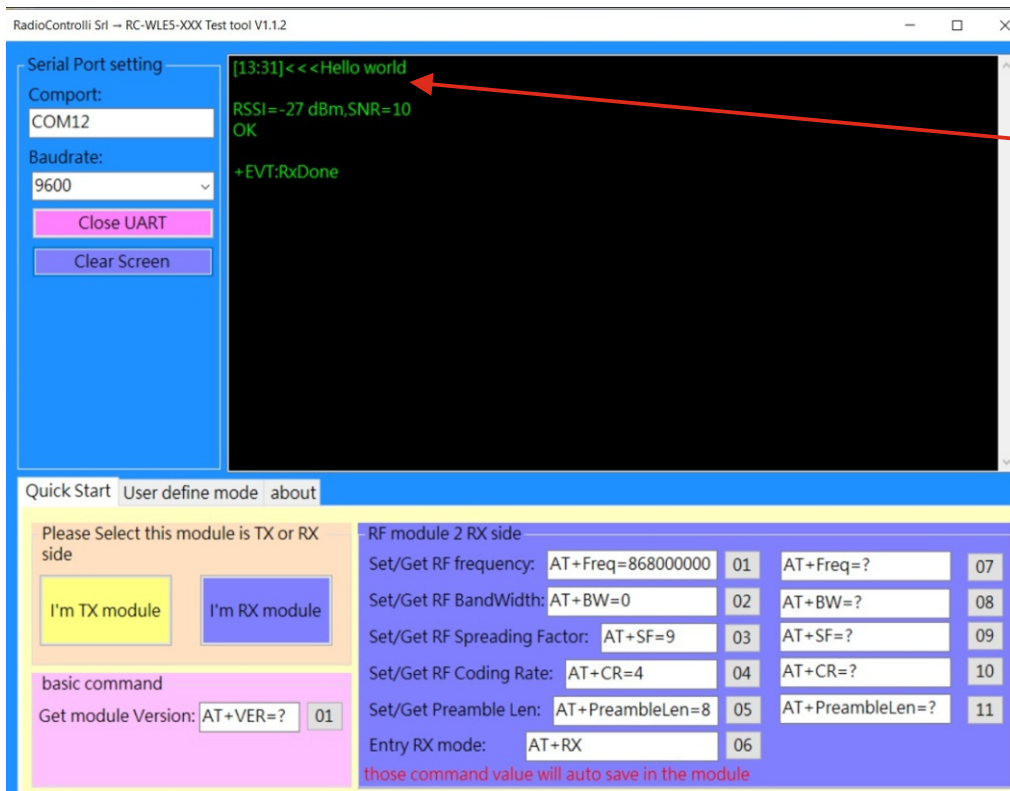
## Quick Start : First message sent

After all this sets, we are ready to send the first message :



**Transmission Station :**

Push this button



**Receiving Station :**

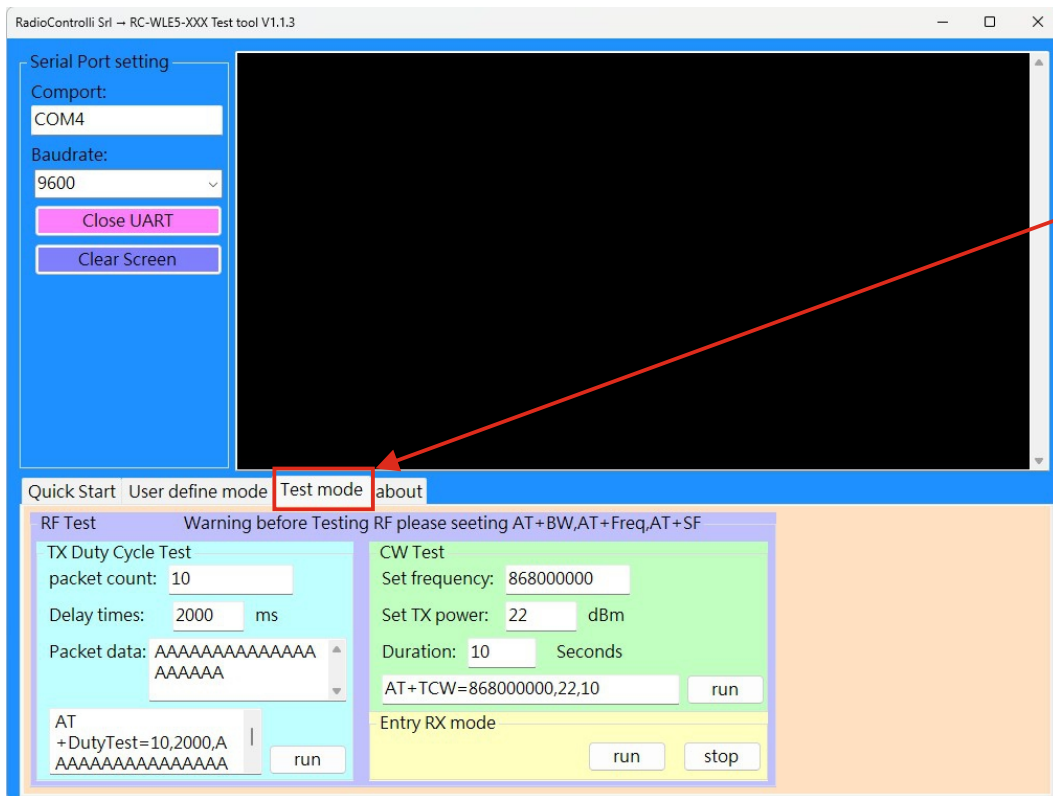
Message received

You can send other messages from the transmission station, changing the «Test data» and pushing the button

Send Test data: HELLO WORLD!!!! 06

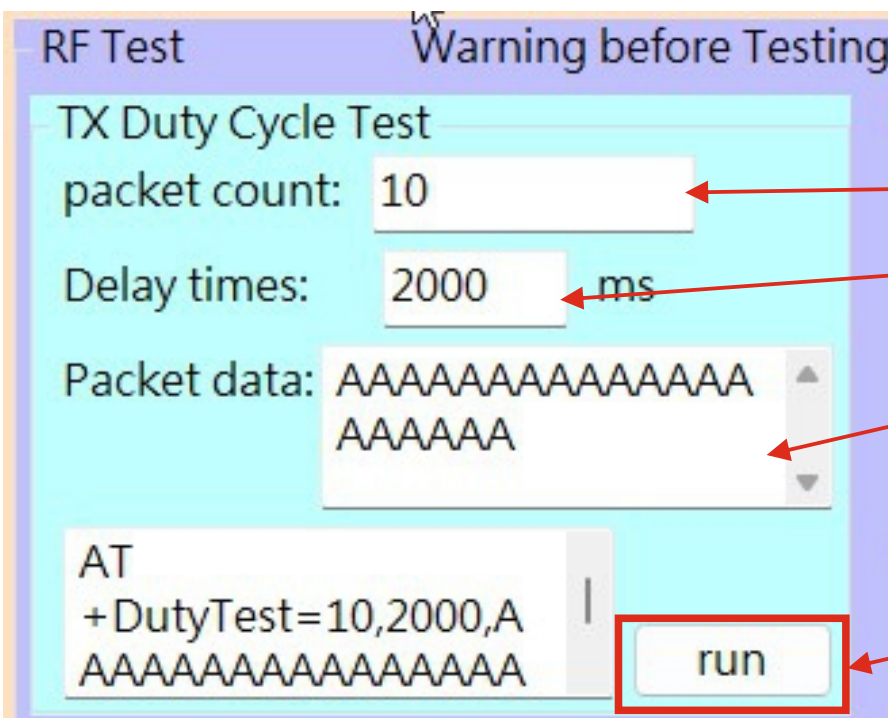
## Quick Start : Packets data sent continuously -1

This software also contains a "test mode" where it is possible to send a certain number of data packets spaced out by a delay to be defined.



**Transmission Station :**

Push this button



**Transmission Station :**

Entry number of the packets

Entry delay times between the packets

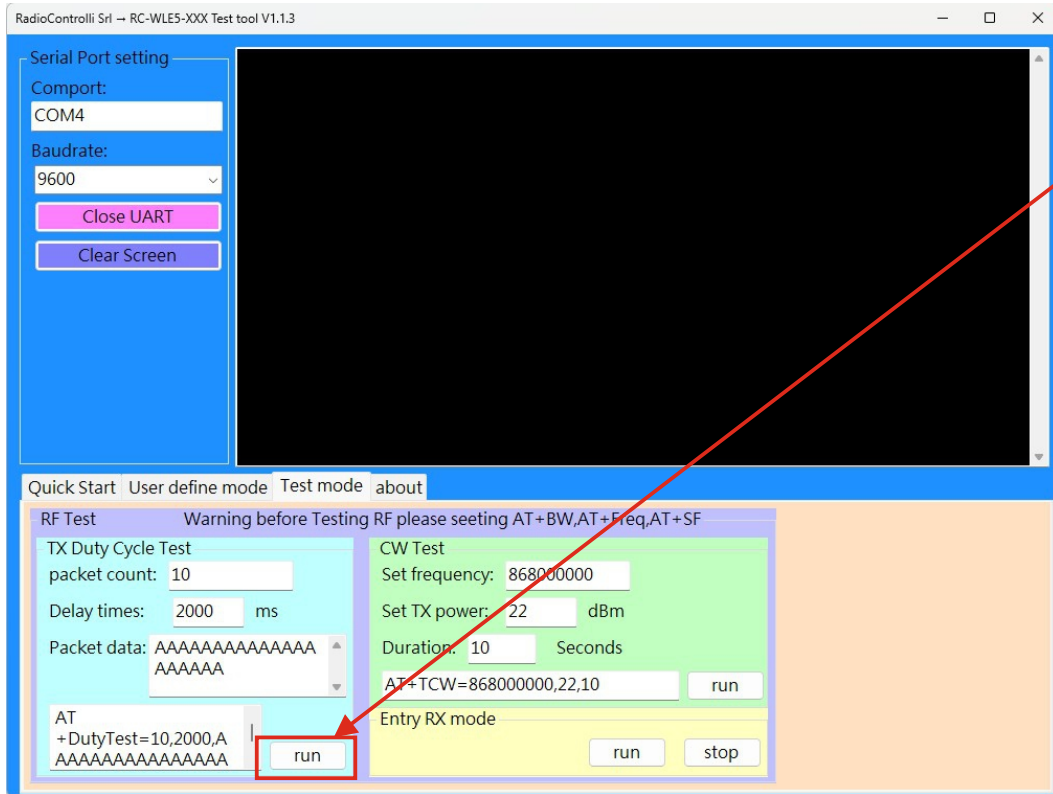
Entry data packet

Push this button to RUN the command

## Quick Start : Packets data sent continuously -2

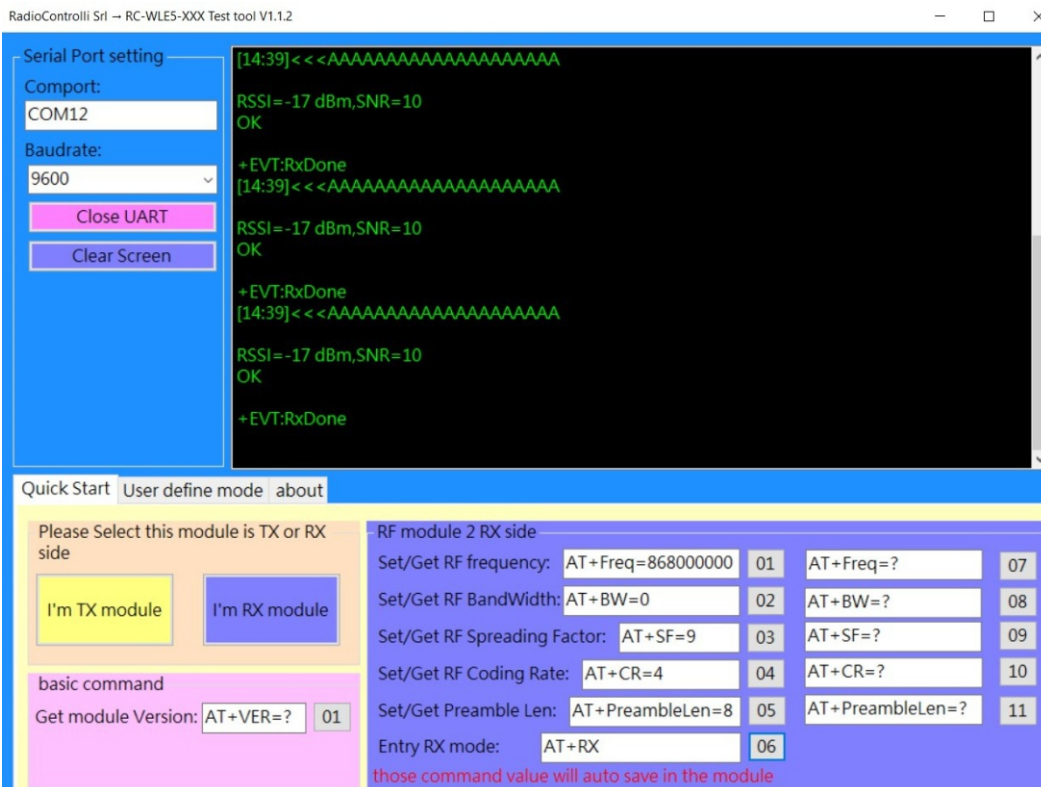
When the indicated button is pressed, the command will be executed and therefore the following will be transmitted:

Number = **10** packets data, contains :«**AAAAAAAAAAAAAAAAAAAA**» interspersed with **2000ms**.



**Transmission Station :**

## Receiving Station : Packets received



**Receiving Station :**

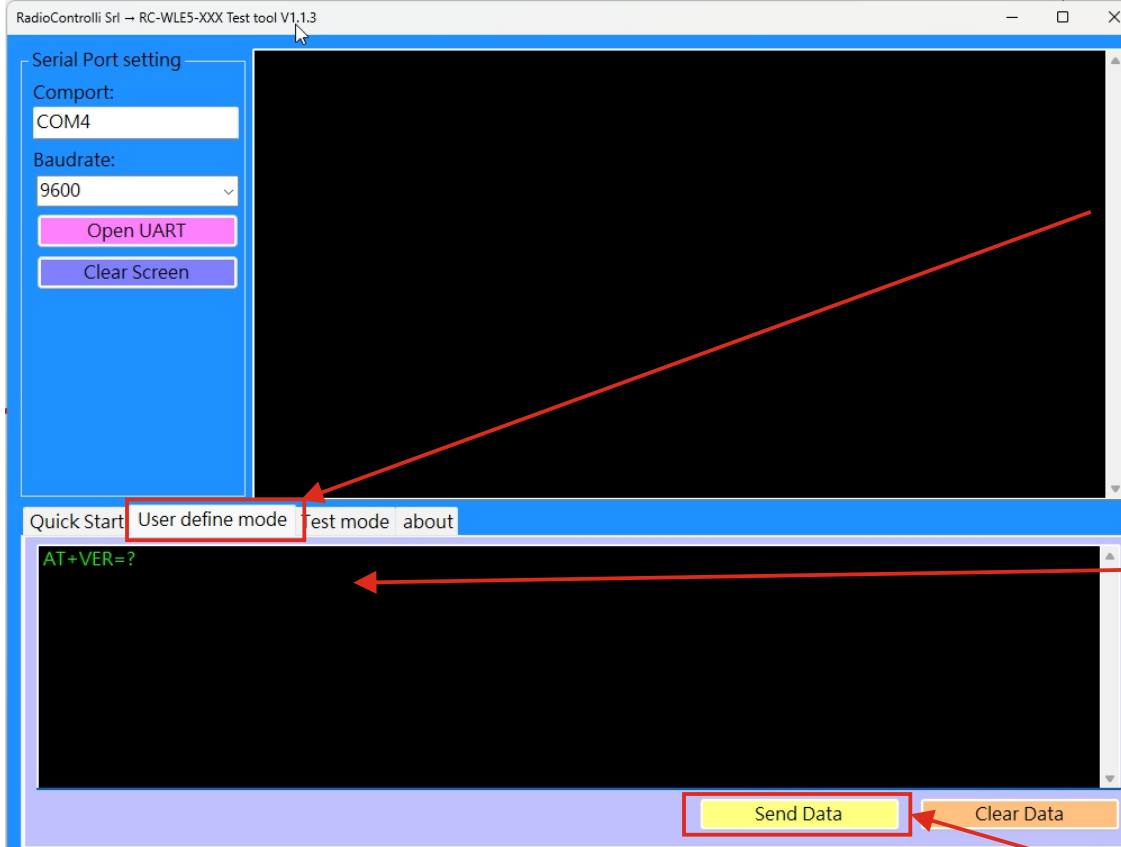
Packets received

## Quick Start : AT command send locally

This software also contains a "User define mode" where it is possible to send AT command locally.

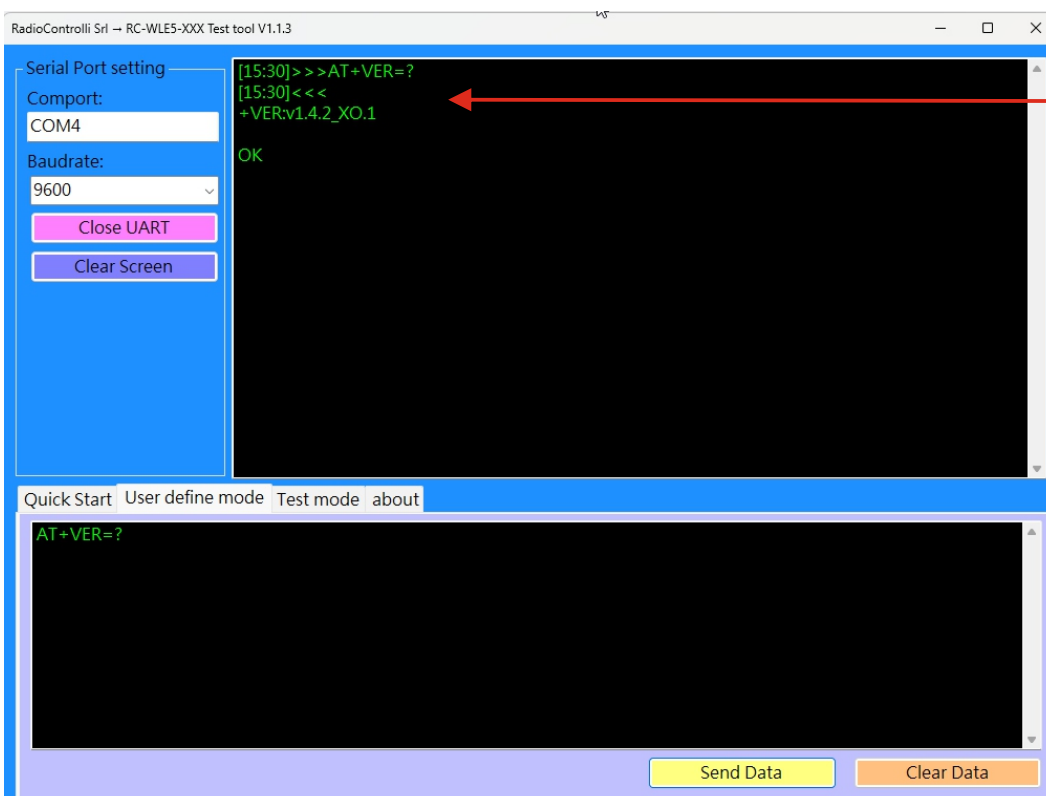
### Transmission or Receiver station

Push this button



It's possible to write the AT command in this AREA, in this case we are sending AT+VER=?

Push this button to Send Data



Answer received from the module.



**Quick Start : Evaluation Kit (RC-RICK-868-EV) composed by :**

N.2 USB LORA Dongle equipped with :

**-Dongle USB Interface**

(based on the CP2102 Silicon Lab USB/Serial)

**-RC-RICK-868 module**

(Hardware = RC-WLE5-868 with pre-installed internal software).

**- Cable UFL-SMA**

**- RC-ANT-868-SMA**

(Antenna 868MHz with SMA connector)



## X-ON Electronics

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for [RF Modules](#) category:*

*Click to view products by [Radiocontrolli](#) manufacturer:*

Other Similar products are found below :

[2221706-1](#) [RAD-900-DAIO6](#) [650200791G](#) [650200714G](#) [RFM95PW-915S2](#) [VG2389S433N0S1](#) [VG6244S580X0M1](#) [RC-WLE5-868-HA](#)  
[RFM219BW-868S2](#) [E49-900M20S](#) [E01-2G4M27SX](#) [DL-297LDA-S](#) [DL-24PA-C](#) [DL-Si24R1-A](#) [VG3411S433N0S1](#) [HY831S](#) [TR1101](#)  
[VG4130S433N0S1](#) [VG4142T868N0M1](#) [VG3411S915N0S1](#) [E310-433T23S](#) [E43-900T13S3](#) [E07-400MM10S](#) [DL-PAN3029-S-915S](#) [DL-](#)  
[PAN3029-S-433S](#) [E01C-2G4M11S](#) [DL-PAN3029-S-868S](#) [E70-433MT14S](#) [E03-2G4M10S](#) [E01C-ML01S](#) [E610-433T20S](#) [E610-](#)  
[DTU\(433C30\)](#) [E01-2G4M13S](#) [E30-400M20S\(4463\)](#) [E49-400M30S](#) [E31-230T27D](#) [E41-400M20S](#) [E07-900MM10S](#) [E01-2G4M20S1B](#) [E70-](#)  
[900T14S](#) [E01C-ML01D](#) [E49-400T30S](#) [E30-170T27D](#) [E62-433T30D](#) [E21-400G30S](#) [E07-900T10S](#) [E04-400M16S](#) [E01-ML01DP4](#) [E610-](#)  
[433T30S](#) [E29-400M22S](#)