

RN2903A

LoRaWAN Getting start

REV 1.0



Prerequisites

Hardware Prerequisites

- Notebook PC
- Board Part Number: DM164139

Software Prerequisites

- Teraterm
- MS windows 10,8,7

Estimated Completion Time: 3 – 3.5 Hours

REV history:

- Initial version 1.0. Date: 1 JUN 18

Table of Contents

Setup Teraterm

Assignment 1: Setup mote and Register on Gateway

- OTAA
- ABP
- Frequency, Duty cycle, Data Rate (20-30 mins)

Assignment 2: Class A, Test Mote Board with Button (20-30 mins)

Assignment 3: Class A command (20-30 mins)

Assignment 4: Class C command (30-45 mins)

Assignment 5: PORT GPIO (20-30 mins)

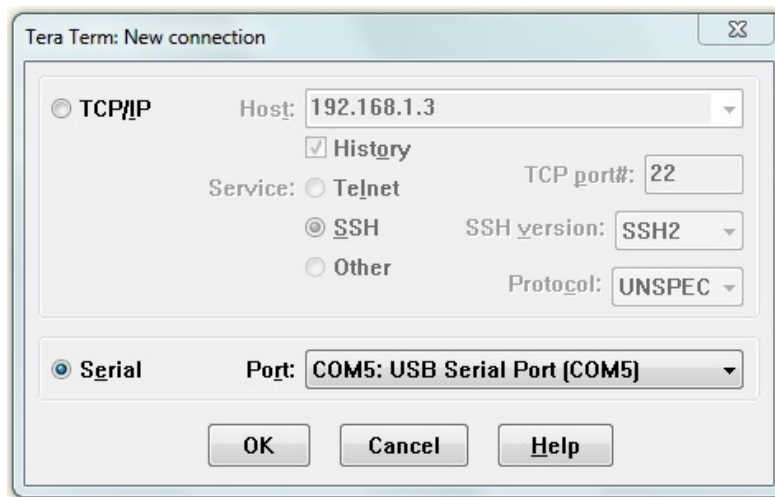
Assignment 6: P2P (20-30 mins)

Summary /Additional information:

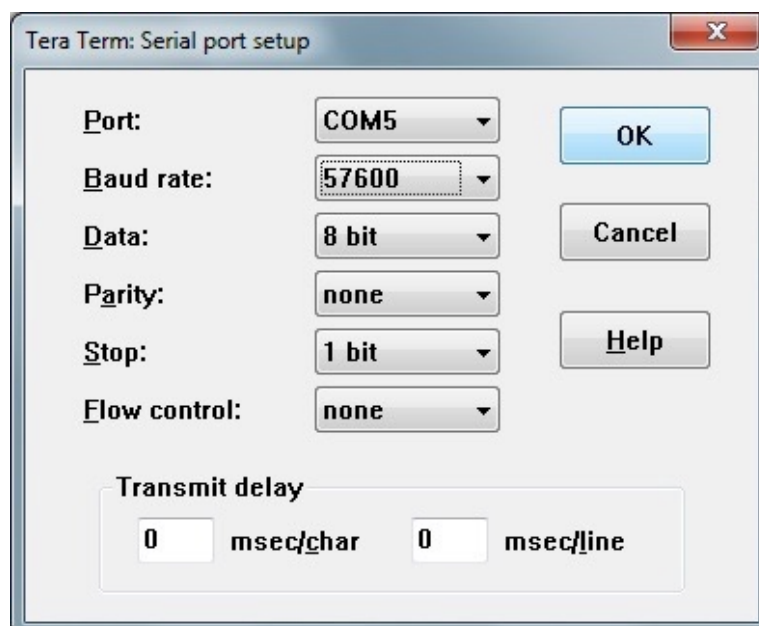
Install and setup TeraTerm



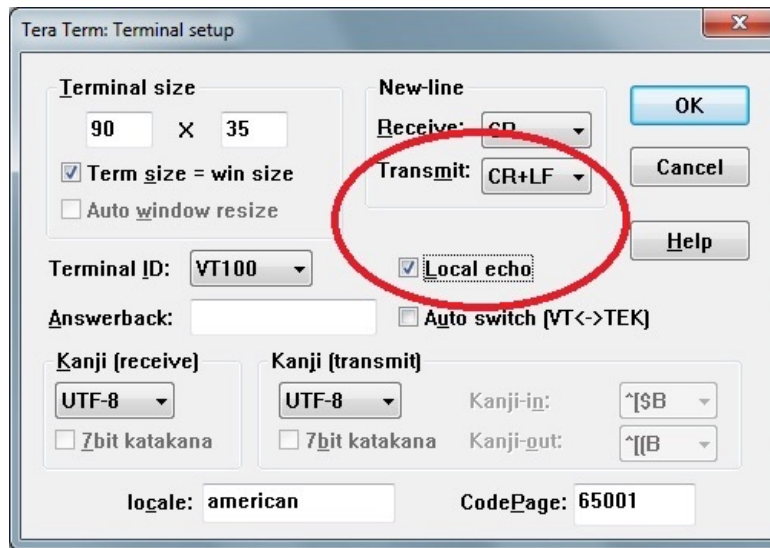
1) Setup comport number.



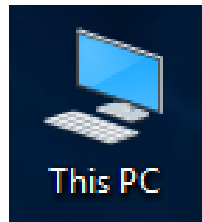
2) Baud Rate setting.



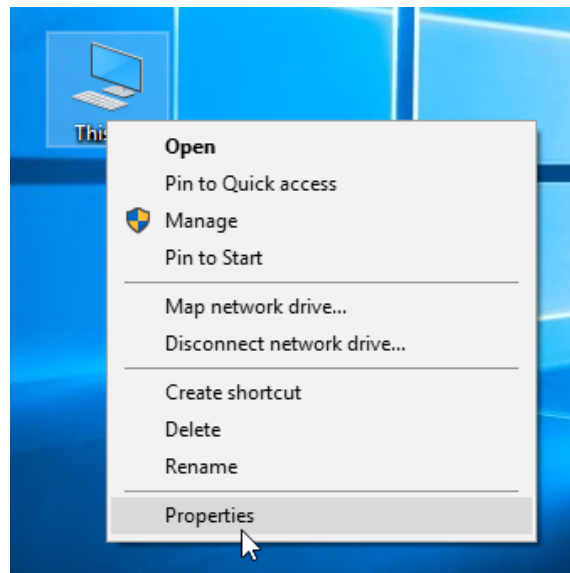
3) Terminal Setup



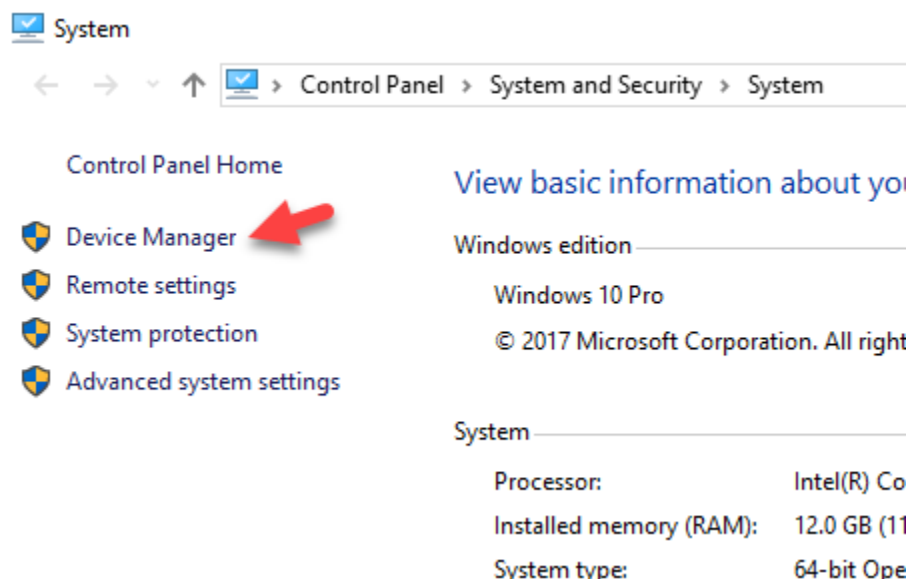
How to find COMPORT No.



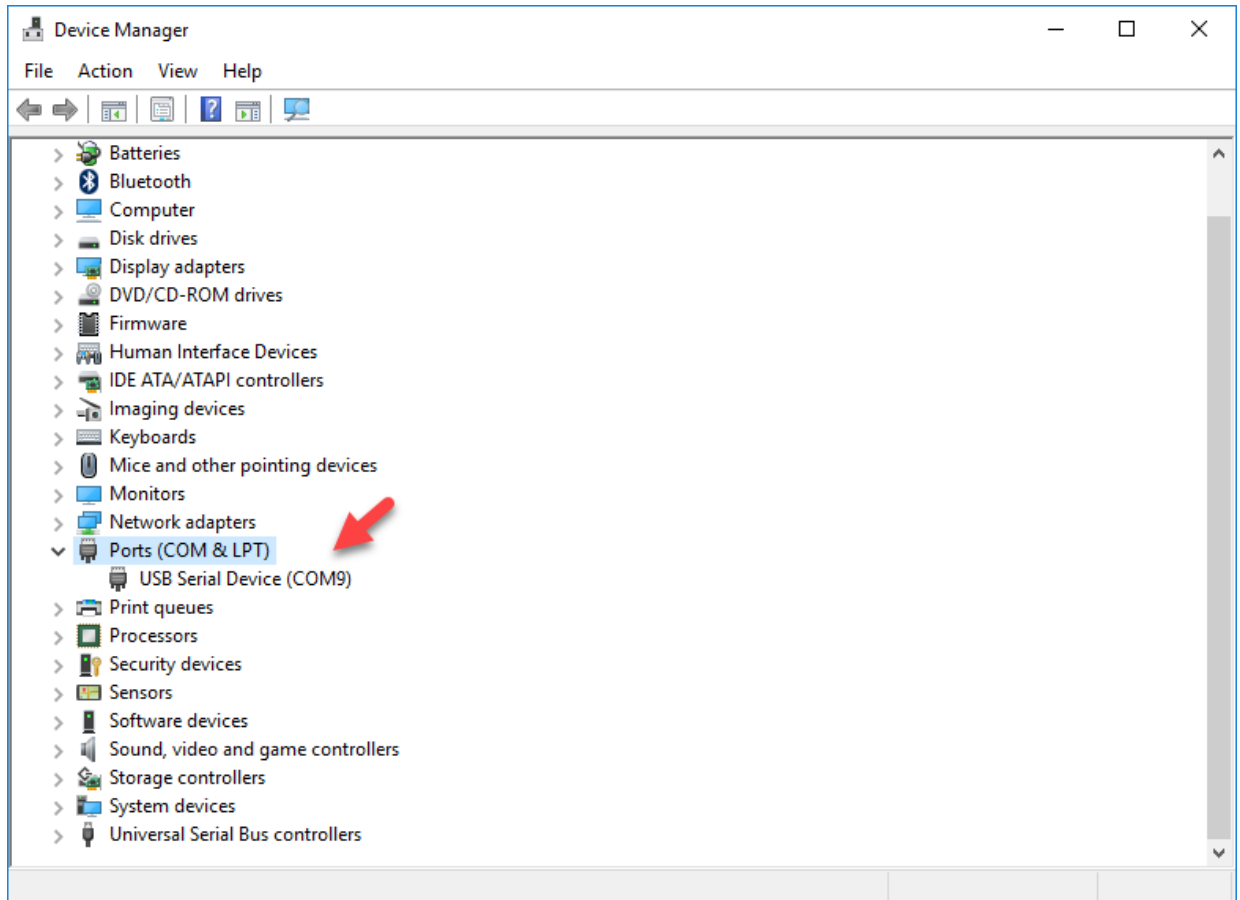
1) Right click on This PC and Select Properties.



2) Select Device Manager



3) On Device Manager , See on Ports (COM & LPT)



Assignment 1: Setup mote and Register on Gateway

1) Recheck Mote firmware

```
sys get ver
```

Response: RN2903 X.Y.Z MMM DD YYYY HH:MM:SS, where X.Y.Z is the firmware version, MMM is month, DD is day, HH:MM:SS is hour, minutes, seconds (format: [HW] [FW] [Date] [Time]). [Date] and [Time] refer to the release of the firmware.

This command returns the information related to the hardware platform, firmware version, release date and time-stamp on firmware creation.

2) System Reset

```
sys reset
```

Response: RN2903 X.Y.Z MMM DD YYYY HH:MM:SS, where X.Y.Z is the firmware version, MMM is month, DD is day, HH:MM:SS is hour, minutes, seconds (format: [HW] [FW] [Date] [Time]). [Date] and [Time] refer to the release of the firmware.

This command resets and restarts the RN2903 module; stored internal configurations will be loaded automatically upon reboot.

Over-the-Air Activation (OTAA)

- Device EUI (`dev_eui`)
- Application EUI (`app_eui`)
- Application key (`app_key`)

*** Note: Don't forget mac save

mac set deveui <devEUI>

<devEUI>: 8-byte hexadecimal number representing the device EUI

Response: `ok` if address is valid

`invalid_param` if address 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 that can be retrieved using the `sys get hweui` command (see [Section 2.3.6.4](#)).

Alternatively, a user provided EUI can be configured using the `mac set deveui` command.

mac set appeui <appEUI>

<appEUI>: 8-byte hexadecimal number representing the application EUI

Response: `ok` if address is valid

`invalid_param` if address is not valid

This command sets the application identifier for the module. The application identifier should be used to identify device types (sensor device, lighting device, etc.) within the network.

mac set appkey <appKey>

<appKey>: 16-byte hexadecimal number representing the application key

Response: `ok` if address is valid

`invalid_param` if address 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.

mac save

Response: `ok`

The `mac save` command must be issued after configuration parameters have been appropriately entered from the `mac set <cmd>` commands. This command will save LoRaWAN protocol configuration parameters to EEPROM. Upon the next system reset the LoRaWAN protocol configuration will be initialized with the last saved parameters. The system may reset by power cycling or a pulse on the `MCLR` pin as well as by using `sys reset`.

Example:

```
mac set deveui 1122334455667788
mac set appeui 1122334455667788
mac set appkey 10002000300040005000600070008000
mac save
```

ABP (Activation By Personalisation)

- Device Address (dev_addr)
- Network Session Key (nwk_skey)
- Application Session Key (app_skey)

mac set devaddr <address>

<address>: 4-byte hexadecimal number representing the device address, from 00000000 – FFFFFFFF

Response: ok if address is valid
invalid_param if address is not valid

This command configures the module with a 4-byte unique network device address <address>. The <address> *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.

mac set nwkskey <nwksesskey>

<nwkSessKey>: 16-byte hexadecimal number representing the network session key

Response: ok if address is valid
invalid_param if address 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.

mac set appskey <appSesskey>

<appSessKey>: 16-byte hexadecimal number representing the application session key

Response: ok if address is valid

invalid_param if address 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.

Example:

```
mac set devaddr ABCD0004
```

```
mac set nwkskey 10002000300040005000600070008000
```

```
mac set appskey 10002000300040005000600070008000
```

```
mac save
```

Note: All number in OTAA and ABP setting is hex base, 0x0 to 0xF.

Set Frequency, Duty cycle, Data Rate

Channel	Ch Freq	Command
0	923.2MHz	923200000
1	923.4MHz	923400000
2	923.0MHz	923000000
3	922.8MHz	922800000
4	921.6MHz	921600000
5	921.8MHz	921800000
6	922.0MHz	922000000
7	922.2MHz	922200000

Example Command

Radio 0: 923100000

Radio 1: 921900000

Enable all multiSF channels

Channel MultiSF 0: radio 0 100000 (mandatory channel 923.2)

Channel MultiSF 1: radio 0 300000 (mandatory channel 923.4)

Channel MultiSF 2: radio 0 -100000

Channel MultiSF 3: radio 0 -300000

Channel MultiSF 4: radio 1 -300000

Channel MultiSF 5: radio 1 -100000

Channel MultiSF 6: radio 1 100000

Channel MultiSF 7: radio 1 300000

Example

Channel 2:

```
mac set ch freq 2 923000000 // Sets ch2 to 923.000MHz (Same channels set in the gateway)
mac set ch dcycle 2 999 // Sets the duty cycle to 0.1% (= 99.9% off)
mac set ch drrange 2 0 5 // Allows data rates 0 to 5 on this channel
mac set ch status 2 on // Enables the channel
```

Channel 3:

```
mac set ch freq 3 922800000 // Sets ch3 to 922.800MHz (Same channels set in the gateway)
mac set ch dcycle 3 999 // Sets the duty cycle to 0.1% (= 99.9% off)
mac set ch drrange 3 0 5 // Allows data rates 0 to 5 on this channel
mac set ch status 3 on // Enables the channel
```

Note: Red character is different on channel and frequency.

Teraterm can support scrip running.

Result:

```
mac set ch freq 2 923000000
mac set ch freq 3 922800000
mac set ch freq 4 921600000
mac set ch freq 5 921800000
mac set ch freq 6 922000000
mac set ch freq 7 922200000
mac set ch dcycle 2 999
mac set ch drrange 2 0 5
mac set ch status 2 on
mac set ch dcycle 3 999
mac set ch drrange 3 0 5
mac set ch status 3 on
mac set ch dcycle 4 999
mac set ch drrange 4 0 5
mac set ch status 4 on
mac set ch dcycle 5 999
mac set ch drrange 5 0 5
mac set ch status 5 on
mac set ch dcycle 6 999
mac set ch drrange 6 0 5
mac set ch status 6 on
mac set ch dcycle 7 999
mac set ch drrange 7 0 5
mac set ch status 7 on
mac save
```


RN2903A LAB Manual REV 1.0 LoRAWAN

Assignment 2: Class A, Test Mote Board with Button



NOTE: For MOTE is connected by USB cable.

1. Push S2
2. Push S3

If push S2 at first time, will go to Testing mode.



Select OTAA to test

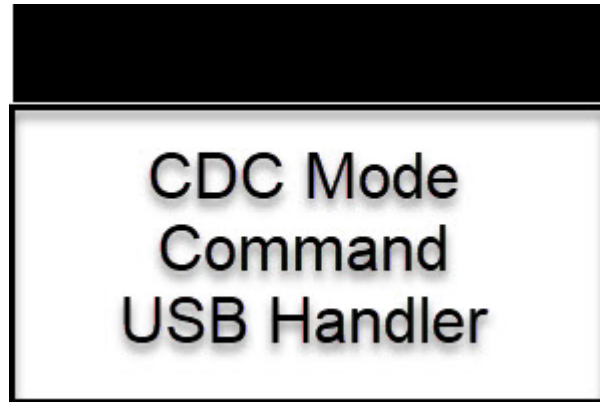


Select (S3) and (SEL) S3



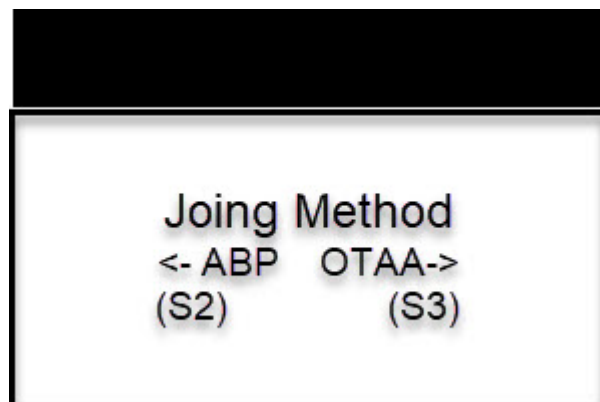
SELECT CNF and Wait (DATA Transmitted)

Mote Board Menu: (Additional menu)



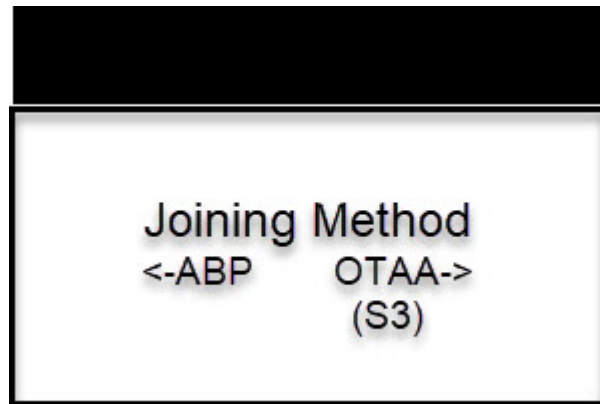
Pic 1.

After plug usb cable to Mote board, press S2 to go to Joining



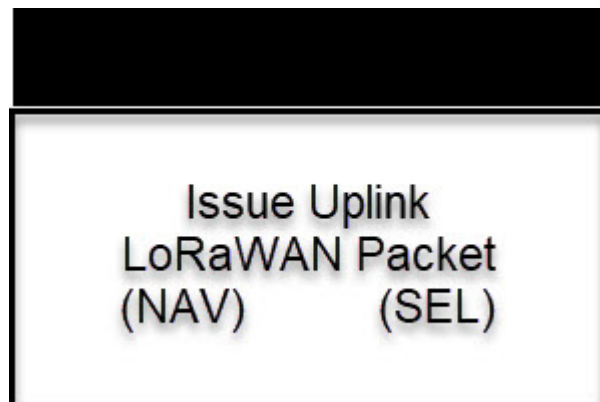
Pic 2.

For example, press S3 (OTAA joining)



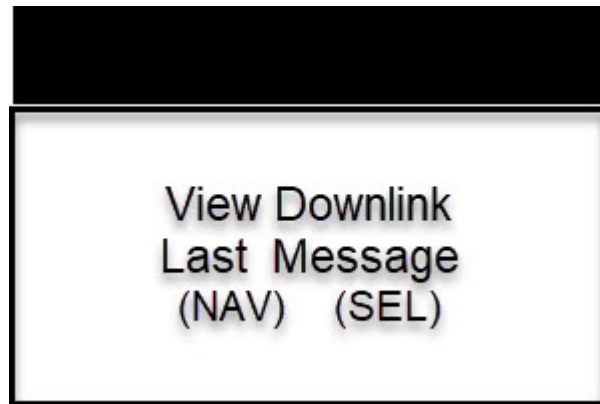
Pic3.

Observe (S3), It is marking and show selecting OTAA.



Pic4.

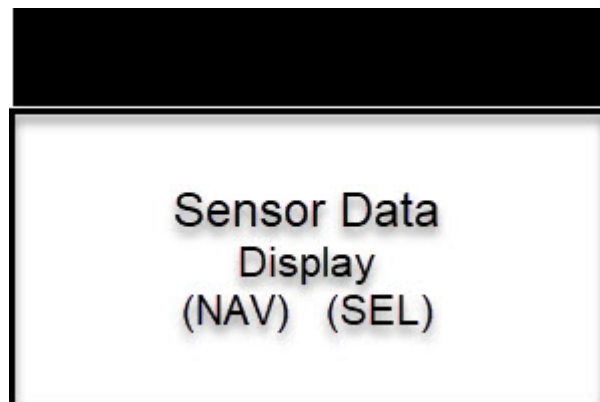
Pic4, menu shows NAV and SEL. If press "SEL", will send temp and light message to gateway. If press NAV, will go to other any many.



Pic 5.

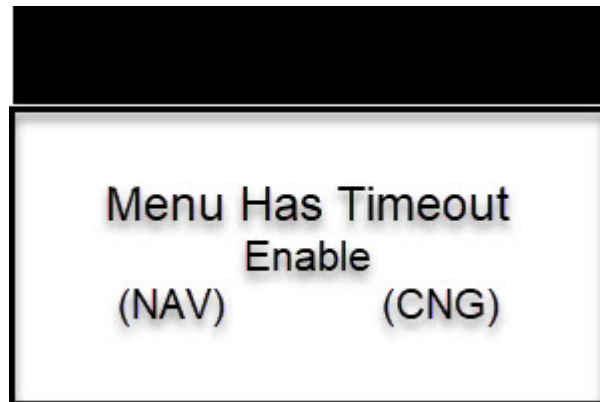
Pic5, You press NAV at pic4, shows downlink last message.

If set some message at network server, after uplink will shows downlink message at this menu, press (SEL).



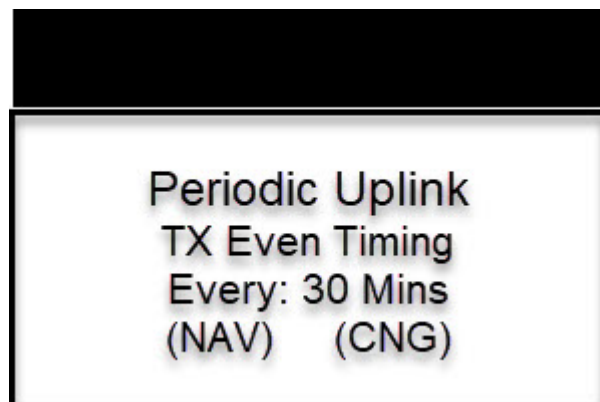
Pic 6.

Pic6, shows Sensor data on mote board.



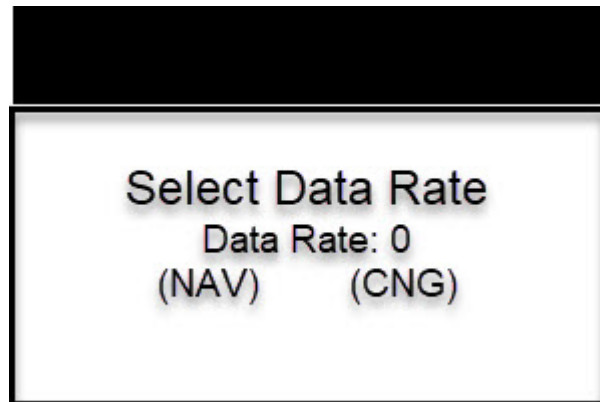
Pic 7.

Pic7 shows time out of menu if use battery operating.



Pic 8.

Pic8 shows Tx or uplink message period.



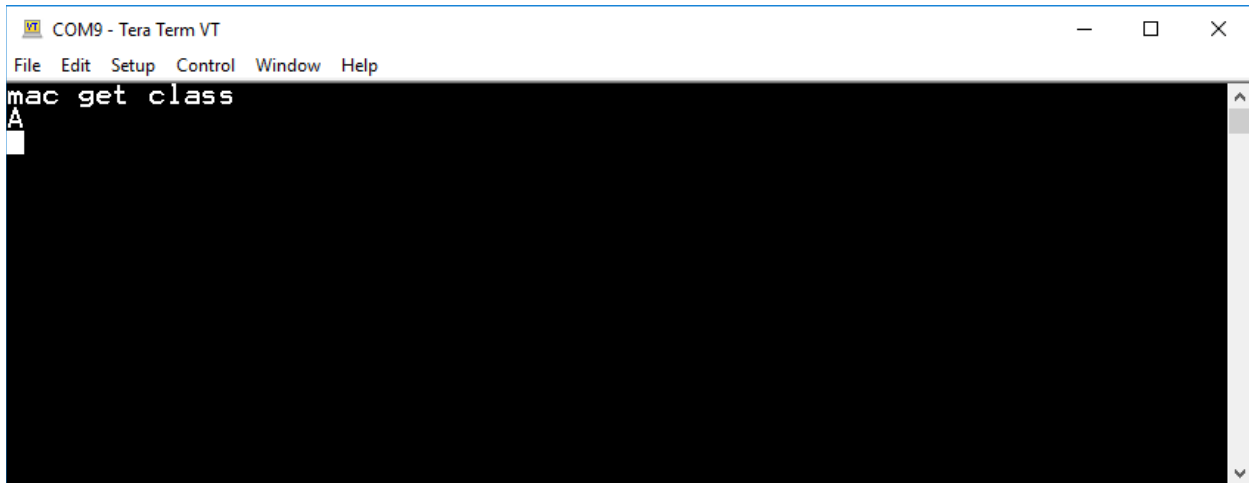
Pic 9.

Pic 9 shows Data Rate for setting. If need to set Data Rate, press (CNG). If no need to set, press (NAV), will go to Uplink menu.

Assignment 3: Class A command

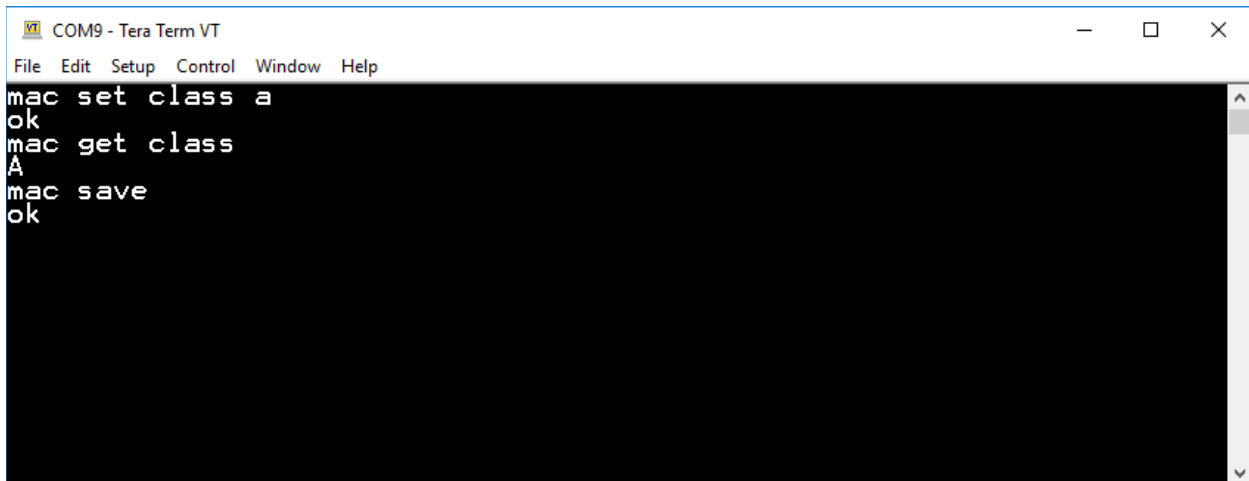
Before any joining, please recheck mote class at first time.

mac get class



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac get class
A
```

mac set class



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac set class a
ok
mac get class
A
mac save
ok
```

Note: After flash fw on mote module, will be set Class A (default).

Join OTAA and Tx (CLASS A)

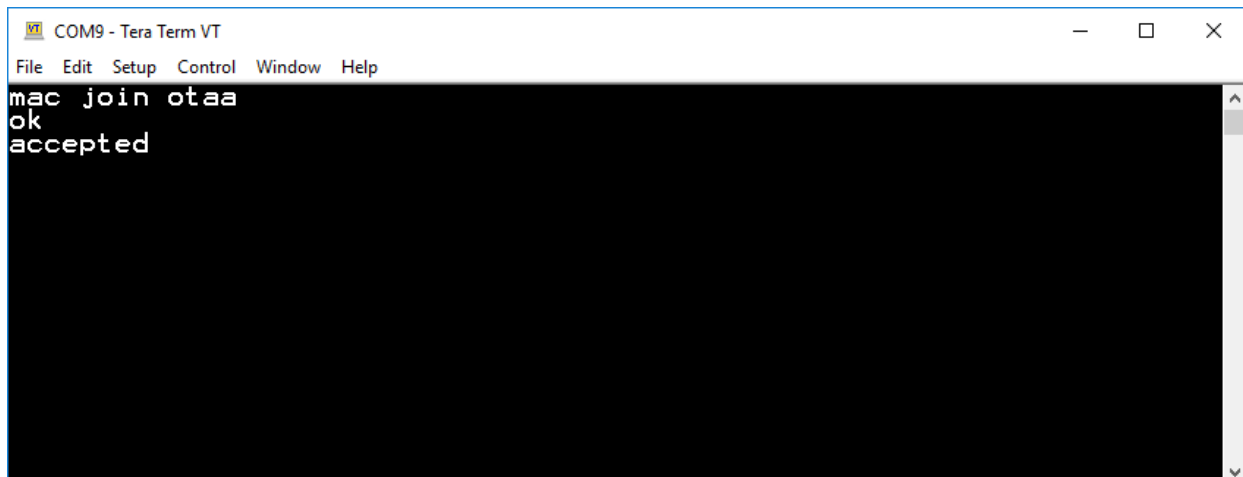
Assignment: Join OTAA and Tx.

Send confirm : ABCD01

Send unconfirm: FFAABBEE

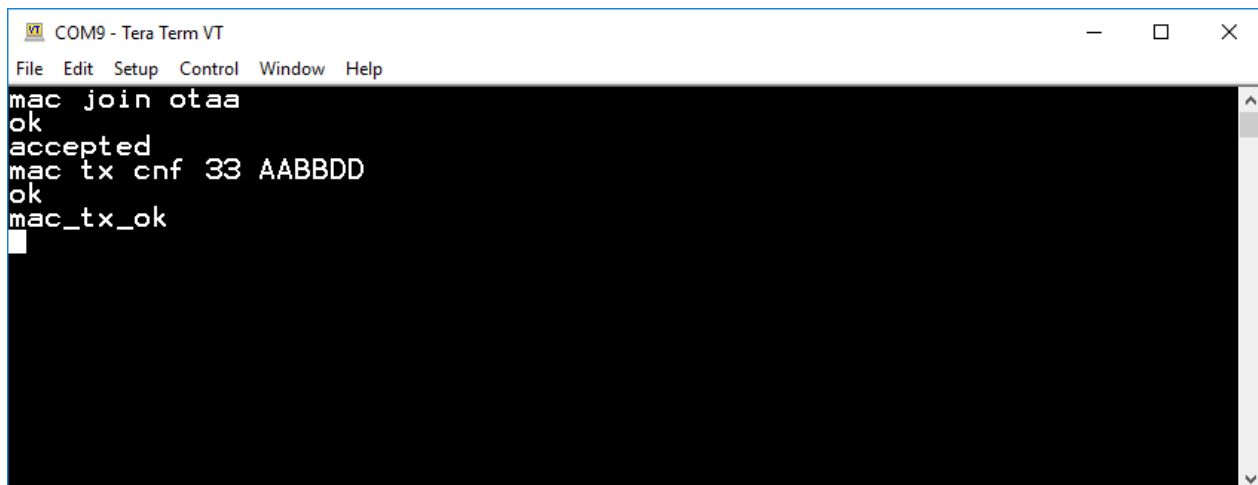
Guide:

- mac join OTAA
- mac tx cnf port XXXX
- mac tx uncnf port YYYY



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join otaa
ok
accepted
```

Send AABBD on port 33 (Confirm)



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join otaa
ok
accepted
mac tx cnf 33 AABBD
ok
mac_tx_ok
```

Send ACAC on port 43 (Unconfirm)

```

COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join otaa
ok
accepted
mac tx uncnf 43 ACAC
ok
mac_tx_ok

```

Note: Observe , mac_tx_ok will show unconfirming and confirming if has no any error message at network server.

Tx confirm and Tx unconfirm on network server.

Received data													DOWNLOAD
Id	Endpoint ID	Received time	Sequence number	Port	Radio ID	Channel	SNR	RSSI	Frequency	Modulation	Data Rate	Coding rate	Payload HEX
251	4A30B000000004	10/26/2017 11:37:34 PM	0	43	0	5	9.2 dB	-35 dBm	921.800 MHz	LoRa	SF12BW125	4/5	ACAC
250	4A30B000000004	10/26/2017 11:33:34 PM	0	33	0	4	7.8 dB	-34 dBm	921.600 MHz	LoRa	SF12BW125	4/5	AABBDD

Join ABP and Tx (CLASS A)

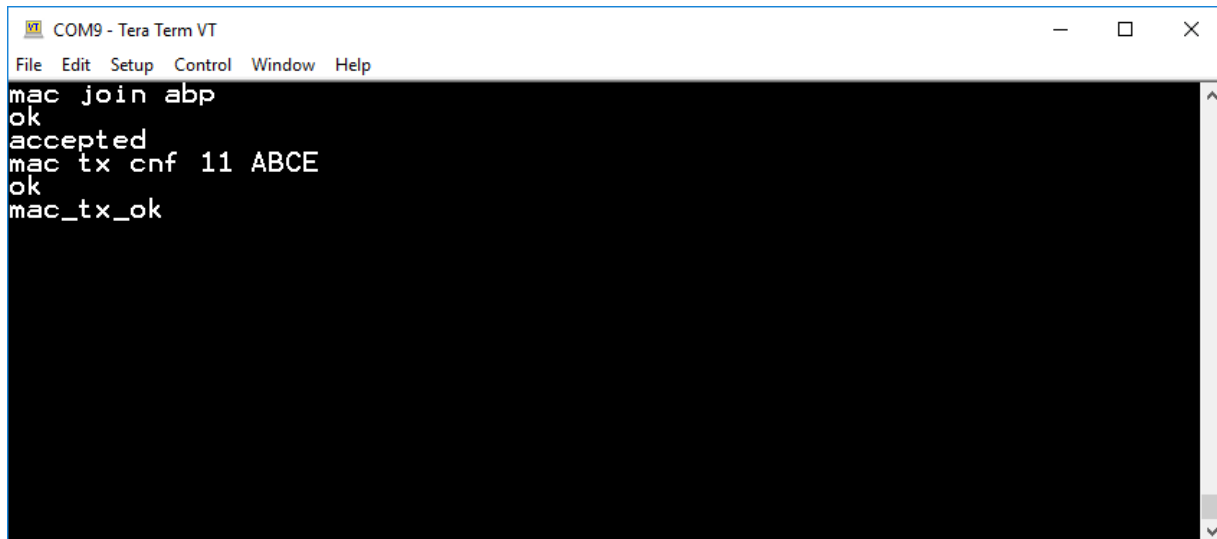
Assignment: Join ABP and Tx.

Send confirm: ABCD01

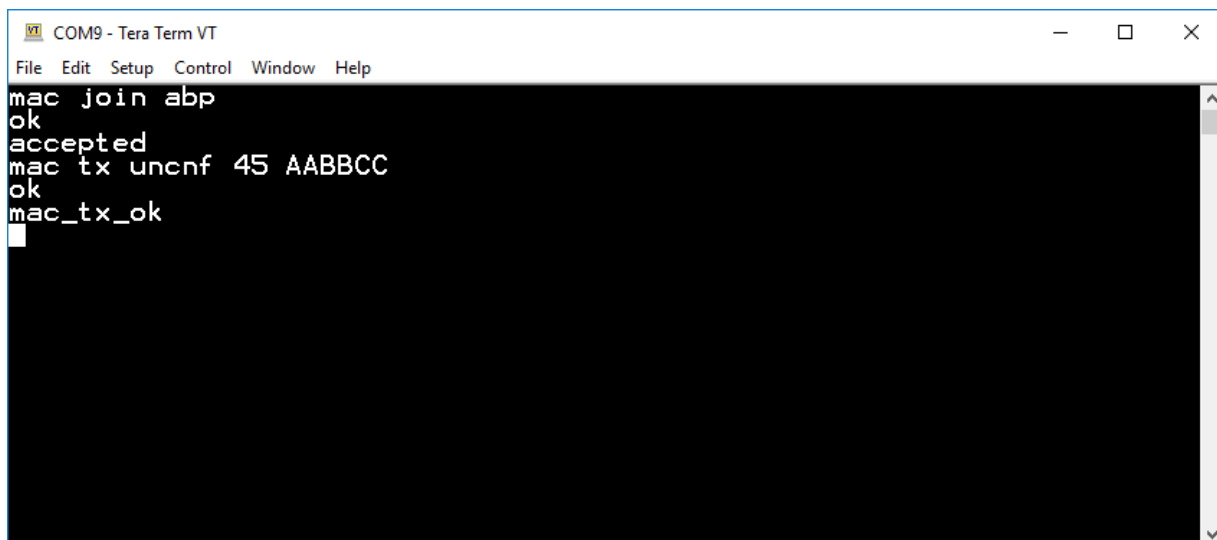
Send unconfirm: FFAABBEE

Guide:

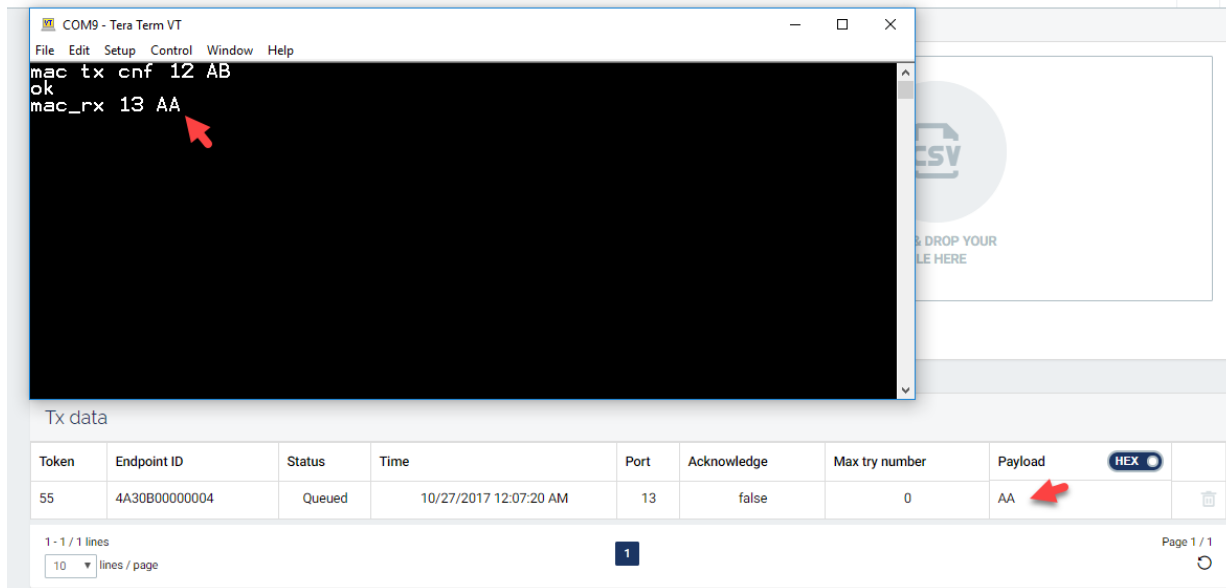
- mac join abp
- mac tx cnf port XXXX
- mac tx uncnf port YYYY



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join abp
ok
accepted
mac tx cnf 11 ABCE
ok
mac_tx_ok
```



```
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join abp
ok
accepted
mac tx uncnf 45 AABBC
ok
mac_tx_ok
```



COM9 - Tera Term VT

```
File Edit Setup Control Window Help
mac tx cnf 12 AB
ok
mac_rx 13 AA
```

Tx data

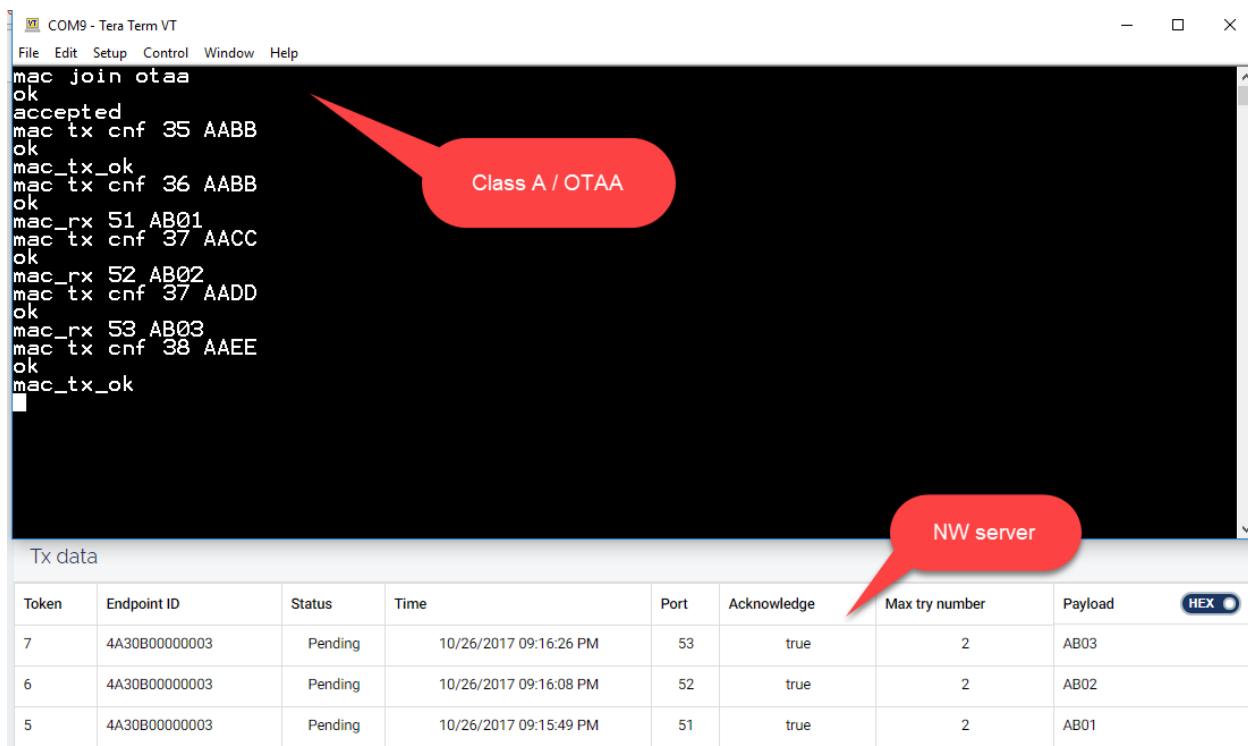
Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
55	4A30B00000004	Queued	10/27/2017 12:07:20 AM	13	false	0	AA	

1 - 1 / 1 lines
10 lines / page

Note: Class A download link as AA via port 13

Class A will send Tx at 1 time and get Rx then go to sleep.

Example: Test class A, Tx at 3 times and Rx at 3 times.



COM9 - Tera Term VT

```
File Edit Setup Control Window Help
mac join otaa
ok
accepted
mac tx cnf 35 AABB
ok
mac_tx_ok
mac tx cnf 36 AABB
ok
mac_rx 51 AB01
mac tx cnf 37 AACC
ok
mac_rx 52 AB02
mac tx cnf 37 AADD
ok
mac_rx 53 AB03
mac tx cnf 38 AEAE
ok
mac_tx_ok
```

Tx data

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
7	4A30B00000003	Pending	10/26/2017 09:16:26 PM	53	true	2	AB03	
6	4A30B00000003	Pending	10/26/2017 09:16:08 PM	52	true	2	AB02	
5	4A30B00000003	Pending	10/26/2017 09:15:49 PM	51	true	2	AB01	

Assignment 4: Class C command

Guide:

mac get class

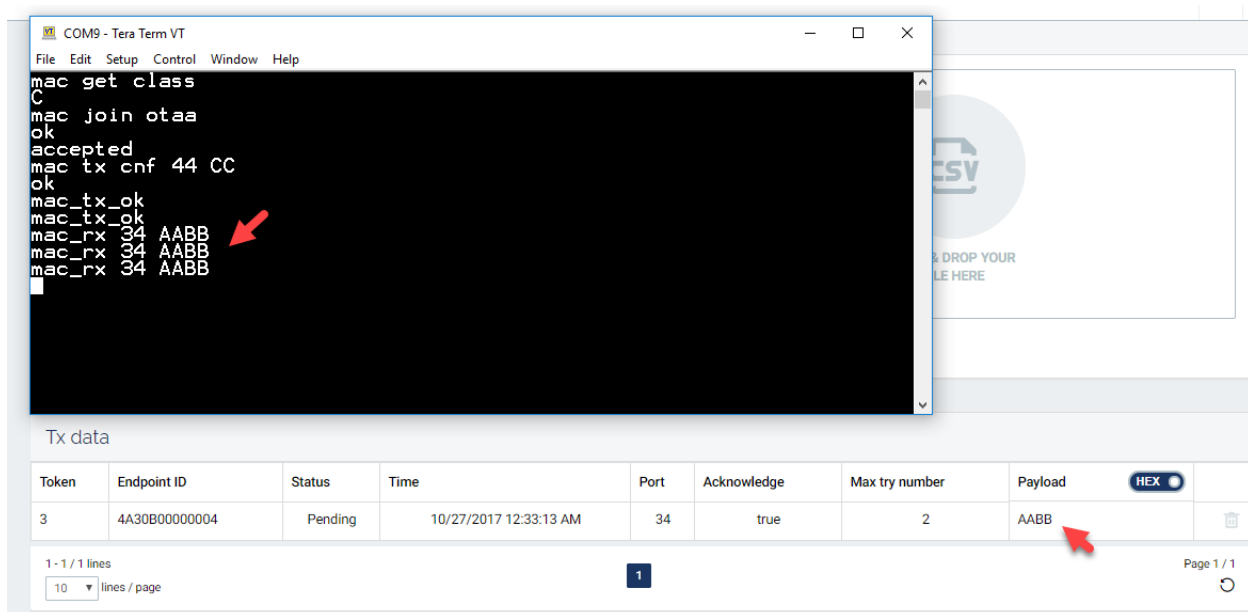
mac set class c

mac join otaa

mac tx cnf port XXXX

mac tx uncnf port YYYY

Join OTAA: CLASS C



The screenshot shows a Tera Term VT terminal window with the following commands and responses:

```
mac get class
C
mac join otaa
ok
accepted
mac tx cnf 44 CC
ok
mac_tx_ok
mac_tx_ok
mac_rx 34 AABB
mac_rx 34 AABB
mac_rx 34 AABB
```

A red arrow points to the 'mac_rx 34 AABB' line in the terminal output.

Below the terminal, the 'Tx data' table is displayed:

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
3	4A30B00000004	Pending	10/27/2017 12:33:13 AM	34	true	2	AABB	

A red arrow points to the 'AABB' payload value in the table. The 'HEX' toggle is currently turned on.

Page 1 / 1

RN2903A LAB Manual REV 1.0 LoRAWAN



The screenshot shows a Tera Term VT terminal window with the following commands and responses:

```
mac get class
C
mac join otaa
ok
accepted
mac tx cnf 44 CC
ok
mac_tx_ok
mac_tx_ok
mac_rx_34 AABB
mac_rx_34 AABB
mac_rx_34 AABB
mac_rx_12 EFFF
mac_rx_55 DFFF
mac_rx_66 AFFFFF
mac_rx_12 EEAA
```

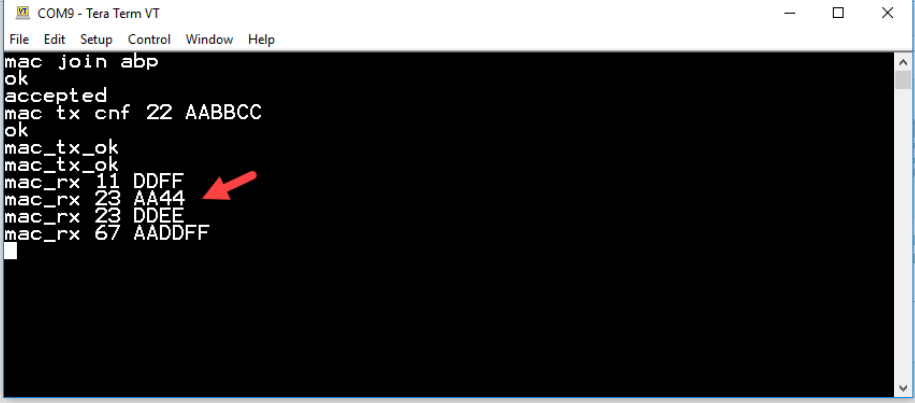
Below the terminal window is a 'Tx data' table with the following data:

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
7	4A30B00000004	Sent	10/27/2017 12:37:15 AM	12	false	0	EEAA	

At the bottom of the screenshot, there is a status bar showing '1 - 1 / 1 lines', a dropdown menu set to '10 lines / page', a page number '1', and 'Page 1 / 1'.

Note: Class C, RX will receive data from Gateway until next Tx again.

Join ABP: CLASS C



```

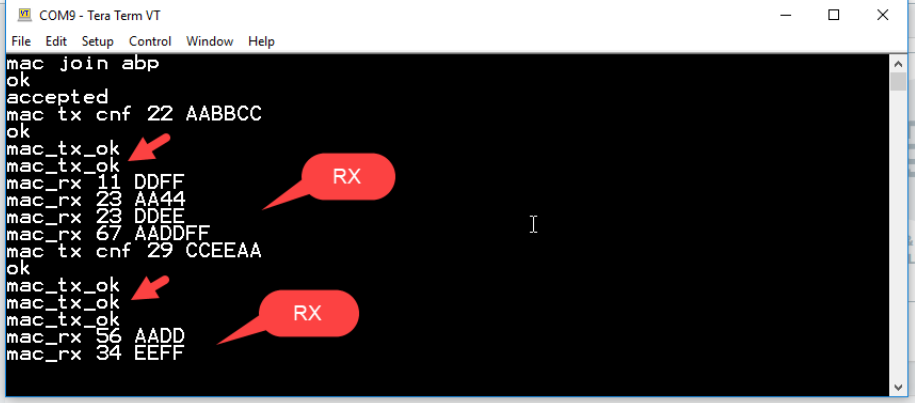
COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join abp
ok
accepted
mac tx cnf 22 AABCC
ok
mac_tx_ok
mac_tx_ok
mac_rx 11 DFFF
mac_rx 23 AA44
mac_rx 23 DDEE
mac_rx 67 AADDF
  
```

Tx data

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
11	ABCD0004	Sent	10/27/2017 12:43:40 AM	67	false	0	AADDF	

1 - 1 / 1 lines
10 lines / page
Page 1 / 1

Note: Class C, open downlink continually until next uplink again.



```

COM9 - Tera Term VT
File Edit Setup Control Window Help
mac join abp
ok
accepted
mac tx cnf 22 AABCC
ok
mac_tx_ok
mac_tx_ok
mac_rx 11 DFFF
mac_rx 23 AA44
mac_rx 23 DDEE
mac_rx 67 AADDF
mac tx cnf 29 CCEEA
ok
mac_tx_ok
mac_tx_ok
mac_tx_ok
mac_rx 36 AADD
mac_rx 34 EEFF
  
```

Tx data

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
13	ABCD0004	Sent	10/27/2017 12:46:34 AM	34	false	0	EEFF	

Uplink (Gateway /Network server)

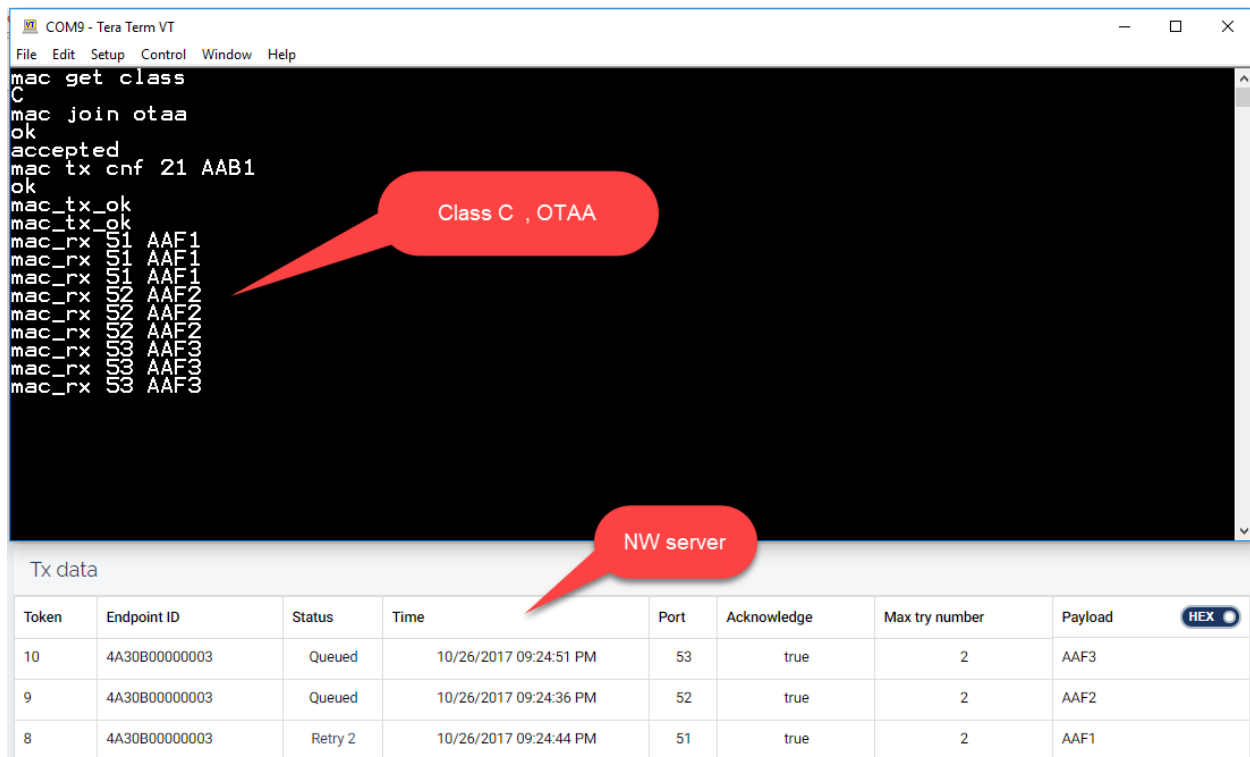
Received data ABP DOWNLOAD

Id	Endpoint ID	Received time	Sequence number	Port	Radio ID	Channel	SNR	RSSI	Frequency	Modulation	Data Rate	Coding rate	Payload	HEX
8	ABCD0004	10/27/2017 12:46:07 AM	1	29	0	4	8 dB	-38 dBm	921.600 MHz	LoRa	SF12BW125	4/5	CCEAA	
7	ABCD0004	10/27/2017 12:46:02 AM	1	29	0	0	9.5 dB	-36 dBm	923.200 MHz	LoRa	SF12BW125	4/5	CCEAA	
6	ABCD0004	10/27/2017 12:42:13 AM	0	22	0	5	8.5 dB	-31 dBm	921.800 MHz	LoRa	SF12BW125	4/5	AABBCC	
5	ABCD0004	10/27/2017 12:42:08 AM	0	22	0	1	8.8 dB	-35 dBm	923.400 MHz	LoRa	SF12BW125	4/5	AABBCC	
4	4A30B000000004	10/27/2017 12:33:34 AM	0	44	0	7	8 dB	-32 dBm	922.200 MHz	LoRa	SF12BW125	4/5	CC	
3	4A30B000000004	10/27/2017 12:33:29 AM	0	44	0	4	10.5 dB	-32 dBm	921.600 MHz	LoRa	SF12BW125	4/5	CC	
2	4A30B000000004	10/27/2017 12:31:21 AM	0	12	0	7	7.2 dB	-35 dBm	922.200 MHz	LoRa	SF12BW125	4/5	AA	
1	4A30B000000004	10/27/2017 12:31:17 AM	0	12	0	4	7.5 dB	-34 dBm	921.600 MHz	LoRa	SF12BW125	4/5	AA	

1 - 8 / 8 lines 10 lines / page 1 OTAA Page 1 / 1

Note: Uplink: mote send message to gateway, Downlink: gateway/Network server send message to mote.

Additional Example: Class C set Tx from network server at 3 data and repeat 2 times.



COM9 - Tera Term VT

```

File Edit Setup Control Window Help
mac get class
C
mac join otaa
ok
accepted
mac tx cnf 21 AAB1
ok
mac_tx_ok
mac_tx_ok
mac_rx 51 AAF1
mac_rx 51 AAF1
mac_rx 51 AAF1
mac_rx 52 AAF2
mac_rx 52 AAF2
mac_rx 52 AAF2
mac_rx 53 AAF3
mac_rx 53 AAF3
mac_rx 53 AAF3

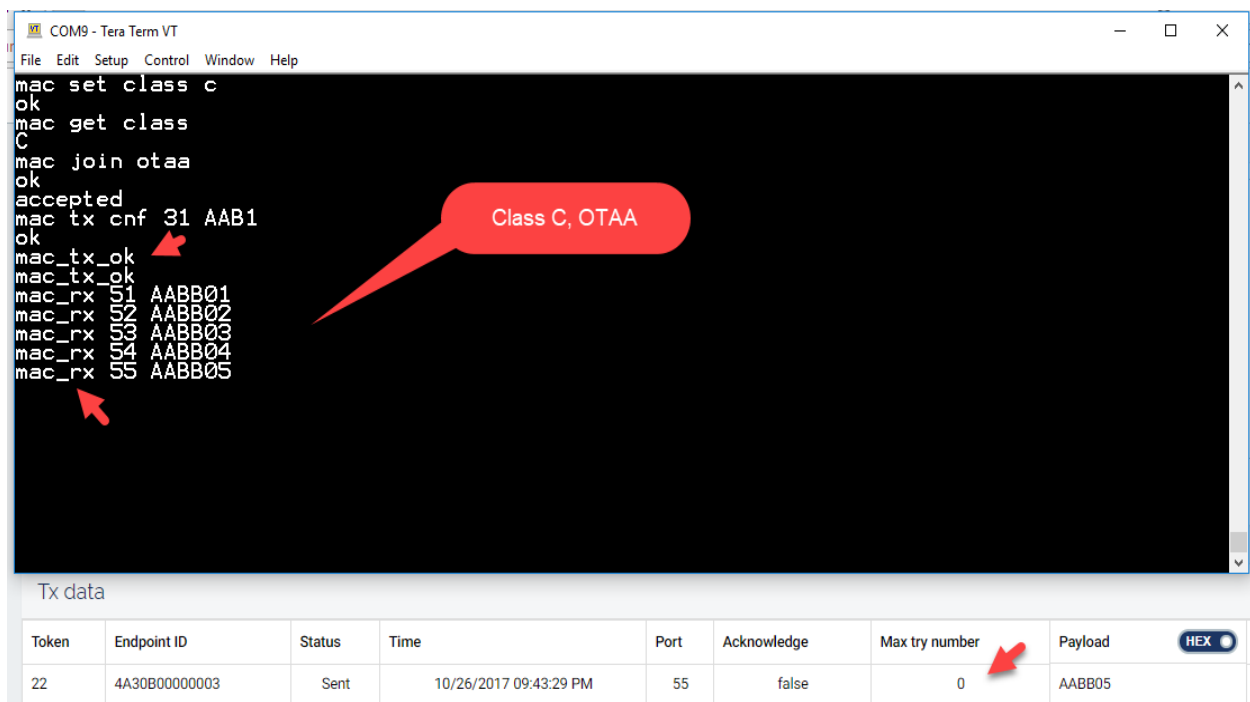
```

Class C , OTAA

NW server

Tx data

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
10	4A30B00000003	Queued	10/26/2017 09:24:51 PM	53	true	2	AAF3	
9	4A30B00000003	Queued	10/26/2017 09:24:36 PM	52	true	2	AAF2	
8	4A30B00000003	Retry 2	10/26/2017 09:24:44 PM	51	true	2	AAF1	



COM9 - Tera Term VT

```

File Edit Setup Control Window Help
mac set class c
ok
mac get class
C
mac join otaa
ok
accepted
mac tx cnf 31 AAB1
ok
mac_tx_ok
mac_tx_ok
mac_rx 51 AABB01
mac_rx 52 AABB02
mac_rx 53 AABB03
mac_rx 54 AABB04
mac_rx 55 AABB05

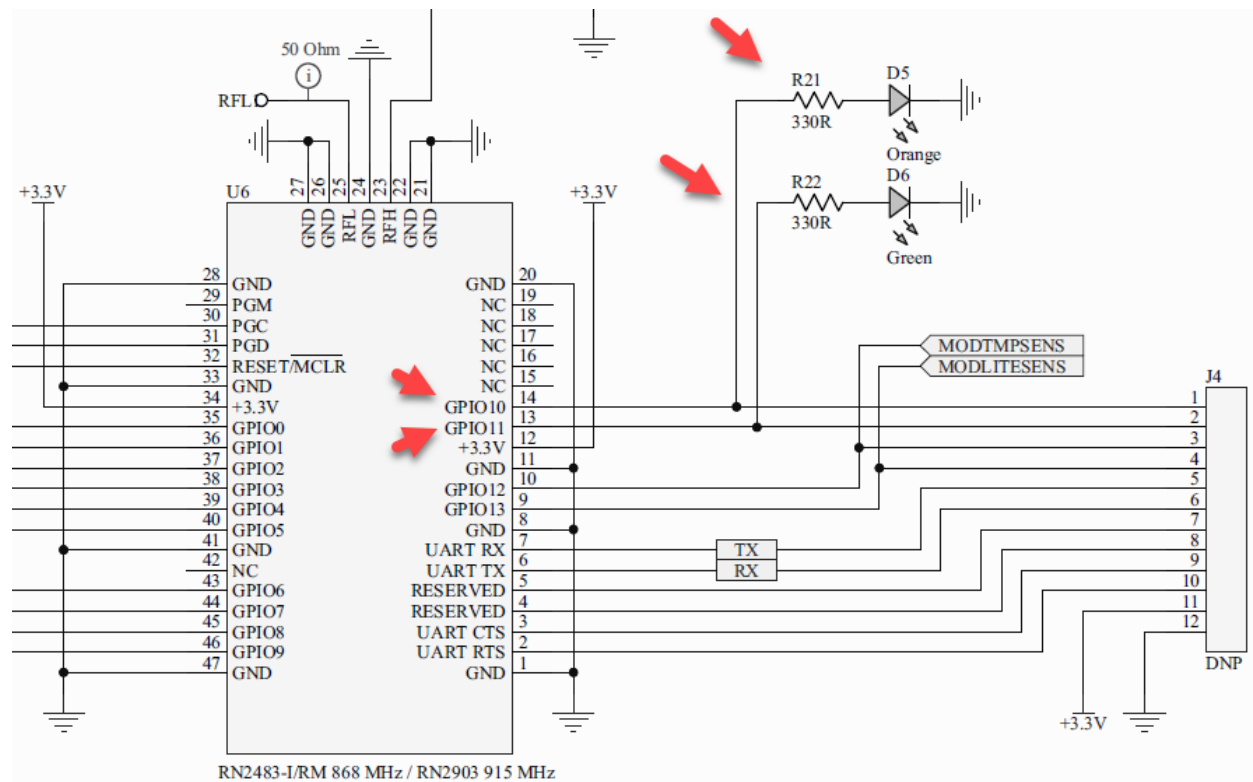
```

Class C, OTAA

Tx data

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload	HEX
22	4A30B00000003	Sent	10/26/2017 09:43:29 PM	55	false	0	AABB05	

Assignment 5: PORT GPIO



- sys set pinmode GPIOxx digout
- sys set pinmode GPIOxx digout
- sys set pindig GPIOxx 1 // GPIO 10 = "1"

Result:

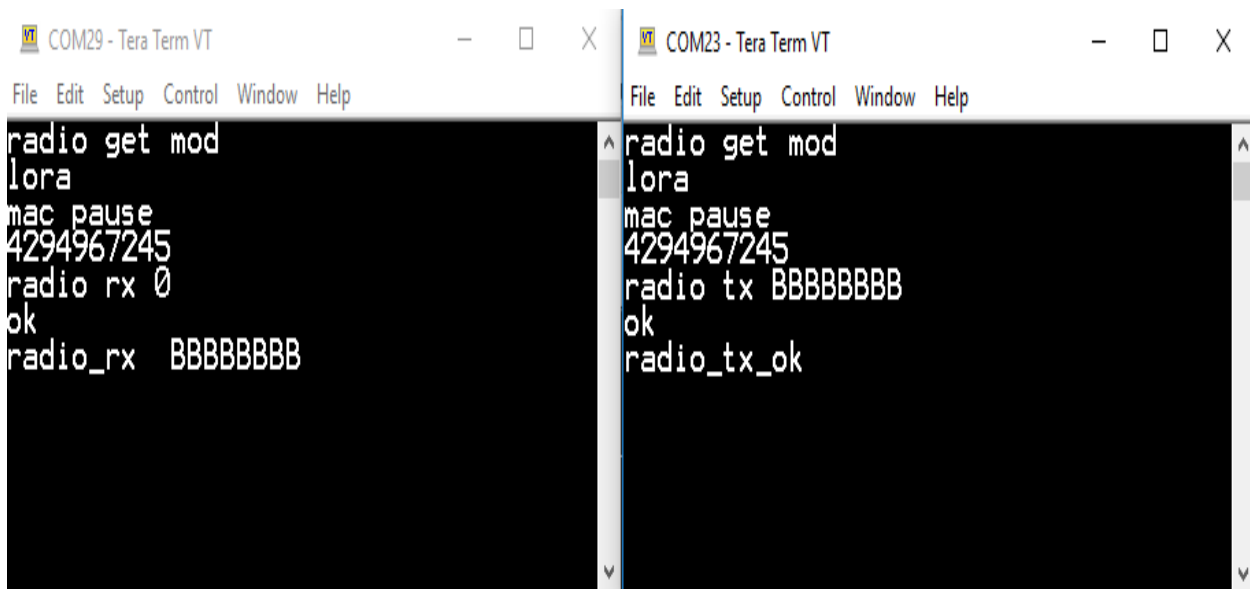
- `sys set pinmode GPIO10 digout`
- `sys set pinmode GPIO11 digout`
- `sys set pindig GPIO10 1 // GPIO 10 = "1"`
- `sys set pindig GPIO11 1`
- `sys set pindig GPIO10 0 // GPIO 10 = "0"`
- `sys set pindig GPIO11 0`

Assignment 6: P2P

Guide

- `radio get mod`
- `mac pause`
- `radio rx 0`
- `radio tx XXXXYYYY`

Note: XXXXYYYY is message !!!



```
COM29 - Tera Term VT
File Edit Setup Control Window Help
radio get mod
lora
mac pause
4294967245
radio rx 0
ok
radio_rx BBBBBBBBB

COM23 - Tera Term VT
File Edit Setup Control Window Help
radio get mod
lora
mac pause
4294967245
radio tx BBBBBBBB
ok
radio_tx_ok
```

COM29 is receiver and COM23 is transmitter

Note: FSK mode can support STAR network. LoRa mode support one by one only.

Summary

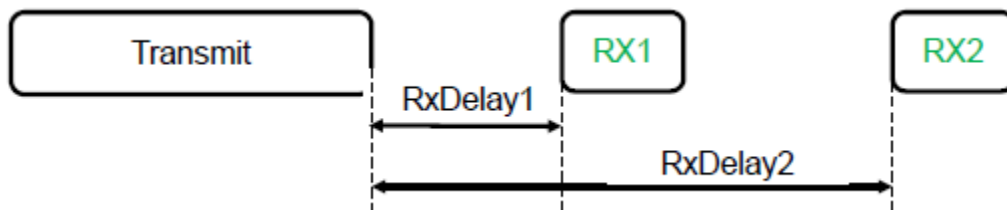
Class A: support

- OTAA
- ABP
- Tx confirm, unconfirm.

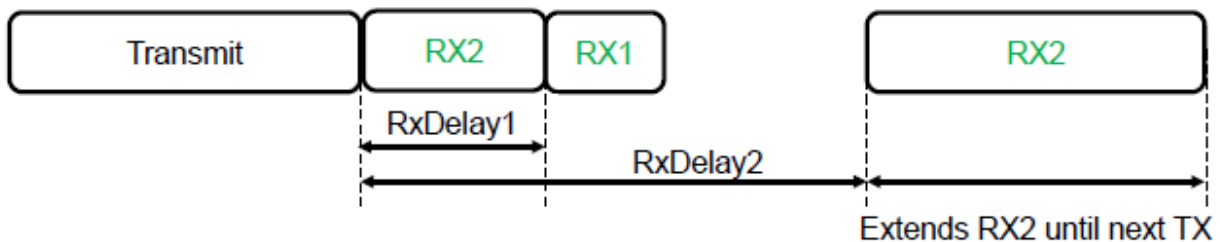
Class C: support

- OTAA
- ABP
- Tx confirm, unconfirm.

Class A



Class C



Additional information:

Note: Observing

OTAA and ABP Joining:

We can set detail of OTAA and ABP joining in our mote module at one time.

So, we can select to join of mote to network server by OTAA or ABP.

Class A and Class C

Register mote for Class A or Class C on Network server can set one Class.

If you need to change Class A or Class C in next time, must register new class of mote on network server

Additional information:

Example 1: 0004A30B00000003 is set class A and OTAA and ABP.

OTAA

ABP

Manage endpoints

List of OTAA endpoints

Device EUI	Application EUI	Application key	Rx window	Rx2 frequency	Rx2 Data Rate	Class	
0004A30B00000001	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000002	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000003	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000004	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000005	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	

OTAA

ABP

Manage endpoints

List of ABP endpoints

Device address	Network session key	Application session key	Rx window	Rx2 frequency	Rx2 Data Rate	Class	
ABCD0001	10002000300040005000600070008000	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
ABCD0002	10002000300040005000600070008000	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
ABCD0003	10002000300040005000600070008000	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
ABCD0004	10002000300040005000600070008000	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
ABCD0005	10002000300040005000600070008000	10002000300040005000600070008000	Auto	923.200 MHz	2	A	






Example 2: 0004A30B00000003 is set Class C on OTAA, ABP still be Class A.

OTAA

ABP

Manage endpoints

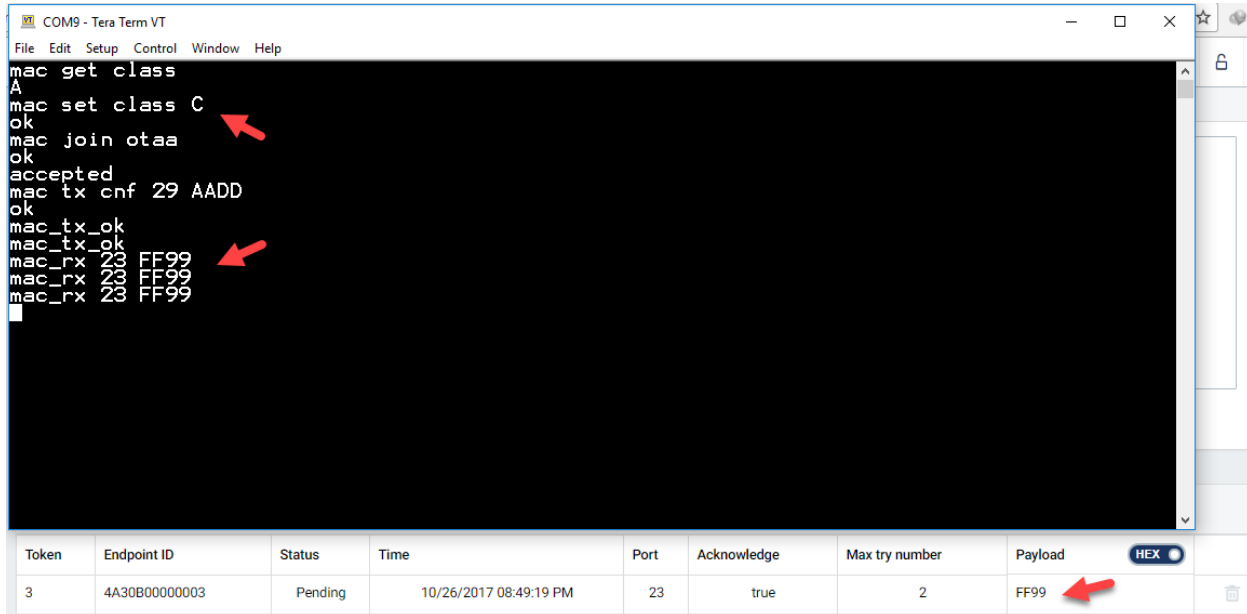
List of OTAA endpoints

Device EUI	Application EUI	Application key	Rx window	Rx2 frequency	Rx2 Data Rate	Class	
0004A30B00000001	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000002	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000003	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	C	
0004A30B00000004	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	
0004A30B00000005	10203040A0B0C0D0	10002000300040005000600070008000	Auto	923.200 MHz	2	A	

Note: My mote module is set class C with OTAA and is set class A with ABP.

Result testing.

Class C and OTAA joining



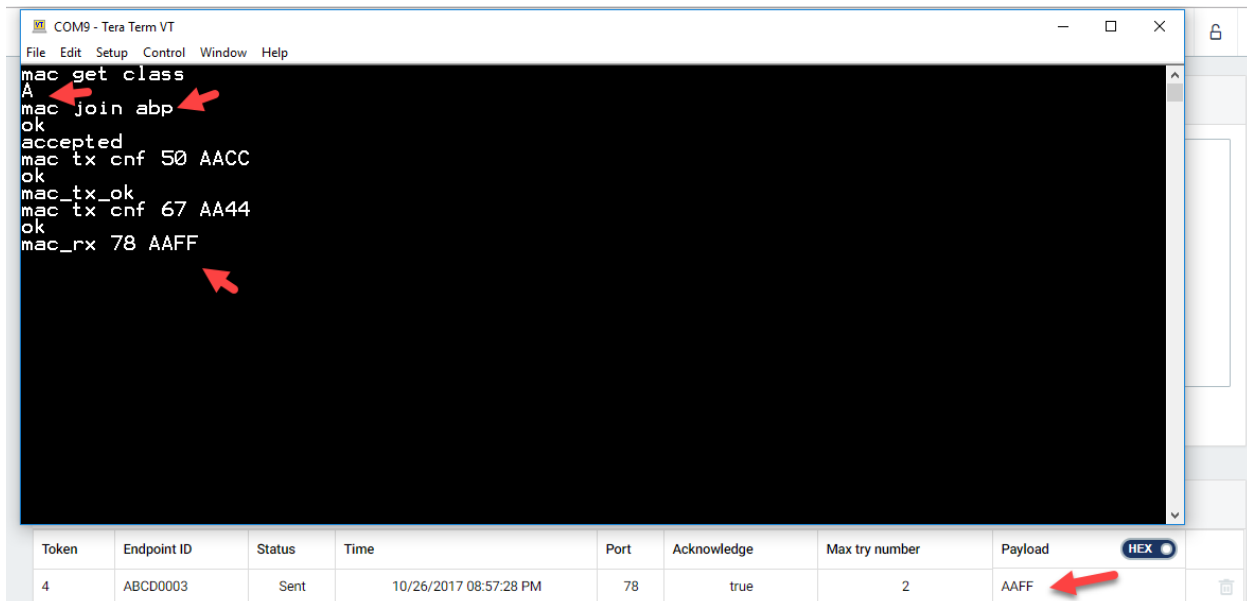
```

mac get class
A
mac set class C
ok
mac join otaa
ok
accepted
mac tx cnf 29 AADD
ok
mac_tx_ok
mac_tx_ok
mac_rx 23 FF99
mac_rx 23 FF99
mac_rx 23 FF99

```

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload
3	4A30B00000003	Pending	10/26/2017 08:49:19 PM	23	true	2	FF99

Class A and ABP joining



```

mac get class
A
mac join abp
ok
accepted
mac tx cnf 50 AACC
ok
mac_tx_ok
mac tx cnf 67 AA44
ok
mac_rx 78 AAFF

```

Token	Endpoint ID	Status	Time	Port	Acknowledge	Max try number	Payload
4	ABCD0003	Sent	10/26/2017 08:57:28 PM	78	true	2	AAFF

Note: We can use class C on OTAA joining and use class A on ABP

RN2903A LAB Manual REV 1.0 LoRAWAN



For Update fw of RN2903 module (AS923) LoRa module



Material programmer

PICKit3 or ICD3 or RealICE

Software you need

MPLAB X IDE and specifically MPLAB IPE to program the Firmware

- Connect micro-USB cable from J1 connector to your PC
- Target device: PIC18LF46K22
- PICKit3 to J5 connector

RN2903A LAB Manual REV 1.0 LoRAWAN

For Develop fw MOTE



Material programmer

PICKit3 or ICD3 or RealICE

Software you need

MPLAB X IDE and specifically MPLAB IPE to program the Firmware

- Connect micro-USB cable from J5 connector to your PC
- Target device: PIC18LF45K50
- PICKit3 to J5 connector

Reference website link:

<https://www.microchip.com/wwwproducts/en/RN2903>

<https://www.microchip.com/wwwproducts/en/RN2483>

[RN2903 LoRa Technology Module Command Reference User's Guide](#)

<http://ww1.microchip.com/downloads/en/DeviceDoc/40001811A.pdf>

[RN2483 LoRa Technology Module Command Reference User's Guide](#)

<http://ww1.microchip.com/downloads/en/DeviceDoc/40001784F.pdf>

[RN2903 LoRa Technology PICtail/PICtail Plus Daughter Board User's Guide](#)

<http://ww1.microchip.com/downloads/en/DeviceDoc/50002424A.pdf>

THANK YOU

Note: