LoRaWAN™ 101
Hands-On

Getting Up and Running with LoRaWAN™ Long-Range Networking

Agenda

- Internet of Things (IoT)
- LoRaWAN™ Networking Standard
- LoRa® Technology Wireless Modules
- Getting Started with RN2903 Module
- **Hands-on workshop**
  - Connecting a Mote to the Kerlink's SPN Gateway Network

Homework Preparation

- **You will need:**
  - A Laptop with USB port
  - Microchip USB driver (can auto-install, but slow)
    - www.Microchip.com/MCP2200 (under documentation tab)
  - Any generic “Terminal” app (*but not PuTTY*)
    - Termite (http://termite.soft112.com/)
    - TeraTerm (http://teraterm.software.informer.com/)
    - Coolterm (www.macupdate.com/app/mac/31352/coolterm)
    - Etc
  - **Settings: 57600bps, 8n1, no flow control, echo on, set options to include CR+LF**

Lab Summary

- **In the following labs you will:**
  - Setting up "Serial Terminal" app
    - Simple Commands

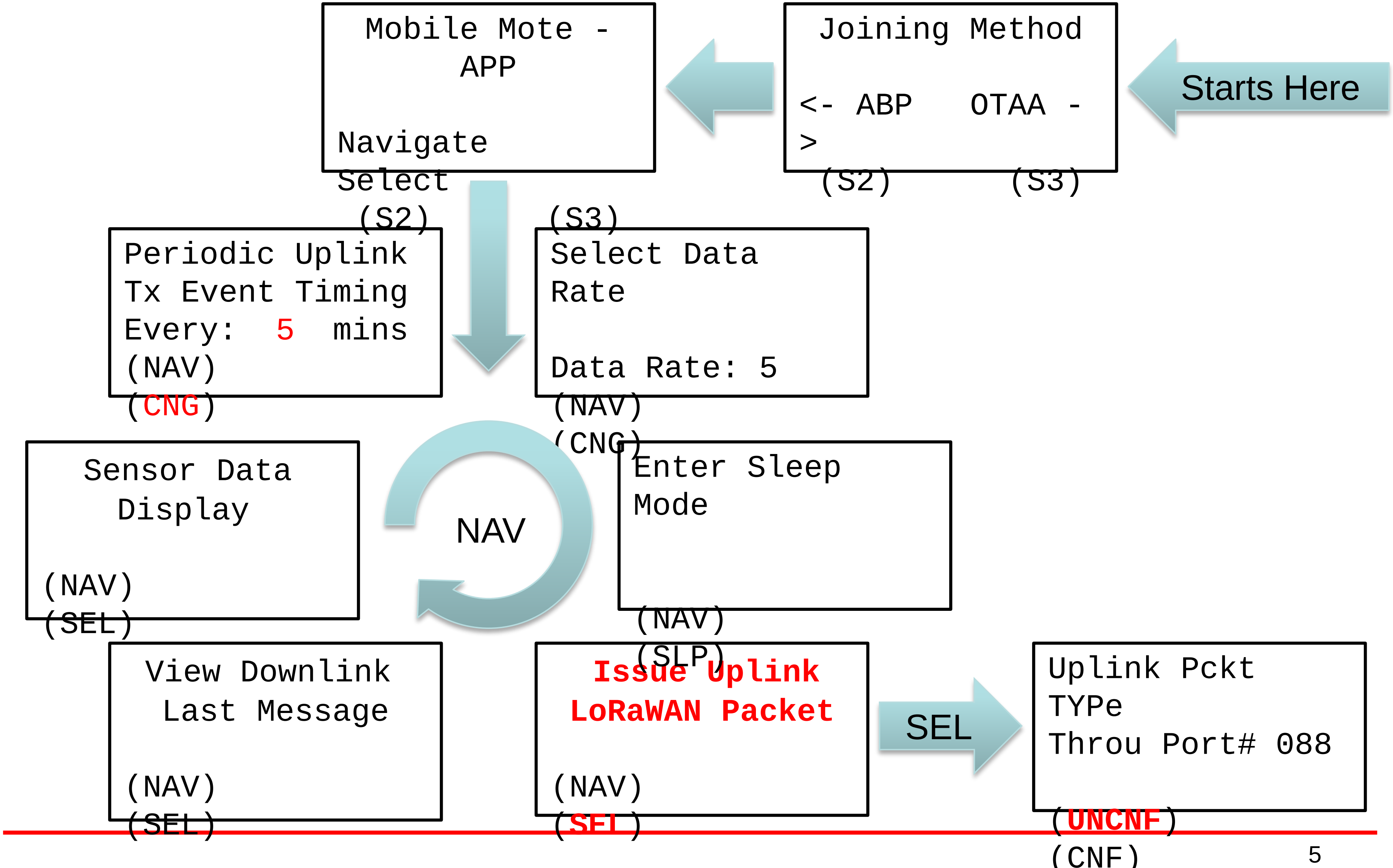
    Lab 1
  - Setup Account for Application in MyDevice

    Lab 2
  - RN2903 Module configuration and Over-the-Air Activation (OTAA)

    Lab 3
  - Sending data to Network Application Server

    Lab 4

Mote Menus



Joining Method

<- ABP OTAA ->

(S2) (S3)

Starts Here

Mobile Mote - APP

Navigate Select

(S2) (S3)

Periodic Uplink Tx Event Timing

Every: 5 mins

(NAV)

(CNG)

Select Data Rate

Data Rate: 5

(NAV)

(CNG)

Sensor Data Display

(NAV)

(SEL)

NAV

Enter Sleep Mode

(NAV)

(SLP)

View Downlink Last Message

(NAV)

(SEL)

Issue Uplink LoRaWAN Packet

(NAV)

(SEL)

SEL

Uplink Pckt TYPe

Throu Port# 088

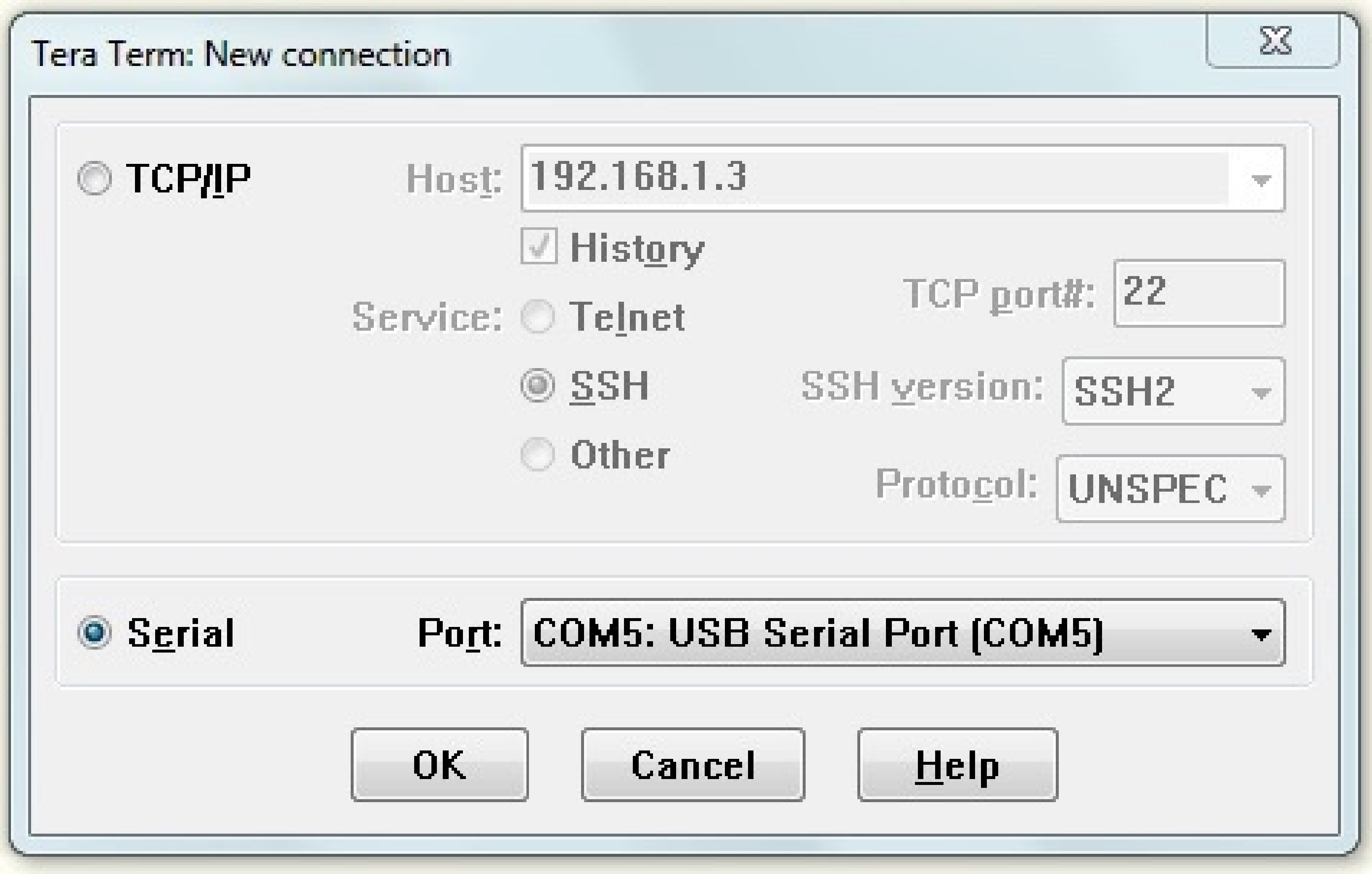
(UNCNF)

(CNF)

Lab 1:

Setting up “Serial Terminal” app

Lab 1 : Comm Port Selection



Tera Term: New connection

○ TCP/IP Host: 192.168.1.3

☑ History

Service: ○ Telnet ● SSH ○ Other

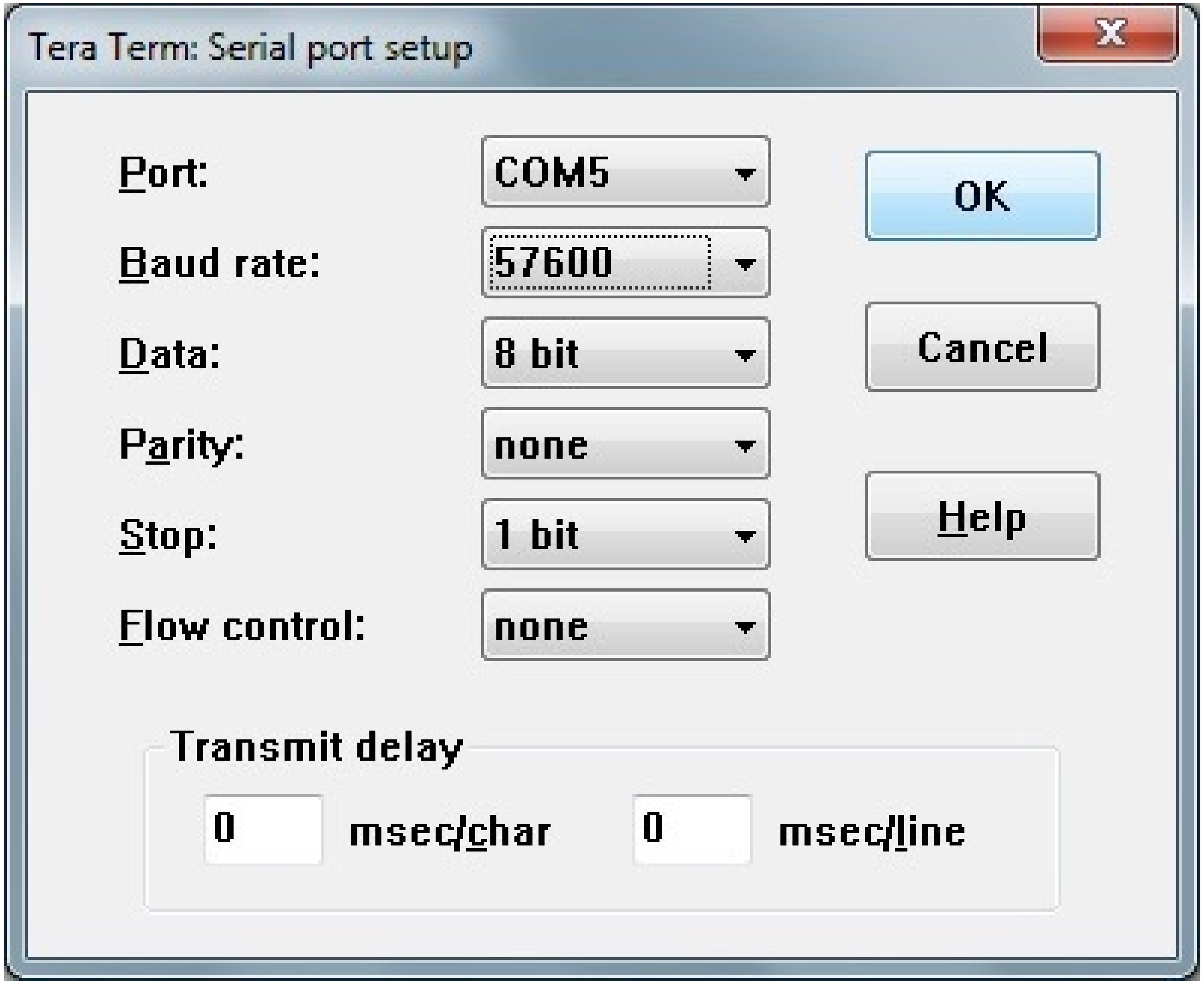
TCP port#: 22

SSH version: SSH2

Protocol: UNSPEC

● Serial Port: COM5: USB Serial Port (COM5)

Lab 1 : Serial Settings



Tera Term: Serial port setup

Port: COM5

Baud rate: 57600

Data: 8 bit

Parity: none

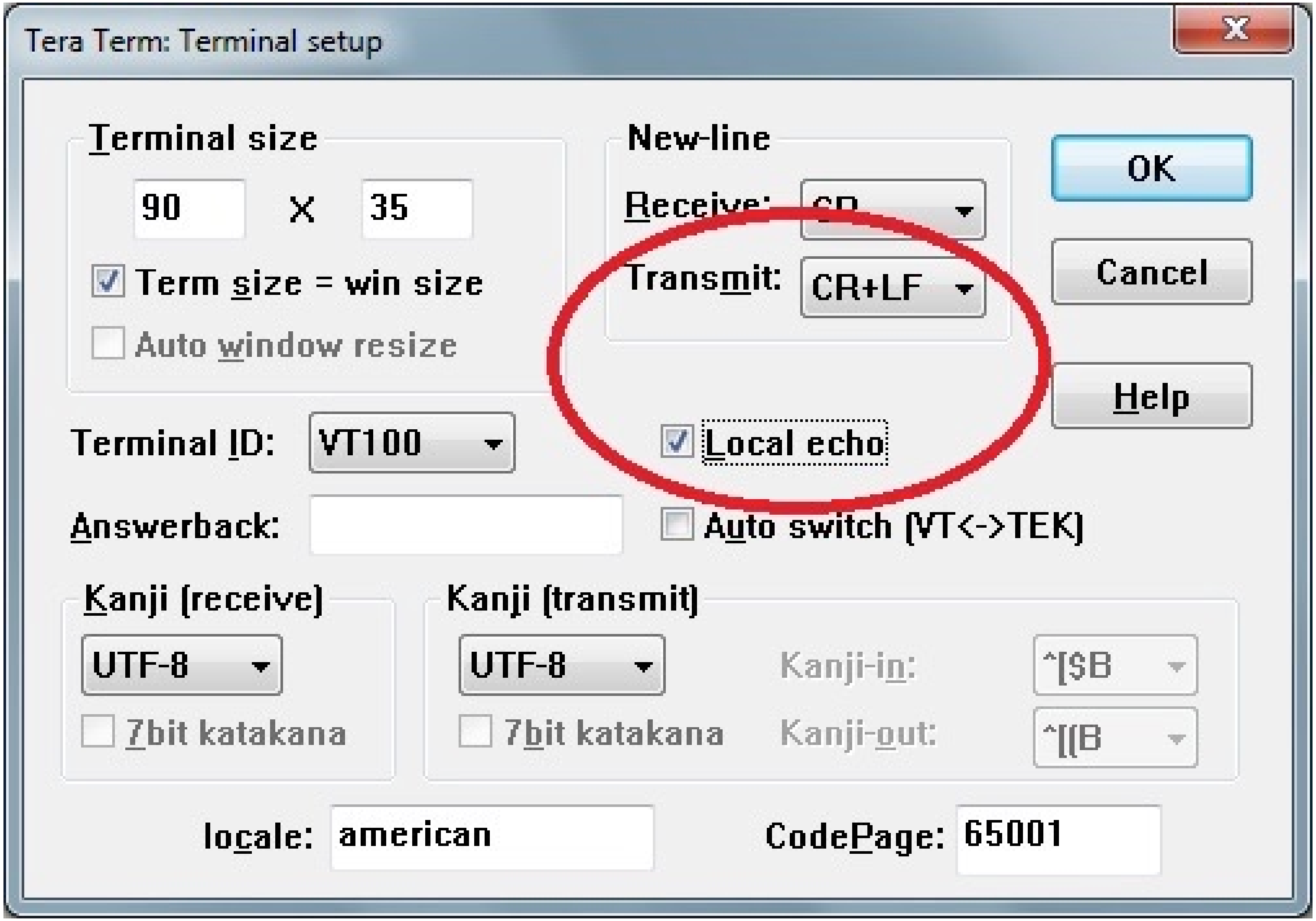
Stop: 1 bit

Flow control: none

Transmit delay

0 msec/char 0 msec/line

Lab 1 : Terminal Settings



Tera Term: Terminal setup

Terminal size

90 X 35

☑ Term size = win size

□ Auto window resize

New-line

Receive: CR

Transmit: CR+LF

Terminal ID: VT100

Answerback:

☑ Local echo

□ Auto switch (VT<->TEK)

Kanji (receive)

UTF-8

□ 7bit katakana

Kanji (transmit)

UTF-8

□ 7bit katakana

Kanji-in: ^[$B

Kanji-out: ^[(B

locale: american

CodePage: 65001

Useful Notes – RN Parser

- **The commands to type are shown as:**
  - This is a command: `sys reset`
  - Commands are case sensitive, parameters are not
  - The parser is sensitive to extra blank spaces
    - E.g “`sys reset`” works
    - “`sys  reset`” or “`sys reset `” does not work
  - Copy/paste can introduce additional chars like LF or CR
  - If you see “invalid_params” hit multiple “Enter” to clear

Lab 1: Connecting to the Node

This lab verifies that the USB/UART communication to the LoRa® node is working correctly

0.1 – Connect board to PC via USB cable.

0.2 – Open Device Manager and locate assigned COM port

0.3 – Using a PC terminal program connect on said port **(settings: 57600bps, 8n1, no flow control, echo on, set options to include CR+LF)**

0.4 – Check for communication by using: `sys get ver`

0.5 – The module will return module name, version & compilation timestamp:

```
RN2903 0.9.5 Sep 02 2015 17:19:55
```

HINT – commands are case sensitive, parameters are not

Lab 1: Basic LoRa® Connectivity (Step 1 – Reset & Find your EUI)

As no two end-devices are allowed the same device address, a unique DevAddr should be used. We can use the **H**ard**W**are **EUI** for this, read from hardware via a command.

Note the HWEUI is 8-bytes whereas devaddr is only 4-bytes. Use the 4-LSBytes (right half of the EUI)

1.1 – Reset module using: `sys RESET`

1.2 – Read the hardware Unique Identifier: `sys get hweui`

HINT – Make a note of your DevAddr – you will need it next

Lab 1 : Resetting Module

- **Reset and restart the RN2903A module.**
- **Type and send the command: sys reset**
- **The module should respond with module name, version and compilation timestamp.**

Example output:

```
< sys reset
> RN2903 0.6.0 Jun 16 2015 14:30:04
```

Lab 1 : Reading hwEUI & devEUI

- **Read the EUI-64 address of the module:**
- **Type and send the command: sys get hweui**
- **The module should respond with an 8-bit hexadecimal number, representing the EUI-64 address of the module, like 0004A30B001A13BB.**
- **The hweui is normally used as deveui.**
  (For this lab learning, we have assigned separate deveui for easy of practice. To set deveui same as hweui 'sys set deveui 0004A30B001B78E4').

```
< sys get hweui
> 0004A30B001B78E4
< sys get deveui
> 0004A30B001B78E4
```

Lab 2:

Setting up account in Application Server

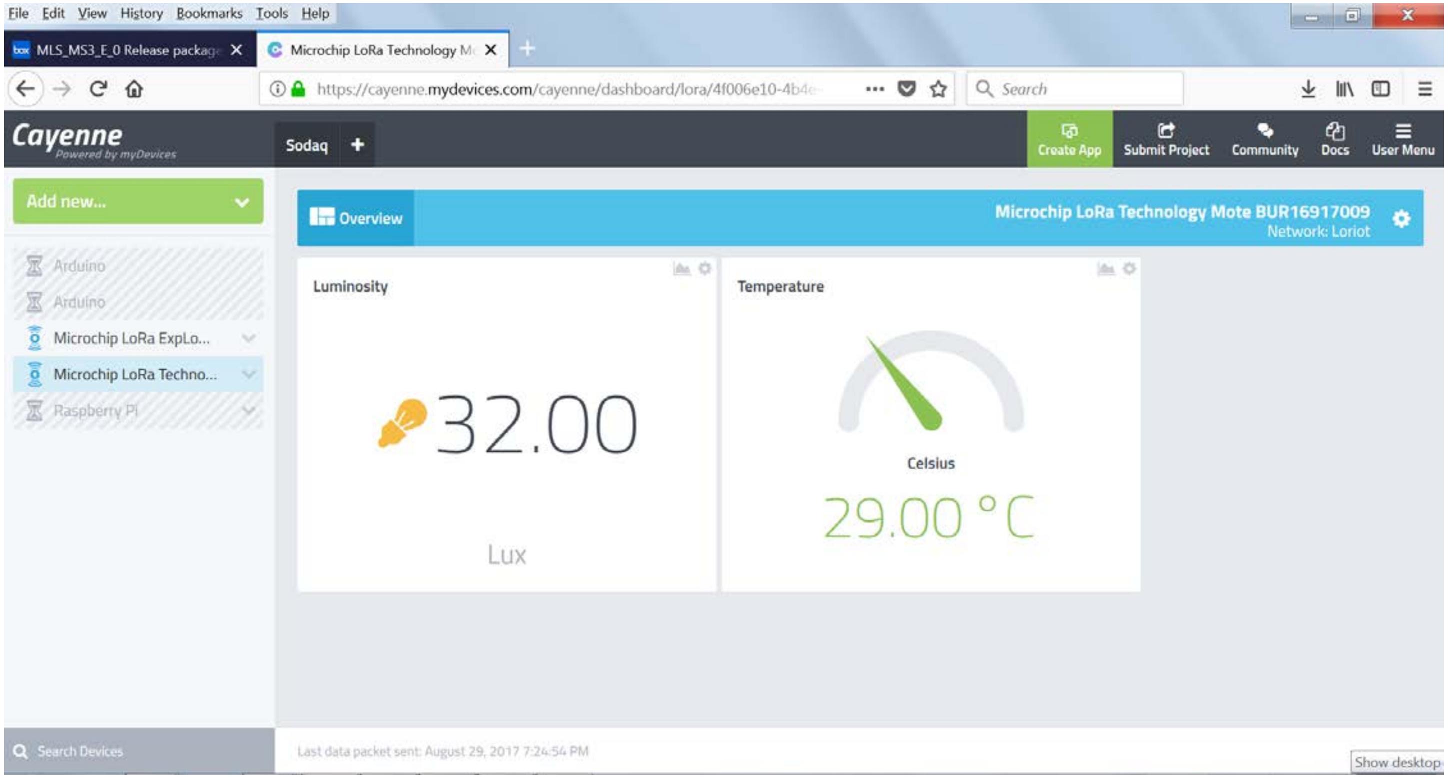
(MyDevice)

MyDevice Dashboard

- **Create a login ID account.**

https://cayenne.mydevices.com/cayenne/login

- **You can add 10 devices onto your account free.**



Microchip LoRa Technology Mote BUR16917009
Network: Loriot

Luminosity
32.00
Lux

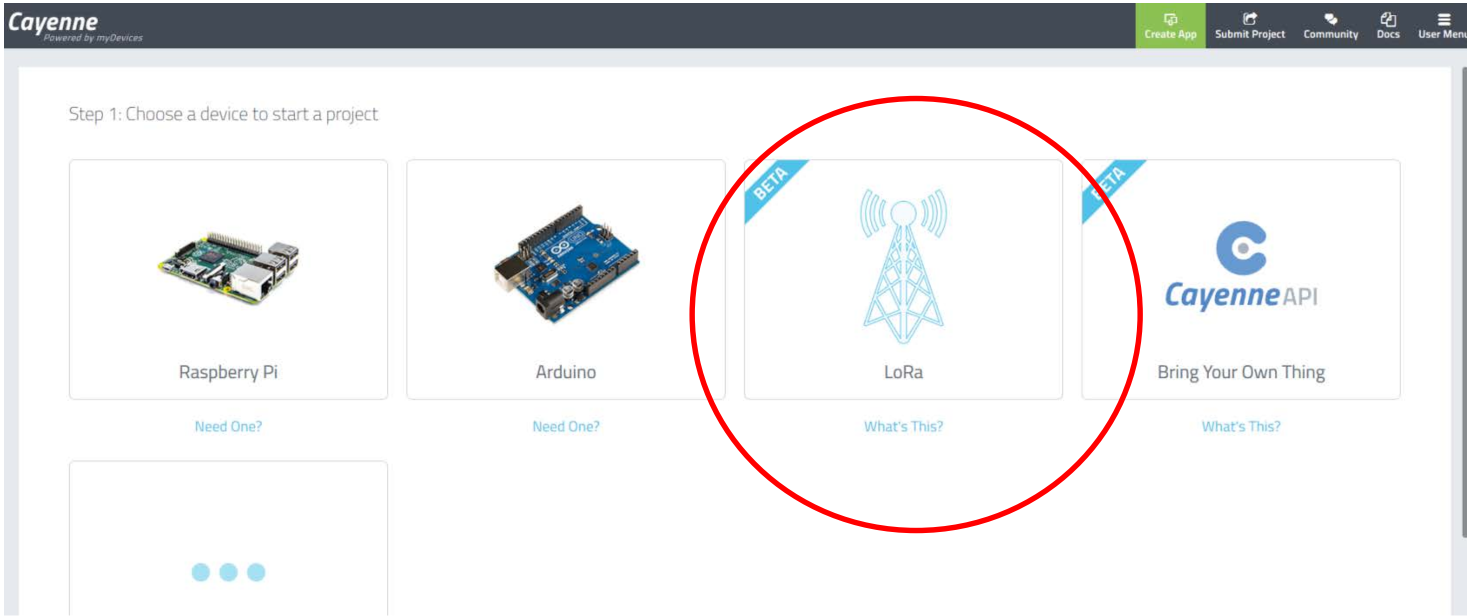
Temperature
Celsius
29.00 °C

Last data packet sent: August 29, 2017 7:24:54 PM

Cayenne MyDevice Setup

Registration of Device

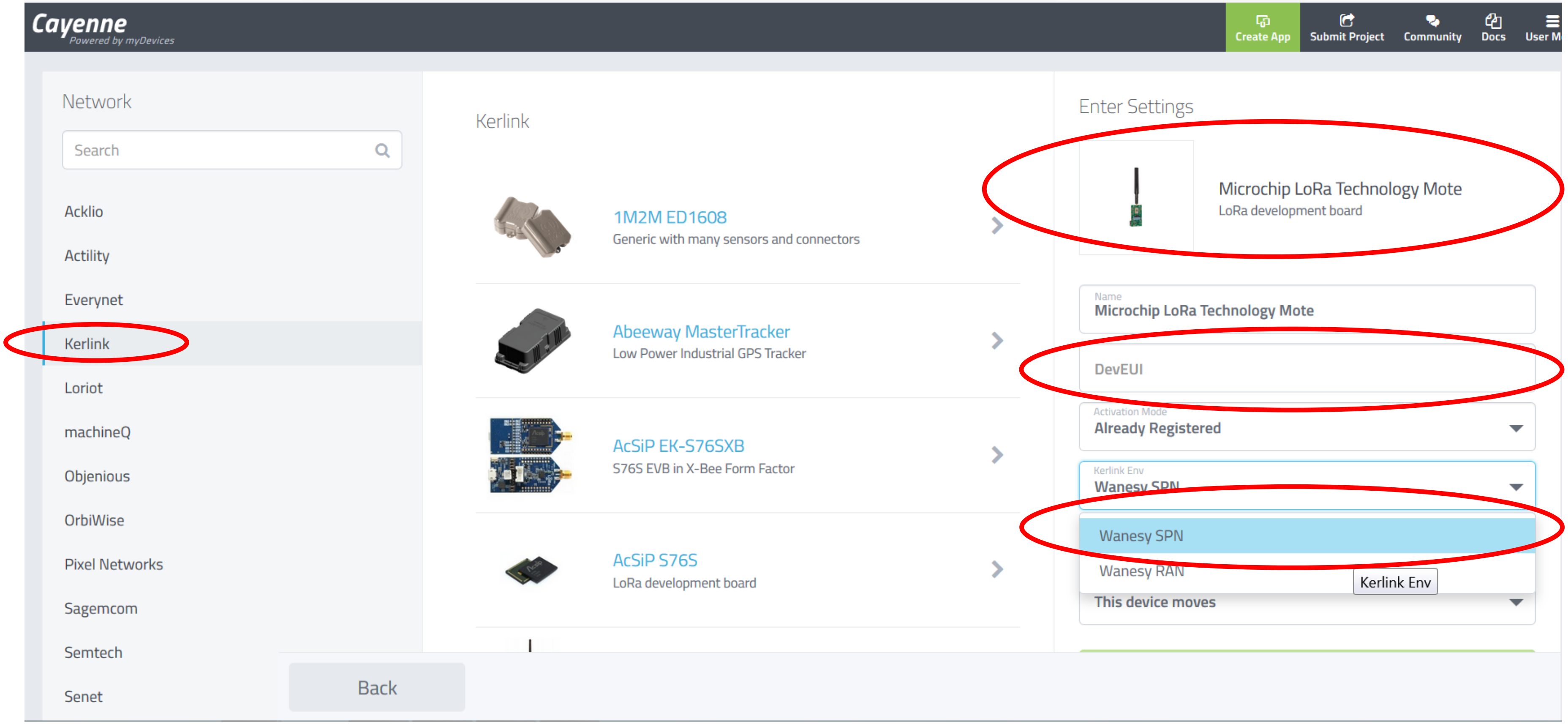
After registration of account



Step 1: Choose a device to start a project

Cayenne MyDevice Setup

Registration of Device



Cayenne
Powered by myDevices

Create App | Submit Project | Community | Docs | User M

Network

Search

- Acklio
- Actility
- Everynet
- Kerlink
- Loriot
- machineQ
- Objenious
- OrbiWise
- Pixel Networks
- Sagemcom
- Semtech
- Senet

Kerlink

1M2M ED1608
Generic with many sensors and connectors

Abeeway MasterTracker
Low Power Industrial GPS Tracker

AcSiP EK-S76SXB
S76S EVB in X-Bee Form Factor

AcSiP S76S
LoRa development board

Back

Enter Settings

Microchip LoRa Technology Mote
LoRa development board

Name: Microchip LoRa Technology Mote

DevEUI

Activation Mode: Already Registered

Kerlink Env: Wanesy SPN

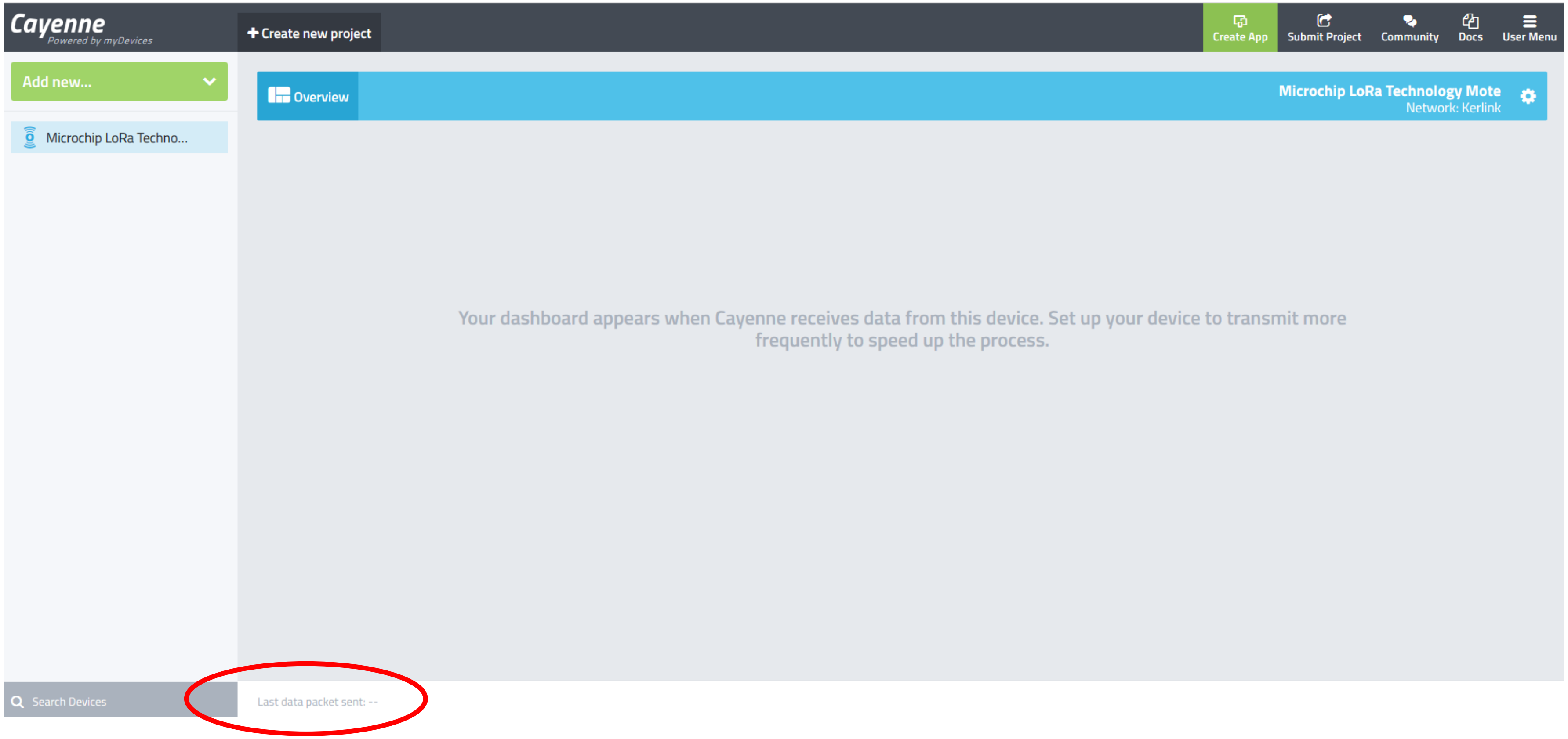
- Wanesy SPN
- Wanesy RAN

Kerlink Env

This device moves

Cayenne MyDevice Setup

Registration of Device

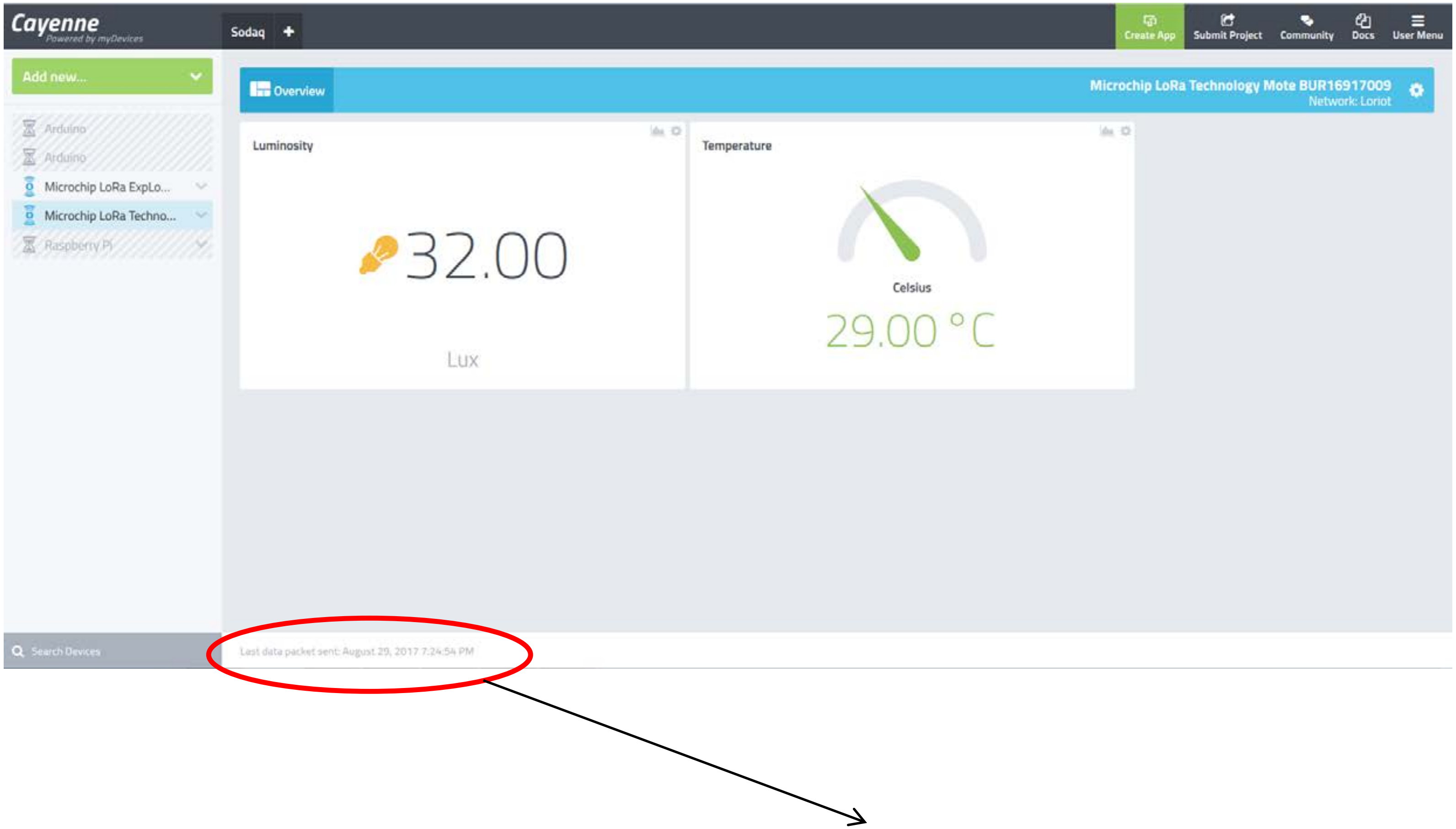


Your dashboard appears when Cayenne receives data from this device. Set up your device to transmit more frequently to speed up the process.

Last data packet sent: --

Cayenne MyDevice Setup

Registration of Device



Microchip LoRa Technology Mote BUR16917009
Network: Loriot

Luminosity: 32.00 Lux

Temperature: Celsius 29.00 °C

Last data packet sent: August 29, 2017 7:24:54 PM

Time stamping of last Tx Data

Lab 3:

RN2903A **Module configuration and Over-the-Air Activation (OTAA)**

Lab 3 Objectives

- Configure the RN2903A Module
- Activate the RN2903A Module using Over-the-Air Activation (OTAA) with the ASCII command set

Lab 3 : End-Device with Over-the-Air Activation (OTAA)

Configure the cryptographic keys for Over-the-Air activation (OTAA). The following keys are needed:

Device EUI, application EUI, application key.

It is recommended to use the EUI-64 address retrieved from hweui as device EUI. This number is UNIQUE for each module and so has to be the device EUI.

For this lab learning, we have set up the deveui for you in your mote board.

Configuration

```
< mac set deveui 0004A30B001B78E4  (DO NOT NEED FOR THIS LAB)
> ok
< mac set appeui 0004A30B00000000
> ok
< mac set appkey 2B7E151628AED2A6ABF7158809CF4F3C
> ok
< mac save
> ok
```

Activation

```
> mac join otaa
< ok
< accepted
```

Lab 4:

Sending data to Network Application Server via host MCU

Lab 3 Objectives

- Configure the RN2903A Module
- Sending data via serial interface to network server

Lab 4 : Sending data to Network Application Server via host MCU

(mac tx <type> <portno> <data>)

<type>: string representing the uplink payload type, either cnf or uncnf

(cnf – confirmed, uncnf – unconfirmed)

<portno>: decimal number representing the port number, from 1 to 223

<data>: hexadecimal value.

Activation by line command

```
> mac join otaa
< ok
< accepted
```

Send Data via unconfirmed message

```
> mac tx uncnf 10 AABBCC
< ok
< mac_tx_ok
```

Send Data via confirmed message

```
> mac tx cnf 5 CCBBAA
< ok
< mac_tx_ok
```

‘mac_err’ if transmission was unsuccessful, ACK not received back from the server

Lab 5:

Sending data to Network Application Server

(MyDevice)

Lab 5 Objectives

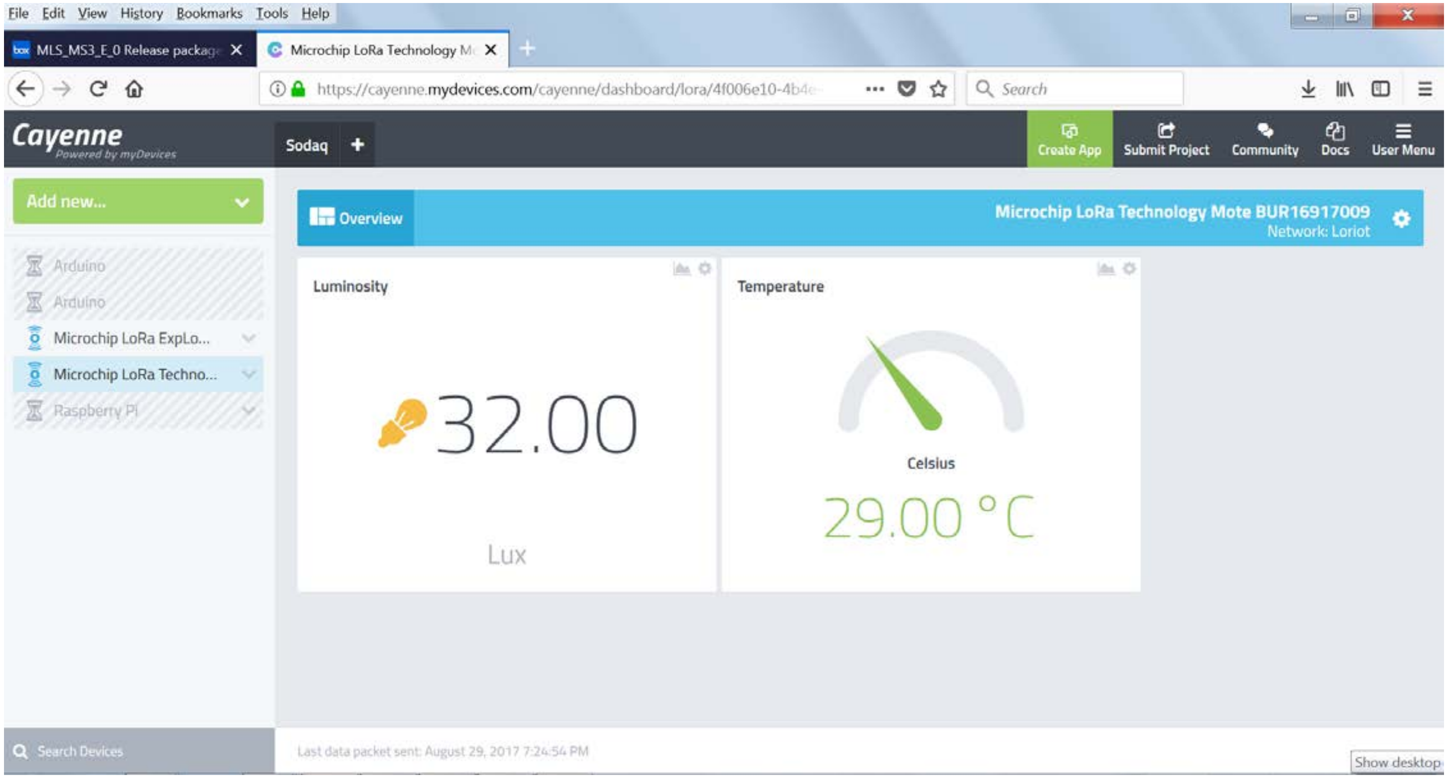
- **Sending data from Mote Board to Network Server and Application**

MyDevice Dashboard

- **Create a login ID account.**

https://cayenne.mydevices.com/cayenne/login

- **You can add 10 devices onto your account free.**



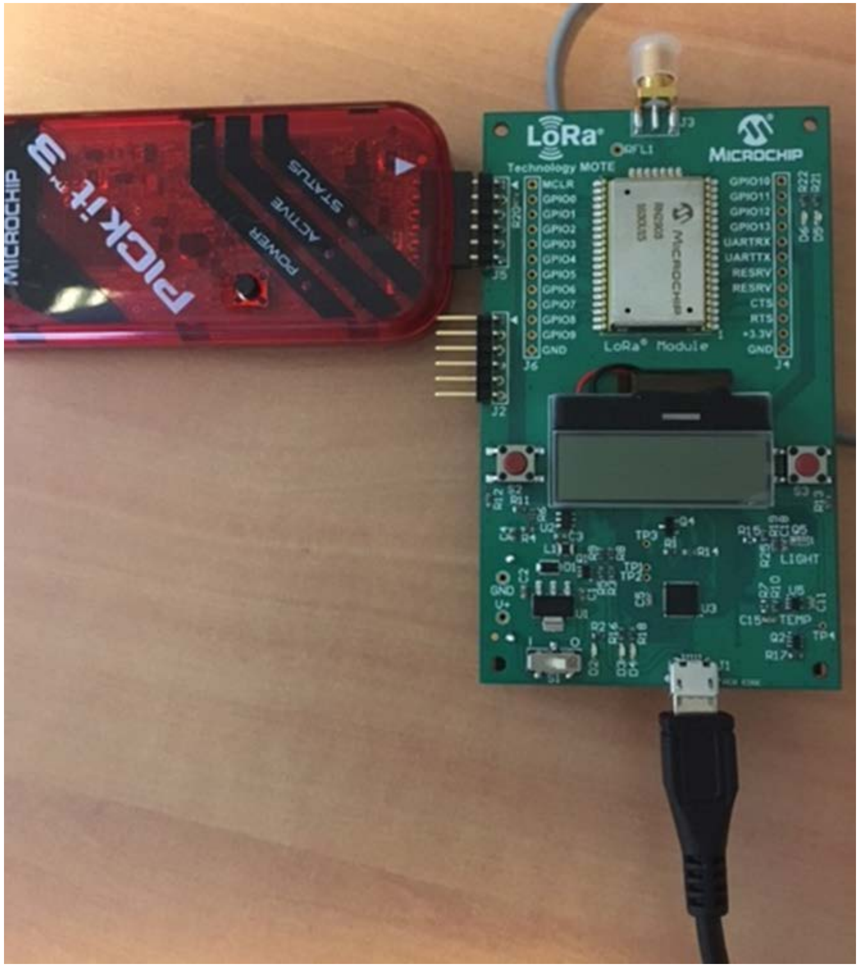
For Further Information:

- **RN2903 Command Reference Users Guide (spec 40001811)**

- **www.Microchip.com/RN2903**

- **50 Pages of Fun!**

Flashing Module Firmware on Mote



Material programmer

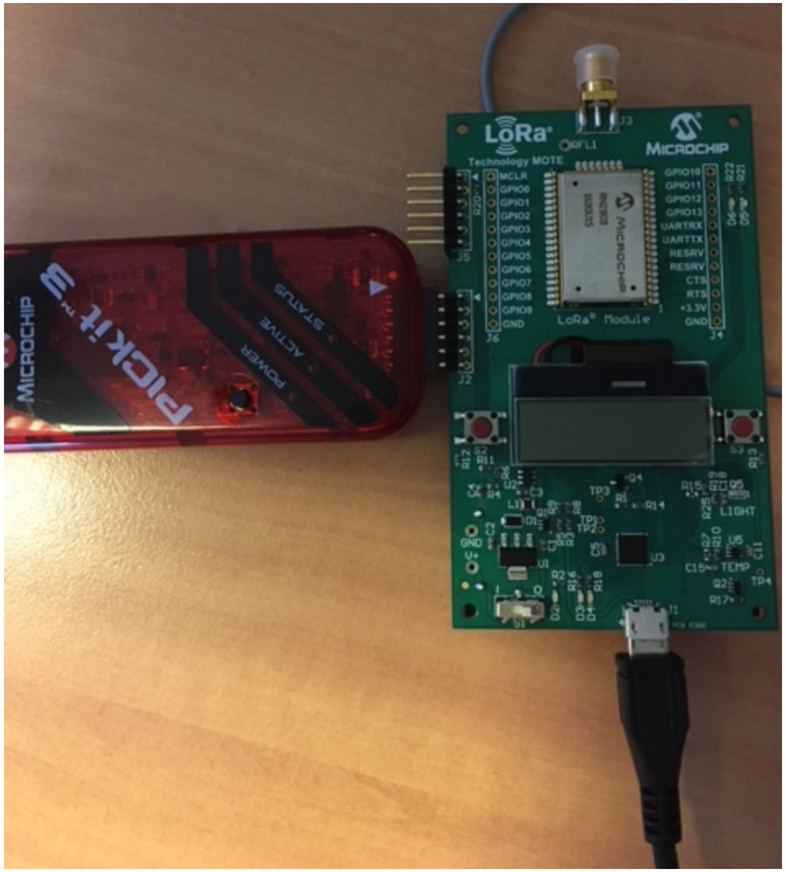
PICKit3 or ICD3 or RealICE

Software you need

MPLAB X IDE and specifically MPLAB IPE to program the Firmware

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

Upgrading Mote Board Firmware



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