

LoRa Studio User Guide

Welcome to LoRa Studio v1.0.0!

LoRa Studio is an application that demonstrates the capabilities and features of Semtech radio chips. It allows users to configure, test, and visualize the performances of the gen4 series in various scenarios. All this without writing even one line of code!

Please make sure you are up to date with the latest version by visiting the [LoRa PLUS webpage](#)!

Getting Started

Supported development kits

LoRa Studio supports the following development kits:

- [LR2021 ARDUINO LEGACY \(STM32\)](#)



Based on the STM32L4 platform, this dev kit is composed of a Nucleo STM32L476RG, an adaptation board to handle the Arduino pinning and the LR2021 radio expansion board. This kit comes as 3 different part numbers depending on the region:

- [LR2021EVK1XBS1_ARDUINO_LEGACY](#) : LoRa Plus™ Development Kit, LR2021, 868MHz for Europe
- [LR2021EVK1XCS1_ARDUINO_LEGACY](#) : LoRa Plus™ Development Kit, LR2021, 915MHz for North America
- [LR2021EVK1XGS1_ARDUINO_LEGACY](#) : LoRa Plus™ Development Kit, LR2021, 490MHz for China and Asia

- [LR2021 ARDUINO NRF54 Dev Kit](#)



Based on the nRF54L15 platform, this dev kit is composed of a nRF54-DK, an adaptation board to handle the Arduino pinning and the LR2021 radio expansion board: This kit comes as 3 different part numbers depending on the region:

- [LR2021EVK1XBS1_ARDUINO](#) : LoRa Plus™ Development Kit, LR2021, 868MHz for Europe
- [LR2021EVK1XCS1_ARDUINO](#) : LoRa Plus™ Development Kit, LR2021, 915MHz for North America
- [LR2021EVK1XGS1_ARDUINO](#) : LoRa Plus™ Development Kit, LR2021, 490MHz for China and Asia

- [LR2021 XIAO Dev Kit](#)



Based on the Seeeduino XIAO platform, this dev kit is composed of the latest integration of the NRF54 MCU in the compact XIAO form factor and the LR2021 radio, either as a module or as an expansion board. This kit comes as 3 different part numbers depending on the region:

- [LR2020EVK1XBS1_XIAO](#) : LoRa Plus™ Development Kit, LR2021, 868MHz for Europe
- [LR2021EVK1XCS1_XIAO](#) : LoRa Plus™ Development Kit, LR2021, 915MHz for North America
- [LR2021EVK1XGS1_XIAO](#) : LoRa Plus™ Development Kit, LR2021, 490MHz for China and Asia

Drivers Installation

- [STM32L4 based kits](#)

On Windows, to make sure that you can properly connect to this kit, you need to install the ST-LINK USB driver available here: [ST-LINK USB driver](#).

On Linux, you can install the *Stlink tool* suite following the instruction on [the GitHub page](#) but make sure that your OS version is [supported](#).

Otherwise, you need to create a udev rule to allow non-root access to the device:

Create a file named [/etc/udev/rules.d/49-stlinkv2-1.rules](#) with the following content:

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374a", \
    MODE=="0666", \
    SYMLINK+="stlinkv2-1"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374b", \
    MODE=="0666", \
    SYMLINK+="stlinkv2-1"
SUBSYSTEMS=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="3752", \
    MODE=="0666", \
    SYMLINK+="stlinkv2-1"
```

[Copy](#)

Then reload the udev rules with the command:

```
sudo udevadm control --reload-rules && sudo udevadm trigger
```

[Copy](#)

After this, unplug and replug the device. You should now have access to it without needing root privileges.

In both cases, once properly installed, you should see a mass storage device usually named NUCLEO_L476RG when plugging the kit to your computer. You should also see a serial port associated with the device.

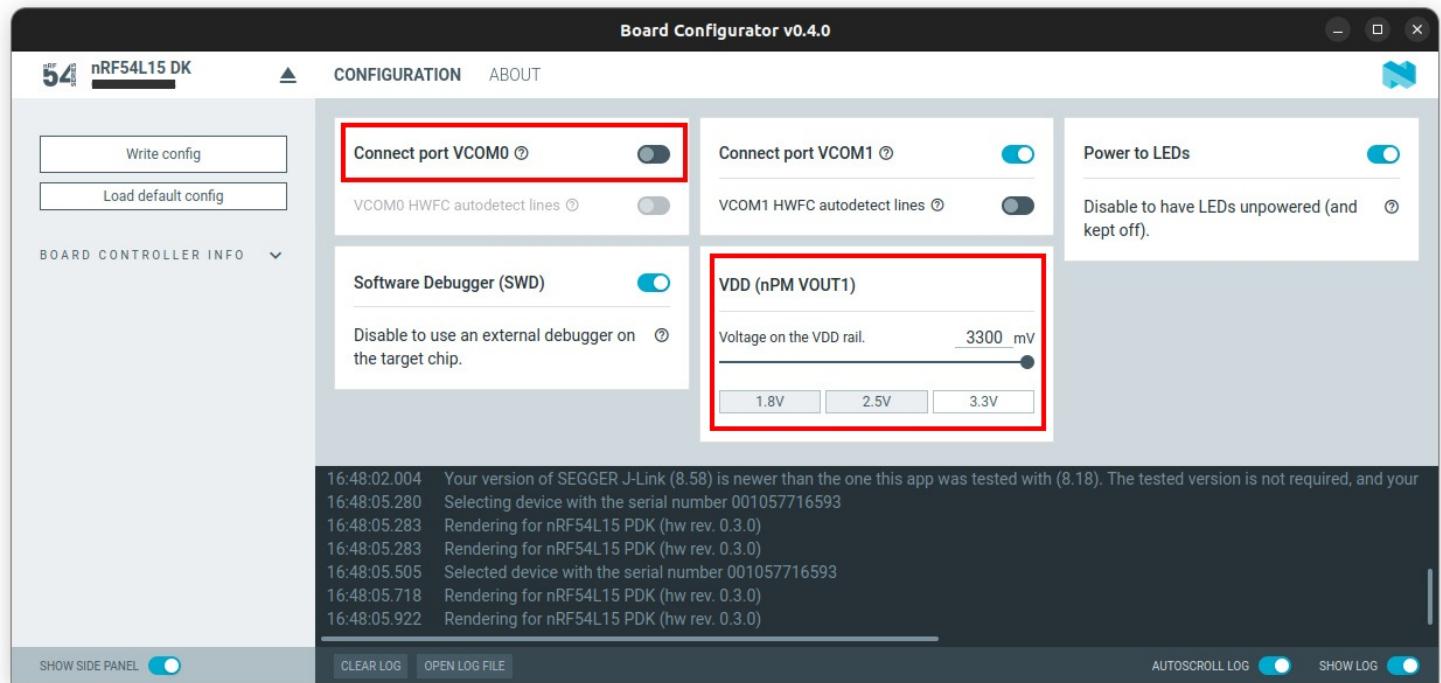
- nRF54L15-DK based kits

To properly handle the Nordic dev kit, you need to install the [nRF Connect for Desktop tool](#) from Nordic website.

Since the nRF54-DK is based on a J-Link debugger, you also need to install the [J-Link driver](#) from Segger website.

Both software packages are available for both Windows and Linux.

Once both software packages are installed, plug the nRF54-DK to your computer. You should see 2 serial ports associated with the device. You will need to deactivate the first one (the one with the lowest number) and set the power voltage to 3.3V as shown below:



Make sure to write back the change by clicking on the *Write Config* button. You can now close the nRF Connect application.

You should now be able to use LoRa Studio with the nRF54-DK based kits.

- Seeeduino XIAO based kits

On Windows, good news : nothing to do! The module is plug and play.

On Linux, you need to create a udev rule to allow non-root access to the device:

Create a file named `/etc/udev/rules.d/99-xiao.rules` with the following content:

```
SUBSYSTEMS=="usb", ATTRS{idVendor}=="2886", ATTRS{idProduct}=="0066", \
    MODE=="0666", \
    SYMLINK+="nrf54"
ATTRS{idVendor}=="2886", ATTRS{idProduct}=="[08]02d", MODE=="0666", ENV{ID_MM_DEVICE_IGNORE}="1", \
    ENV{ID_MM_PORT_IGNORE}="1"
```

[Copy](#)

Then reload the udev rules with the command:

```
sudo udevadm control --reload-rules && sudo udevadm trigger
```

[Copy](#)

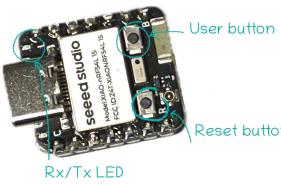
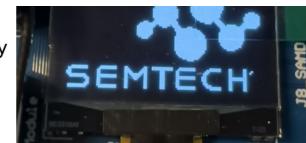
After this, unplug and replug the device. You should now have access to it without needing root privileges.

Using LoRa Studio

Connecting your development kit

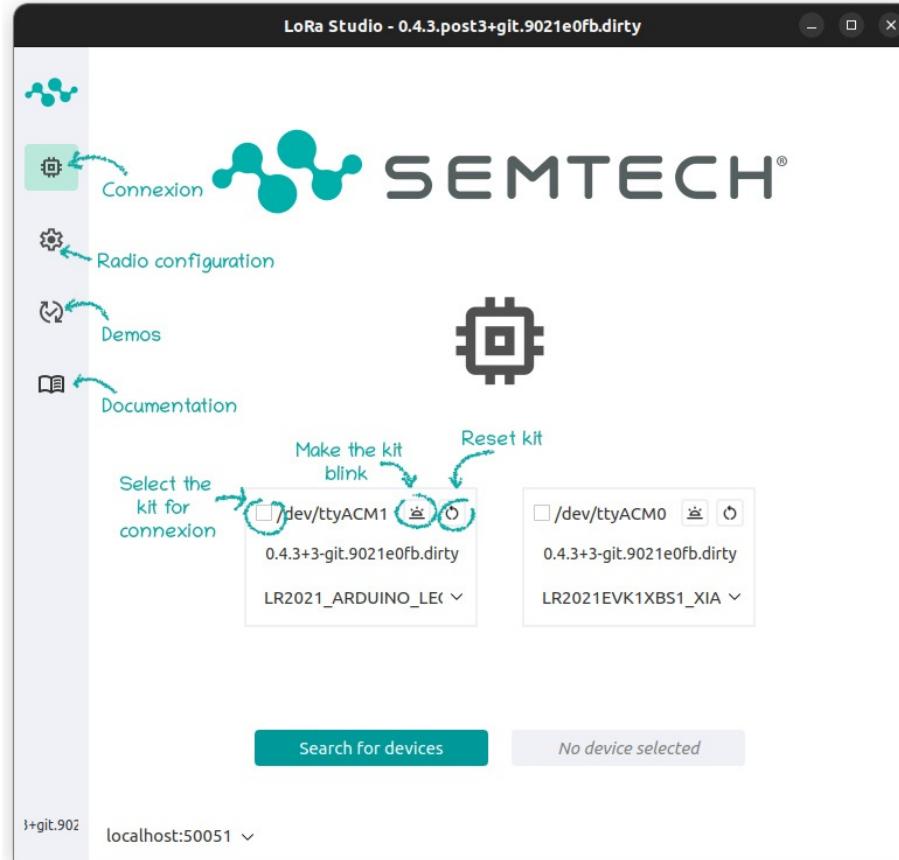


The drivers properly installed, you can connect your kits to your computer. If the kit is equipped with the small OLED screen, you shall see a Semtech logo indicating that the kit is powered on. If it does not appear, please reset manually your kit by pressing the reset button.



Note: If you have a Seeeduino XIAO based kit, you will have to reset with the button on the XIAO board, not the one on the expansion board.

Launch LoRa Studio, here is what it looks like:



At launch, you see the [Connexion Page](#). Until you connect a kit, you can only navigate to the documentation page (this one!) or stay on the connexion page.

If you have multiple kits connected you can identify which serial port is associated with which kit by clicking on the button. This will make LEDs blink on the corresponding kit.

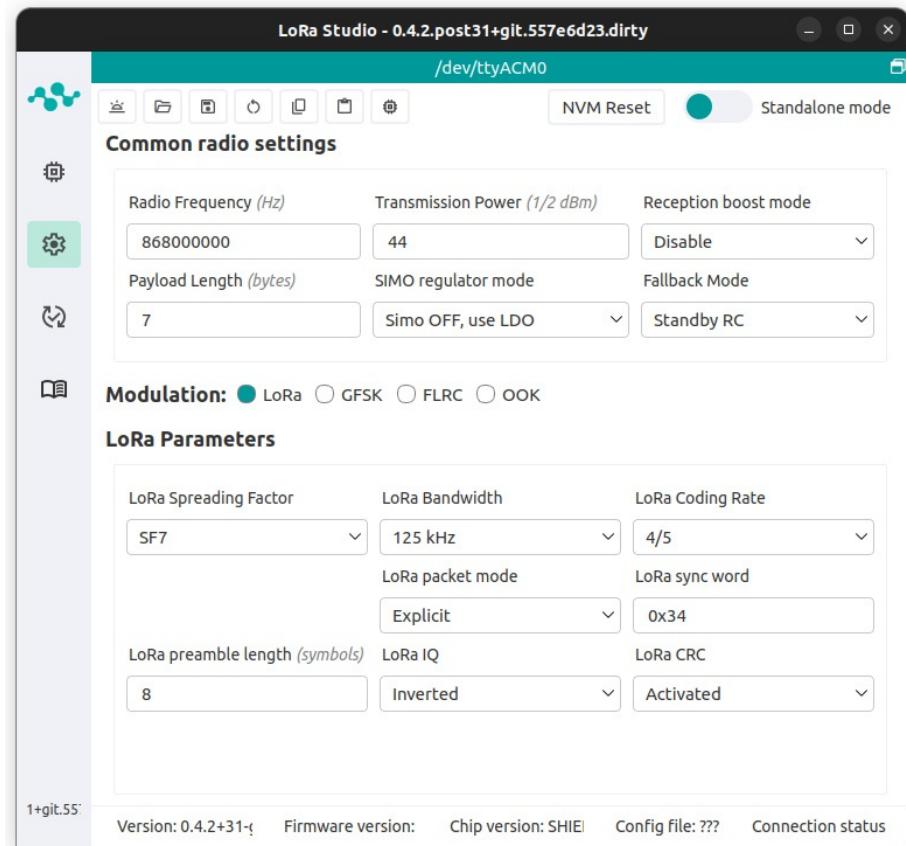
You can also reset your kit using the button.



At startup, LoRaStudio detects the connected development kit(s) and displays for each a card with the kit info as shown above. If no device is detected, make sure that the kit is properly connected and that the drivers are correctly installed or see the [troubleshooting section](#) below.

Important Note about Kit Type: The Kit type is automatically detected **BUT** the part number must be selected manually. If the exact reference is not correctly set, LoRa Studio may not be able to operate the kit's radio properly leading to unexpected behavior. Please verify that the Kit type matches the development kit you are using before proceeding with any operations. Please refer to the [Supported Development Kits](#) section above for the correct kit names.

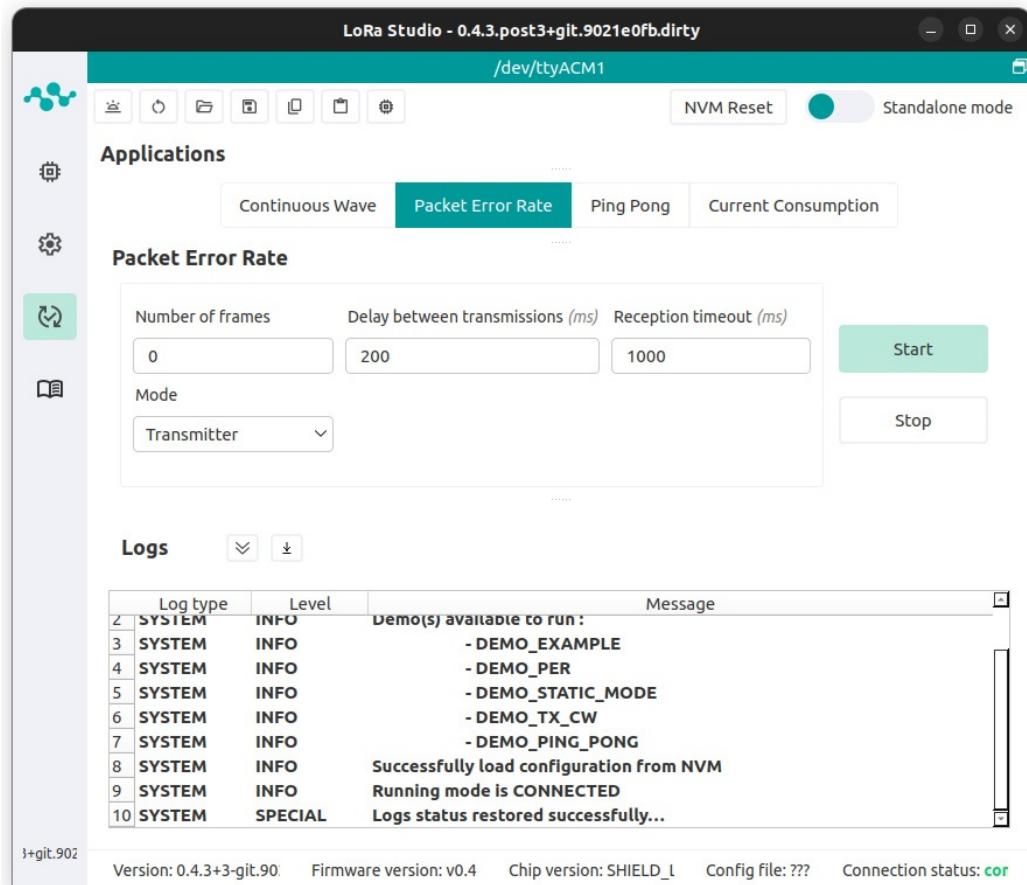
Once connected, LoRa Studio switches automatically to the [Radio Configuration](#) page.



From this page, you can configure the radio parameters of your kit. You can select, among others, the frequency, the output power, the modulation and its parameters.

You can also, like on the connexion page, identify and reset the connected kit using the and buttons.

Once you are satisfied with your configuration, you can go to the [Application](#) page by clicking on the icon in the left menu.



From this page, you can select, set parameters for and run demonstration applications.

You can also, like on the previous pages, identify and reset the connected kit using the and buttons.

Finally, on both the [Radio Configuration](#) and [Application](#) pages, a tool bar allows you to :

- Save the configuration to a JSON file. This will open a dialog to select the location and name of the file.
- Load a configuration from a file.
- Copy the configuration to the clipboard: JSON content can be pasted to wherever you would like or...
- ... paste it on the page of another connected kit to ensure they both use the exact same parameters.
- Update the firmware of the connected kit. This will open a dialog to select the firmware file. The firmware file must be a .bin file and correspond to your kit.



Running demonstration application

From the [Application](#) page, you can select and run different demonstration applications. The available applications are:

- Continuous Wave
- Ping Pong
- PER (Packet Error Rate) Test
- Static modes for current consumption measurement

Continuous Wave Demo

The Continuous Wave demo allows you to transmit a continuous wave signal for testing purposes.

- Once connected to your kit, the application loads and opens automatically the [Radio Configuration](#) page.
- Configure the radio parameters (frequency, power, modulation and modulation parameters)
- Go to the [Application](#) page
- Select Continuous Wave and click on the [Start](#) button
- The kit will start transmitting a continuous wave signal with the configured parameters
- Click on the [Stop](#) button to stop the transmission

Ping Pong Demo

The Ping Pong demo demonstrates bi-directional communication between two devices. This application obviously needs two kits to work with one another. *TIP: Configure only one kit properly then use the copy/paste configuration functionality and change the mode on the second kit afterwards.*

- Once connected to your kit that application loads and open automatically the [Radio Configuration](#) page.
- Configure the radio parameters (frequency, power, modulation and modulation parameters)
- Go to the [Application](#) page
- Select Ping Pong and setup the application parameters:
 - Delay before transmission: set the delay before the first packet is sent in ms
 - Delay between packets: set the delay between the reception and the retransmission of the packet
 - Sync packet threshold: Number of packet to transmit without a response to give enough margin for synchronization
 - Prefix size: Number of bytes used as prefix in the packet
 - Reception timeout: Time in ms to wait for a response before considering that the packet is lost
- Click on the [Start](#) button to start the demo
- The kit will start transmitting packets and wait for a response. If a response is received, it will retransmit the packet after the configured delay. If no response is received within the timeout, it will consider the packet lost and transmit a new one after the configured delay.
- Click on the [Stop](#) button to stop the demo

Packet Error Rate

The Packet Error Rate evaluates link quality and performance metrics. This application needs two kits to work with one set as transmitter and the other as receiver. *TIP: Configure only one kit properly then use the copy/paste configuration functionality and change the mode on the second kit afterwards.*

- Select both your kits on the connection page and click on the Connect button, the application loads and opens automatically the [Radio Configuration](#) page for both kits.
- Configure the radio parameters (frequency, power, modulation and modulation parameters)
- Go to the [Application](#) page
- Select Packet Error Rate and setup the application parameters:
 - Number of frames: Number of frames to receive (as goal) for the reception mode, on which the PER will be calculated. If set to 0, the kit will receive indefinitely and PER will be calculated on the fly.
 - Delay between transmissions: Delay in ms between each frame the transmitter sends
 - Reception timeout: Time in ms to wait before considering a frame lost
 - Mode: Select if the kit will be in transmitter or receiver mode
- Click [Start](#) to begin
- A live calculated PER will be shown in the logs and on the screen for you to see.
- Click on the [Stop](#) button to stop the demo

Static modes for current consumption measurement

This application allows you to set the kit in different static modes to measure the current consumption.

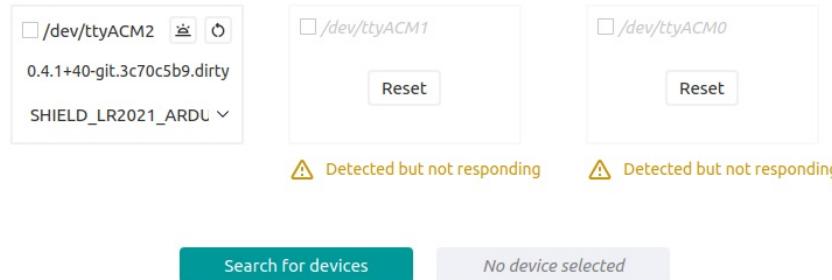
- Once connected to your kit, the application loads and opens automatically the [Radio Configuration](#) page.
- Configure the radio parameters (frequency, power, modulation and modulation parameters)

3. Go to the the [Application](#) page
4. Select Static modes and setup the application parameters:
 - Mode: Select the static mode you want to set the kit in
5. Click on the [Start](#) button to set the kit in the selected static mode
6. The kit will enter the selected static mode
7. Click on the [Stop](#) button to exit the static mode and return to normal operation

Troubleshooting

Common connection issues

- Main page shows a device is "Detected but not responding":



This usually means that the application can see a serial port to communicate with the device, but the device is not responding to commands. It can be that this device is not a supported development kit or the firmware on the kit is not compatible with LoRa Studio. It can also be that the kit has some UART issues. You can try the following steps to resolve the issue:

- Click on the "Search for devices" button.
- Click on the "Reset" button associated to the serial port you believe your board is connected to and then on the "Search" button again.
- Physically reset the kit by pressing the reset button and then on the "Search for devices" button again.

Note: For nRF54L15-DK based kits, even though you made sure that you have deactivated the first serial port in nRF Connect two serial ports will be detected, and one of them will always be not responsive. This does not affect the functionality of LoRa Studio and can be ignored.

- Verify driver installation: Double-check that all necessary drivers are installed and up to date.
- Restart the application: Sometimes, simply restarting LoRa Studio can resolve connection issues.