

n-CORE[®]

Startup Guide

Table of Contents

| | |
|--|----------|
| 1. Environment setup | 3 |
| 1.1. MS Windows setup | 3 |
| 1.2. Linux setup | 4 |
| 2. Devices setup | 5 |
| 2.1. Locating (available in Full License only) | 6 |

1. Environment setup

1.1. MS Windows setup

Step 1: Install the n-Core Sirius devices' driver. The driver is in the `\tools\cp210X` folder.

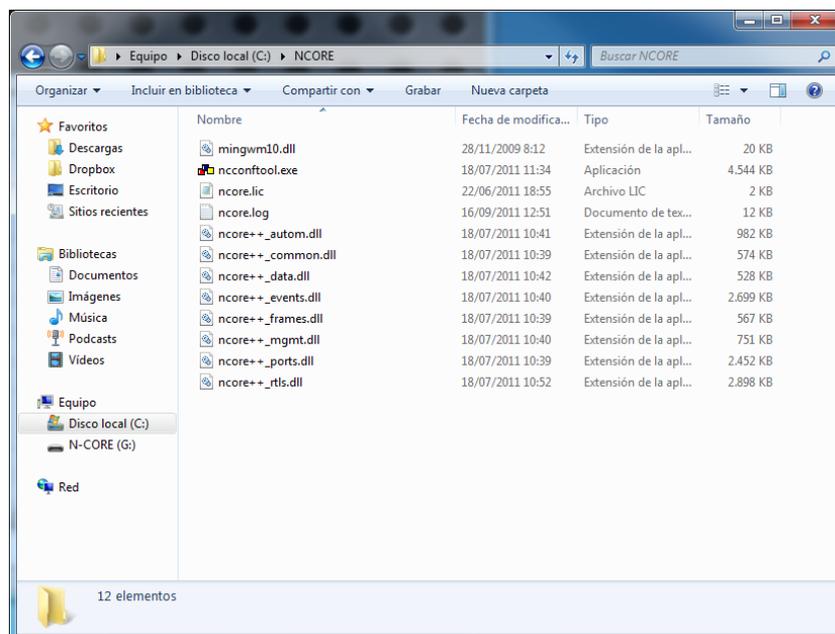
NOTE: you can also download the "CP210x USB to UART Bridge VCP Drivers" from the website of Silicon Laboratories at www.silabs.com

Step 2: Get familiar with the APIs. There are three different APIs in the Windows version:

1. **ncore++:** it is the main and native API of n-Core. It manages all the communication between the PC and the n-Core Sirius devices. All the functionalities are offered in C++ language.
2. **ncore.c:** It is a *wrapper* of ncore++ and offers all the functionalities in C language, easily linked from other languages such as Java or Python.
3. **ncore.net:** It is a *wrapper* of ncore.c and offers all the functionalities in .NET C# language.

Step 3: Setup the n-Core Configuration Tool. This application allows configuring the n-Core Sirius devices. Copy the following files into a new folder, as indicated in the screenshot:

- `ncconftool.exe` (from `"/platform/ncconftool/bin/win32"`).
 - `"/platform/ncconftool/bin/win32/basic/ncconftool_basic.exe"` if you have a Basic license.
- ncore++ API DLLs (from `"/platform/ncore++/lib/win32"`).
- `ncore.lic` (provided by your n-Core distributor).
- `mingwm10.dll` (available on n-Core website or in the MinGW distribution at www.mingw.org).



IMPORTANT NOTE: The license file is unique for every client. Therefore, it is not allowed to configure devices with a different client ID.

Step 4: Execute the n-Core Configuration Tool ("`ncconftool.exe`" or "`ncconftool_basic.exe`" file).

1.2. Linux setup

Step 1: Be sure your Linux distribution has a valid n-Core Sirius devices' driver. Linux distributions including 2.6 series kernel ($\geq 2.6.12$) usually provide CP210X driver as part of the default installation.

If not present, install the n-Core Sirius devices' driver. The driver is in the `"/tools/cp210X/linux"` folder in your n-Core files.

NOTE: you can also download the "CP210x USB to UART Bridge VCP Drivers" from the website of Silicon Laboratories at www.silabs.com

Step 2: Get familiar with the APIs. There are two different APIs in the Linux version:

1. **ncore++:** it is the main and native API of n-Core. It manages all the communication between the PC and the n-Core Sirius devices. All the functionalities are offered in C++ language.
2. **ncore.c:** It is a *wrapper* of ncore++ and offers all the functionalities in C language, easily linked from other languages such as Java or Python.

Step 3: Setup the n-Core Configuration Tool. This application allows configuring the n-Core Sirius devices. In order to setup it, follow the next steps:

- Install ncore++ libraries in your system. Please, read your system documentation about installing libraries in your system. You may need *superuser* permissions to perform these tasks (consult your system administrator and read the "sudo" command manual: type `"man sudo"` in your console).
 - Copy the dynamic libraries (".so" files) from the `"/platform/ncore++/linux/lib"` directory into your system (typically to `"/usr/local/lib"` directory).
 - Copy the headers of (".h" files) from the `"/platform/ncore++/include"` directory into your system (typically `"/usr/local/include"` directory). Example in your console:
 - `sudo mkdir /usr/local/include/ncore++`
 - `sudo cp -R /YOUR_PATH/platform/ncore++/include/* /usr/local/include/ncore++`
 - Use the `"ldconfig"` command to properly install libraries. Please, read the manual of `"ldconfig"` (type `"man ldconfig"` in your console).

NOTE: You can do these steps also for ncore.c libraries, nevertheless this is not necessary for running `ncconftool`.

- Be sure that wxWidgets 2.8 libraries are installed in your system. If not, consult your system administrator about this to install wxWidgets libraries, typically running `"sudo apt-get install wx2.8"` in your console (for Debian-based distributions, such as Ubuntu).
- Install `"ncconftool"` executable in your system (typically copying the `"/platform/ncconftool/bin/linux/ncconftool"` file into your `"/usr/local/bin"` directory).
 - If you have a Basic n-Core license: `"/platform/ncconftool/bin/linux/basic/ncconftool_basic"`
- Be sure that your n-Core license file (`ncore.lic`) is in your current execution directory.

IMPORTANT NOTE: The license file is unique for every client. Therefore, it is not allowed to configure devices with a different client ID.

Step 4: Execute the n-Core Configuration Tool (`"ncconftool"` or `"ncconftool_basic"` file).

2. Devices setup

All the *n-Core Sirius* devices can be configured to perform the following roles:

- **Coordinator:** Creates and manages the ZigBee network. There must be only one *coordinator* in the network.
- **Collector:** Centralizes the information sent by one or more *n-Core Sirius* devices. It is necessary to configure the rest of the nodes in the network in order to establish the redirection to a specific *collector*.
- **Router:** Retransmits the information between the different nodes in the network. In addition, it is used as a reference point by the locating engine.
- **Tag:** It is used as a mobile device by the locating engine. It has the ability to enter in *sleep mode*, that is, remain in a state of low power consumption and without transmitting information over a configurable time period.

Step 1: Find the coordinator. The serial number of the coordinator is printed in the documentation of your Development Kit.

Step 2: Connect the coordinator to the PC via the USB. Then, turn on the device and all the *n-Core Sirius* devices to be configured. The coordinator allows configuring the rest of the nodes via OTA (*Over-The-Air*).

NOTE: The connection can be done via Sockets also (only on *n-Core* version 1.7 and above).

NOTE: Each device can be configured via the USB separately following the next steps.

Step 4: Execute the n-Core Configuration Tool and establish the connection. All the *n-Core Sirius* devices connected to the application will be shown in the tree of nodes. In the case of the coordinator, all the devices in the same network (i.e., the network created by the coordinator) will be shown.

Step 6: Obtain the configuration information from a specific node. You can either:

- Select a node in the tree of nodes and press the "Get Params" button, or
- Double click the node in the tree of nodes.

The application shows the selected node's current configuration stored in its EEPROM.

Step 7 (optional): Be sure that the Work Mode is activated. It is activated by default.

- The *Work Mode* can be activated in the "Network" tab. For more information about the working modes, please read the *Working Modes Manual*.

Step 8 (optional): Change the network parameters. Create your own ZigBee network by changing the following parameters: *Pan Id*, *Extended Pan Id*, *Channel Page* and *Channel Mask*.

- Once you assign the new values, press the "Set Params" button to store the data into the device's memory.
- Finally, reset the device.

IMPORTANT NOTE: All devices in the ZigBee network must be equal for these four parameters. It is recommended to configure the coordinator at the end of the process. This way, the setup network will remain available for the rest of the devices.

Step 9 (optional): Change the device's role. As stated previously, all the *n-Core Sirius* devices can be configured to perform the following roles: *Coordinator*, *Collector*, *Router* and *Tag*.

- Go to the "Application Device Type" parameter in the "Advanced" tab.

- Get the parameters of the node by selecting it in the tree of nodes and pressing the "Get Params" button.
- Modify the "Device Type" parameter according to the new role.
- Once you assign the new values, press the "Set Params" button to store the data into the device's memory.
- Finally, reset the device.

2.1. Locating (available in Full License only)

The **n-Core Locating Engine** allows obtaining the position of the devices configured as *tags*. *Tags* periodically send a broadcast signal which is collected by the devices configured as *routers* and *collectors*. The broadcast signals are stored in the "Tags Tables" which are used to calculate the positions of the *tags*.

To enable the locating features, please follow these steps:

Step 1: Connect the *coordinator* to the computer.

Step 2: Turn on the device and all the *n-Core Sirius* devices to be configured.

Step 3: Run the *n-Core Configuration Tool*.

Step 4: Check the role of the device to be configured (see Step 9 on previous section).

Step 5: Modify the parameters of the device:

- In the case of a *router*: modify the coordinates and the "Tags Table Period". The "Tags Table period" must be greater than 0.
- In the case of a *tag*: modify only the "Broadcast Period" (greater than 0).

Step 6: Press the "Set Params" button to store the data into the device's memory. Reset the device.

Step 7: In the case of a *tag*, select the node and get their current settings. Modify the "Sleep Period" (period that the *tag* will remain in *sleep mode*).

Step 8: Press the "Set Params" button to store the data into the device's memory. Reset the device.

Step 9: Test the locating features: select the "Register Messages" option. The *n-Core Configuration Tool* will show the "Tags Table" of every router and also the RSSI and LQI values received.



www.n-core.info