



BFi91XTD DECT NR+ Evaluation Kit

Getting Started & Integration Overview

Setting up the Thingy:91 X kit over USB or Bluetooth LE · Wi-Fi and cloud · mesh networks · sensors · security keys · firmware updates

PUBLIC EDITION

Document	FLS-GS-XTD-EN, public edition 1.2
Date	October 2026
Applies to	FLSEMI October 2026 firmware for the Thingy:91 X kit; host tool <code>flsemi</code>
Audience	IT, system-integration and software staff evaluating the kit
More detail	For the full Setup and System Integration Guide and the host-interface documentation, leave your company details at flsemi.us/downloads#sdk .

© 2026 FL Semiconductor LLC (FLSEMI). Nordic Semiconductor, Thingy:91 X, nRF9151, nRF5340 and nRF7002 are trademarks of Nordic Semiconductor ASA. ThingsBoard is a trademark of ThingsBoard, Inc. WireGuard is a registered trademark of Jason A. Donenfeld.

Contents

1 About this overview

2 The kit at a glance

3 Setting up the workstation

4 Network uplink

4.1 Wi-Fi · 4.2 ThingsBoard · 4.3 WireGuard

5 Building a DECT NR+ network

5.1 Concepts · 5.2 Bands · 5.3 Mesh · 5.4 Star · 5.5 Site plan

6 Security keys

7 Sensors

8 Bluetooth LE

9 Integrating with your software

10 Firmware updates

11 Troubleshooting

12 Limits and regulatory notes

1 About this overview

This overview is for the people who set up a BFi91XTD kit on a site network and connect it to their own software. It shows how to:

- prepare a workstation and talk to the kit over USB or Bluetooth LE;
- bring the kit onto Wi-Fi and your own ThingsBoard server, optionally through a WireGuard tunnel;
- build a DECT NR+ mesh network and distribute a site plan;
- install a network security key;
- read sensor data, locally and from the cloud;
- update the firmware of both processors.

For the full Setup and System Integration Guide, with the complete command reference and the host-interface documentation, leave your company details at flsemi.us/downloads#sdk.

Evaluation kit. The kit is supplied under 47 CFR §2.803 to product developers, software developers and system integrators for evaluation only. It is not an FCC-authorized end product. FLSEMI's DECT NR+ firmware has not yet been through DECT NR+ certification or formal conformance testing.

2 The kit at a glance

A BFi91XTD kit is a Nordic Semiconductor Thingy:91 X running two FLSEMI firmware images:

Processor	Job
nRF5340 — FLSEMI gateway firmware	USB, Wi-Fi (nRF7002), Bluetooth LE, the cloud client and the WireGuard tunnel, the configuration commands, the on-board sensors and power management. It drives the DECT NR+ stack over an internal SPI link.
nRF9151 — FLSEMI DECT NR+ stack	The DECT NR+ MAC, DLC and Convergence layers (ETSI TS 103 636-4/-5), mesh routing and link security.

You configure everything through the nRF5340. The gateway also keeps a **profile** (network, carrier, role, report period) and re-applies it to the nRF9151 whenever that chip restarts.

Interface	What it carries
USB, first serial port	Configuration commands and firmware updates
USB, second serial port	A diagnostic command shell
Wi-Fi (IPv6 only)	Uplink to your ThingsBoard server, remote commands, the WireGuard tunnel
Bluetooth LE	The same configuration commands and firmware updates, and the command shell over the Nordic UART Service (chapter 8)
DECT NR+	The mesh or star network

Data flows like this:

- Every DECT NR+ node sends a sensor report every N seconds. The **sink** — the node whose gateway has the uplink — collects the reports and posts one ThingsBoard device per node.
- Network-wide settings (the site plan, the report period, the security policy and the IPv6 prefix) are written once at the sink and distributed over the air.

3 Setting up the workstation

3.1 Install the tool

Python 3.9 or later on macOS, Windows or Linux. The host tool is open source (BSD-3-Clause):

```
python3 -m pip install --upgrade "git+https://github.com/flsemi/flsemi-xtd"
flsemi version
```

- **macOS:** no driver needed. For Bluetooth, run the tool from Terminal.app so the permission prompt can appear.
- **Windows 10/11:** the kit appears as two COM ports.
- **Linux:** add your user to the `dialout` group, and keep ModemManager away from `/dev/ttyACM*`.

The tool talks only to the device on the cable or radio, and never to an FLSEMI server.

3.2 Find the ports and make first contact

```
flsemi cfg devices          # what is attached; the port shown is PORT
flsemi cfg -d PORT status   # firmware, link to the nRF9151, Wi-Fi state
flsemi cfg -d PORT chipid   # identity of both processors, including the RD ID
flsemi cfg -d PORT inventory # parts on the board, and whether Bluetooth is built in
flsemi ui -d PORT           # local dashboard at http://127.0.0.1:8765 (read-only)
```

- The second serial port of the same kit is the shell (SHELLPORT). It never answers configuration commands.
- Only one program can open a port at a time.
- Record the **RD ID** from `chipid`. It names the node on the air, in site plans and in ThingsBoard (`nrplus-<rdid>`).

4 Network uplink

4.1 Wi-Fi

The kit firmware is **IPv6-only**, following ETSI TS 103 874-3. The Wi-Fi network must provide router advertisements (SLAAC) and an IPv6 route to your server.

```
flsemi cfg -d PORT scan
flsemi cfg -d PORT wifi --ssid "SiteNet" --psk 'secret' --sec wpa2
flsemi cfg -d PORT wifi --radio on          # default "auto" = Wi-Fi on only when this kit is the
sink
flsemi cfg -d PORT adv --reg US
flsemi cfg -d PORT ip                      # the IPv6 address in use
```

The Wi-Fi password is write-only.

4.2 ThingsBoard over CoAP

The kit ships with no cloud connection. Point it at your own ThingsBoard server, with the CoAP transport on UDP 5683:

```
flsemi cfg -d PORT cloud --host tb.example.com --token GATEWAY_ACCESS_TOKEN --interval 30
flsemi cfg -d PORT cloud --pkey PROVISION_KEY --psec PROVISION_SECRET # node devices created
automatically
flsemi cfg -d PORT cloud                                           # state and delivery
counters
```

The CoAP uplink is not encrypted by itself. Without a tunnel, telemetry and device tokens are sent as plaintext CoAP. Use the WireGuard tunnel (4.3) unless the path to your server is a network you control.

4.3 WireGuard tunnel

The gateway contains a WireGuard client. When the tunnel is up and your ThingsBoard host is an address inside it, the uplink travels through the encrypted tunnel to your own WireGuard server.

1. On your WireGuard server, choose a tunnel prefix, for example `fd00:1234::/64`, with the server at `fd00:1234::1`.
2. Configure the kit:

```
flsemi cfg -d PORT vpn --endpoint vpn.example.com --port 51820 --server-pub SERVER_PUBLIC_KEY \
--my-addr fd00:1234::12/64 --allowed fd00:1234::/64 --keepalive 25 --enable on
```

3. The kit generates its own key pair. Read its public key on the shell:

```
flsemi sh -d SHELLPORT 'vpn down'
flsemi sh -d SHELLPORT 'vpn show'
flsemi sh -d SHELLPORT 'vpn up'
```

4. Add that key to the server as a peer with `AllowedIPs = fd00:1234::12/128`.
5. Point the uplink into the tunnel: `flsemi cfg -d PORT cloud --host fd00:1234::1`.
6. Check the server's `wg show` for a recent handshake.

If the tunnel goes down, the uplink stops. It never falls back to plaintext on the open internet.

5 Building a DECT NR+ network

5.1 Concepts

Term	Meaning
Network ID	32-bit ID shared by every node of one network
Carrier	The DECT NR+ channel number. It also selects the band.
Sink	The root of a mesh, with the uplink
Relay	A node that joins a parent and serves children of its own
Leaf	A node that only joins
Parent preference	A preferred parent. A node falls back to any parent if the preferred one is lost.

5.2 Bands

United States: use **Band 9** (1.9 GHz, carriers **1703–1711**) or **Band 4** (902–928 MHz, carriers **525–551**, every second carrier). Bands 1, 2 and 22 are not available in the US. Keep every carrier you configure inside these ranges.

Both bands have run on air on the Thingy:91 X and the nRF9151 DK. Band 4 throughput and range are not yet characterised. One network runs on one band.

```
flsemi cfg -d PORT dect # includes the bands this board supports
flsemi cfg -d PORT dect --band 9 # stores the middle carrier of Band 9
```

5.3 Mesh network

Configure the sink first, then the relays outward, then the leaves.

```
# sink
flsemi cfg -d PORT dect --network 0x12340001 --band 9 --carrier 1707 --sink on --autostart on --sense 30
flsemi cfg -d PORT dect --role sink --role-carrier 1707 --hop 5
flsemi cfg -d PORT dect --snapshot

# relay: joins on 1707, serves its children on 1703
flsemi cfg -d PORT dect --network 0x12340001 --carrier 1707 --sink off --autostart on --sense 30
flsemi cfg -d PORT dect --role relay --role-carrier 1707 --ft-carrier 1703 --snapshot

# leaf: joins the relay on its serving carrier
flsemi cfg -d PORT dect --network 0x12340001 --carrier 1703 --sink off --autostart on --sense 30
flsemi cfg -d PORT dect --role leaf --role-carrier 1703 --snapshot
```

- Always pass `--hop` when writing the sink role. It sets how deep the mesh may grow.
- To pin a child to a relay, give the child the carrier the relay serves on, and its RD ID with `--parent`.
- Check the tree with `flsemi profile -d PORT site` on the sink, or with the `topology` attribute in ThingsBoard.

5.4 Star network

The stack also runs as a star: one hub and members that talk to it directly, with no forwarding. Star operation is selected per node before its role starts. The procedure is in the full guide for kit holders.

5.5 Site plan

The sink distributes a **site plan** over the air (TS 103 636-5 Annex C), and every node adopts it hop by hop. The plan can set which nodes relay and on which carrier, each node's parent, sensor reporting, the security policy, and the mesh IPv6 prefix with its service addresses (TS 103 874-3).

```
{
  "site": "Building A",
  "relays": [ {"id": "0x23456789", "ft_carrier": 1703, "parent": "0x12345678", "join_carrier": 1707} ],
  "nodes": [ {"id": "0x3456789a", "parent": "0x23456789", "join_carrier": 1703} ],
  "sense": { "period_s": 30, "routed": true, "dst": 0 },
  "policy": { "sec_require": true },
  "ipv6": { "prefix": "fd00:1:2:3::" },
  "compact": true
}
```

```
flsemi cfg -d PORT plan site.json      # publish from the sink
flsemi cfg -d PORT plan                # what is stored and live
flsemi cfg -d PORT plan --lock on      # ignore the ThingsBoard copy
```

The plan can also come from ThingsBoard, as the gateway device's shared attribute `site_plan`. The last editor wins unless the lock is on, and `diverged` shows when the two copies disagree.

6 Security keys

DECT NR+ link security follows Mode 1 of TS 103 636-4: AES-CMAC integrity and AES-CTR ciphering, derived from a 128-bit network key that every node of the network shares.

Security is off by default. Links are unsecured until you install a key.

```
python3 -c "import secrets; print(secrets.token_hex(16))" # one random key per network
flsemi cfg -d PORT sec --key 32_HEX_DIGITS --save # on every node
flsemi cfg -d PORT sec --require on # refuse unsecured
associations
flsemi cfg -d PORT sec # profile_has_key, require
```

When a key is installed:

- The key is **write-only** on every interface. `--save` keeps it in the nRF9151's secure storage, encrypted with the chip's hardware-unique key.
- Install the key on every node before the nodes associate. Keys are not sent over the air.
- The stack also supports ciphered beacons and key rotation without downtime. The procedures are in the full guide.
- Link security protects the DECT NR+ radio hops. The gateway deciphers, and the WireGuard tunnel protects the gateway-to-server leg.
- Factory-resetting the nRF9151 erases its key. Install it again afterwards.

7 Sensors

Key	Unit	Source
temp, hum, press, gas	°C, %RH, hPa, Ω	Environmental sensor
iaq, co2, voc, iaqAcc	index, ppm eq., ppb eq., 0–3	Air-quality algorithm. These keys appear after the sensor's run-in; iaqAcc 3 = fully calibrated.
roll, pitch, heading, moving	degrees, flag	Motion sensors and magnetometer
vbat, chg, on_batt	mV, state, flag	Battery and charger
die, supply	°C, mV	nRF9151, reported by every node
seq, up, boot, rst, fault	—	Health of each node

```
flsemi cfg -d PORT sense # latest sample per node
flsemi cfg -d PORT motion # attitude and motion of this kit
flsemi cfg -d PORT sense --temp-offset auto # corrects self-heating when Wi-Fi is on
```

- **Report period:** `dect --sense N` per node, or `"sense": {"period_s": N}` in the site plan.
- **From the cloud:** each node is a ThingsBoard device `nrplus-<rdid>` carrying these keys. Read them with ThingsBoard's REST or WebSocket API.

8 Bluetooth LE

The kit firmware includes Bluetooth LE. Over it the kit offers the same configuration commands over Bluetooth LE, plus a command shell and firmware updates.

- **Opening the window.** The kit does not advertise until a window is opened, either with `flsemi cfg -d PORT ble -on` (10 minutes by default, up to 60) or by pressing **Button 2** twice (5 minutes). LED2 breathes blue while the window is open. The kit advertises as `BFi53-<rdid>`.
- **Pairing.** LE Secure Connections pairing is required before anything answers. Accept the pairing prompt on first use. Up to three bonds are kept; `ble --unpair` forgets them.
- **Configuration.** Any command with `-d ble:<rdid>`, for example `flsemi cfg -d ble:12345678 status`. The dashboard works too: `flsemi ui -d ble:12345678`.
- **Command shell over NUS.** The diagnostic shell is also available over the Nordic UART Service (service `6e400001-b5a3-f393-e0a9-e50e24dcca9e`). Write command lines ending in a carriage return to `...0002`; the output arrives as notifications on `...0003`. Any NUS terminal app that bonds with the device can be used.

- **Firmware updates (FOTA).** `dfu`, `dfu91` and `ota` (chapter 10) work over Bluetooth with `-d ble:<rdid>`. Nordic Semiconductor's nRF Connect Device Manager app can also update the nRF5340. An nRF5340 update resets the gateway, which closes the window: open a new one before confirming the new image.
- **Limits.** One connection at a time. A window closes by itself when its time runs out.

9 Integrating with your software

9.1 Local control

`flsemi` commands are plain CLI calls that return non-zero on failure, so they script cleanly. Under the tool, every command is a Simple Management Protocol (SMP) request with a CBOR payload, sent over USB serial or Bluetooth GATT. The open-source tool is the reference implementation if you want to call it from your own code.

```
flsemi profile -d PORT dump -o kit-12345678.json      # every setting except secrets
flsemi profile -d PORT diff kit-12345678.json
flsemi profile -d PORT restore kit-12345678.json --psk 'wifi-secret'
flsemi profile -d PORT site -o site.json             # who is attached to whom, and health
```

9.2 ThingsBoard data model

Device	Telemetry	Attributes
Gateway (your token)	Its own sensors, <code>uptime</code> , <code>nodes</code>	Firmware versions, <code>modem_link</code> , <code>position</code> , <code>sink</code> , <code>topology</code> , <code>plan_source</code> , <code>plan_lock</code> , <code>plan_diverged</code>
<code>nrplus-<rdid></code>	That node's report (chapter 7)	<code>rdid</code> , <code>gateway</code>

The gateway also answers ThingsBoard server-side RPC. You can send management commands to a node (reboot, report now), re-publish the site plan, and stop forwarding a node. Use ThingsBoard's inactivity alarms to detect a node going silent.

9.3 Network requirements

Direction	Protocol
Kit → ThingsBoard	UDP 5683 (CoAP), IPv6 — direct or inside the tunnel
Kit → WireGuard server	UDP (your port), IPv6
Kit ↔ router	IPv6 router advertisements (SLAAC); no IPv4

No inbound connection to the kit is needed.

10 Firmware updates

Both processors update over USB (or Bluetooth, chapter 8) without a debugger. The settings survive: Wi-Fi, cloud, profile, and the nRF9151 key. Use only the images FLSEMI supplies for your board. Keep power connected while LED2 is white.

10.1 nRF5340 gateway — test first, then confirm

```
flsemi cfg -d PORT dfu                                # slots; the running image must be
confirmed
flsemi cfg -d PORT dfu gateway.signed.bin --test --reset
flsemi cfg -d PORT status                             # after it re-enumerates: new version?
flsemi cfg -d PORT dfu --confirm                       # only when it works
```

An unconfirmed image reverts to the next reset, so verify before you confirm.

10.2 nRF9151 DECT NR+ stack

```
flsemi cfg -d PORT dfu91 stack.signed.bin
flsemi cfg -d PORT status                # link: true
flsemi cfg -d PORT sec                   # key still present
```

- The nRF9151 has a single application slot. If a transfer is interrupted, send the image again.
- If the transfer times out on the first attempt, run `flsemi sh -d SHELLPORT 'nrf91 reset'` and send it again.

10.3 Across the mesh

A sink can hand a new nRF9151 image to the nodes of its mesh over the air. Each node pulls the image, installs it and serves its own children.

```
flsemi cfg -d PORT ota stack.signed.bin      # on the sink
flsemi cfg -d PORT ota --board thingy --client on  # on each kit that should take it
```

11 Troubleshooting

Symptom	Check
"No response" from the tool	You may have given the shell port, or another program has the port open.
status shows the nRF9151 link down	<code>flsemi cfg -d PORT sys --reset modem</code> . After a failed dfu91, send the image again.
On Wi-Fi, but nothing reaches the cloud	Does <code>ip</code> show a global IPv6 address? Look at <code>cloud counters</code> and <code>last_err</code> . With a tunnel, check the server's <code>wg show</code> .
A node never joins	The network ID and carrier must match. If security is required, the key must be installed on both ends.
Carriers revert after a change	The site plan or the gateway profile still names the old carriers. Change it there.
Plan from ThingsBoard not applied	<code>plan: source, lock, diverged</code>
Bluetooth: the kit cannot be found	Open a window first (Button 2 twice, or <code>ble --on</code>).
Bluetooth: pairing keeps failing	Forget the device on the phone or computer, then try again.

12 Limits and regulatory notes

- IPv6 only.
- No FLSEMI-hosted cloud: you run your own ThingsBoard server.
- The CoAP uplink is not encrypted by itself; use the WireGuard tunnel. No DTLS.
- Link security is off until you install a key, and each node is keyed individually.
- No throughput or range figures are stated for the kit. Band 4 is not yet characterised.
- The host interface and the Embedded SDK are supported only on nRF91 devices and related products shipped by FLSEMI, or shipped under licence from FLSEMI.
- No GNSS while DECT NR+ runs: node positions are what you enter (`flsemi cfg geo --lat 37.0 --lon -95.0 ...`).
- Bluetooth allows one connection at a time and only inside an open window.
- Battery runtime and temperature range are not yet characterised.
- Firmware is not offered for download; updates are sent to kit holders on request.

Regulatory. The kit is supplied under 47 CFR §2.803 to product developers, software developers and system integrators for evaluation only, and is not for resale or for use in a deployed product or service. Its board is marked “For evaluation only; not FCC approved for resale.” It is not an FCC-authorized end product. FLSEMI's DECT NR+ firmware has not yet been through DECT NR+ certification or formal conformance testing. In the United States, use Band 9 (carriers 1703–1711) or Band 4 (carriers 525–551) only.

FL Semiconductor LLC (FLSEMI) · 400 W Emmett St, Suite 13, Kissimmee, FL 34741 · hello@flsemi.us · flsemi.us/dect-eval-kit · Thingy:91 X is a product of Nordic Semiconductor ASA, bought by FLSEMI through regular distribution. Specifications are subject to change.