

XDK LoRa-Extension

Quick Start Guide

Document revision	1.1
Document release date	07.11.2018
XDK LoRa-Extension Version	1.0
Notes	Data in this document is subject to change without notice. Product photos and pictures are for illustration purpose only and may differ from the real product's appearance.

The XDK LoRa-Extension is a wireless and secure communication solution that can exchange data up to a range of 40 km. Public or private LPWANs can be joined or created. For more information, see our XDK LoRa-Extension 2-pager (xdk.io/extensions or lora-alliance.org/)

In advance, install the workbench, version 3.5.0 on a PC (see xdk.io/learning – workbench installation).

Structure of the quick start guide:

- A. Connecting the XDK LoRa-Extension to the XDK
- B. Flashing of the LoRaThingsNetworkDemo Firmware
- C. The things network (TTN)
- D. Integration into Cayenne
- E. Links
- F. Document History and Modification

A. Connecting the XDK LoRa-Extension to the XDK

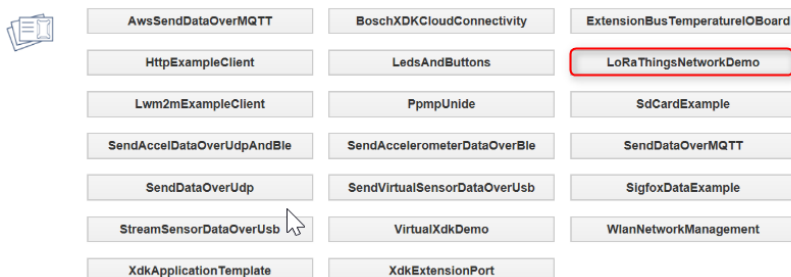
Plug the XDK LoRa-extension to the XDK. It is possible to screw a monitoring plate underneath both devices for a better fixation. The downloading file for the 3D-printer is available on XDK.io/extensions



B. Flashing of the LoRaThingsNetworkDemo Firmware

Use the workbench, version 3.5.0, to flash the LoRaThingsNetworkDemo firmware (see highlighted below) on the XDK

XDK-Examples

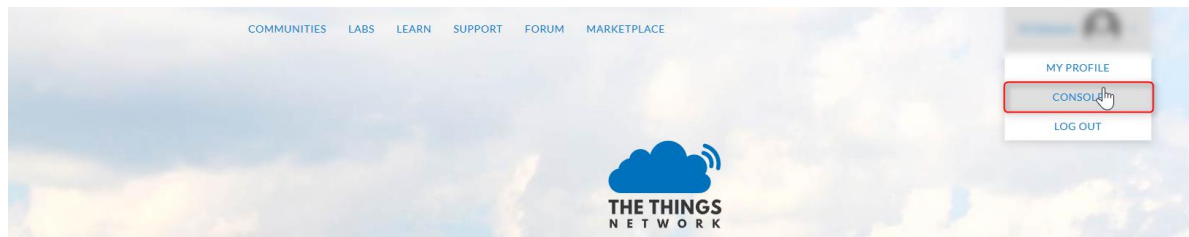


Flashing: Select LoRaThingsNetworkDemo then press Flash

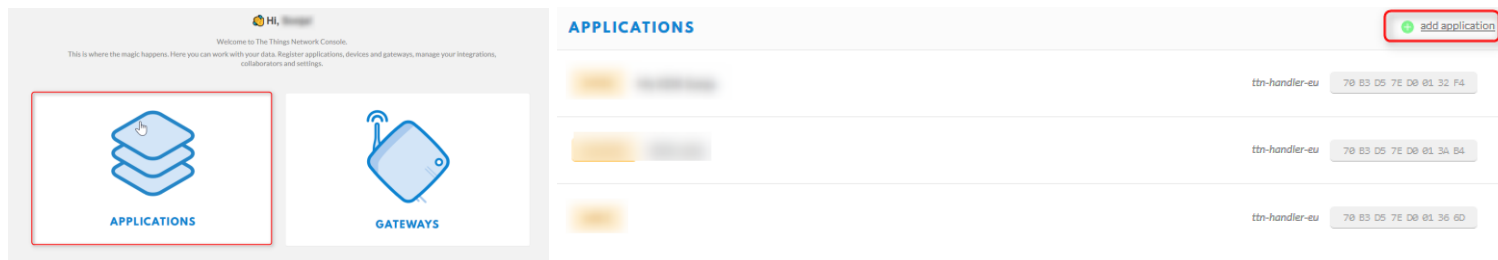


C. The things network (TTN)

1. Register at TTN with username, email and password then activate your account via mail (see link below)
2. Log in and press console

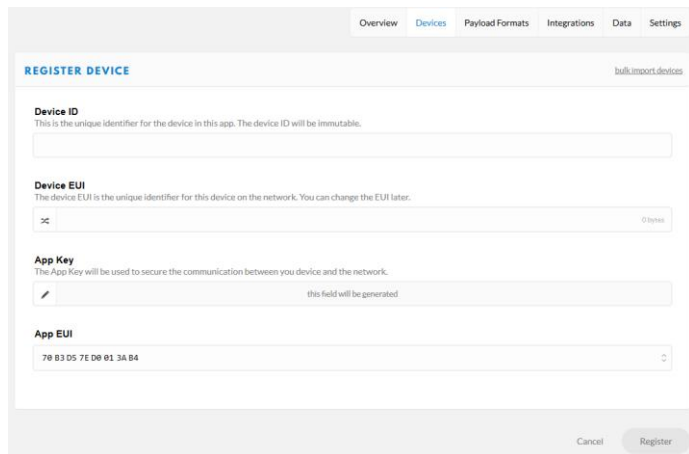


3. Press Applications, then add an application and enter details



4. Register a device

- Device EUI: Boot the application and copy the Hardware Device EUI from the console in the workbench to the field in the TTN



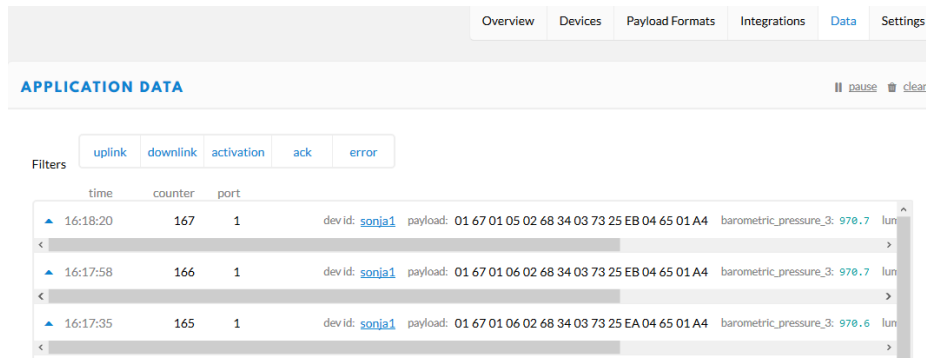
- App EUI: Define the `App_LoRa_App_EUI` in the header file `AppController.h` with the given App EUI by copying the code
- App Key: Define the `App_LoRa_App_Key` in the header file `AppController.h` with the generated App Key

```

55 //
56 * APP_LORA_APP_EUI is unique to the Application Server and each Application Server will have its own AppEUI
57 */
58 #define APP_LORA_APP_EUI    0x0123456789ABCDEF
59 //
60 * APP_LORA_APP_KEY is the data encryption key used to "encode" the messages between the end nodes and the Application Server
61 */
62 #define APP_LORA_APP_KEY \
63     0xAA, 0x55, 0xAA, 0xAA, 0x55, 0xAA, 0xAA, 0x55, 0xAA, 0xAA, 0x55, 0xAA, 0xAA, 0x55, 0xAA, 0x55
64
    
```

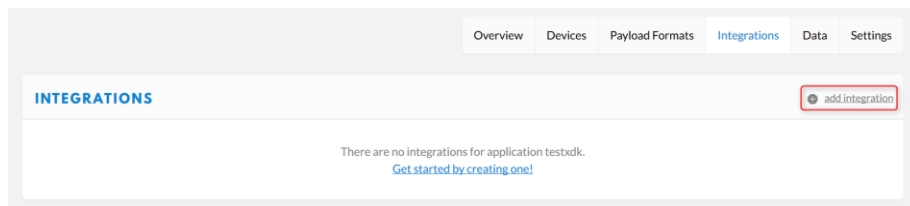
→ You can switch the parameters by pressing “<>” (makes copying easier)

5. Save and flash the new firmware on the device
6. You can see the data of your application in the overview/data



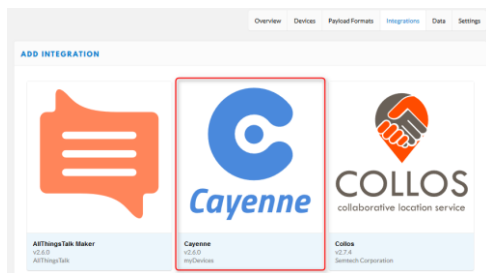
D. Integration into Cayenne

1. Add Integration



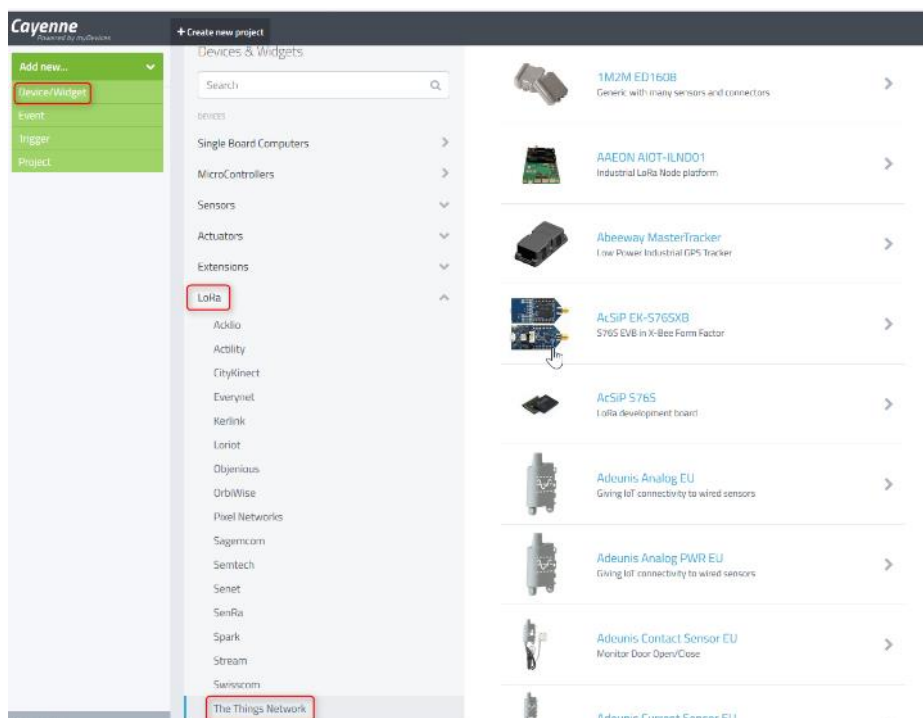
2. Choose Cayenne

- Access key: Choose default key



3. Sign up at Cayenne with name, email and password then activate your account via mail (see link below)

4. Add new Device/Widget, select LoRa, then The Things Network and then search for the Bosch XDK110



5. Enter settings and provide your DevEUI

Enter Settings



Bosch XDK110
Universal programmable IoT sensor node

This device uses [Cayenne LPP](#)

Name
Bosch XDK110

DevEUI

Activation Mode
Already Registered

Tracking

Location
This device moves

Add device

6. Now you can see details of your project in the overview/data

E. Links

- <https://xdk.bosch-connectivity.com/software-downloads>
- <https://xdk.bosch-connectivity.com/extensions>
- <https://www.thethingsnetwork.org/>
- <http://xdk.bosch-connectivity.com/>
- <https://cayenne.mydevices.com>
- <https://loro-alliance.org/>

F. Document History and Modification

Rev. No.	Chapter	Description of modification/Changes	Editor	Date
1.0		Version 1.1 initial release	SB	2018-11-07