



Powerline Bundle for Internet of Things Quickstart Guide

I2SE GmbH

April 10, 2017

Contents

1 Revisions	3
2 Introduction	3
3 Connect the bundle ...	4
3.1 ... for home automation systems	4
3.2 ... for electric vehicle charging systems	6
4 Start the bundle	8
4.1 Duckbill 2 SPI LED	8
5 Configure as EVSE / PEV	8
5.1 Build requirements for Linux	8
5.2 Preparation	9
5.3 Configure a EVB as EVSE	9
5.4 Configure a EVB as PEV	10
5.5 SLAC test	10
6 Push button simple connect	11
7 Contact	12

1 Revisions

Revision	Release Date	Changes
5	April 10, 2017	add section for EVSE/PEV configuration
4	March 22, 2016	adopt to Duckbill 2 SPI, add diagnostic jumper, add important notes, add LED behavior
3	December 10, 2015	updated section „Contact“, changed document title, swapped connection order for Duckbill SPI, added note about Duckbill image
2	Januar 14, 2014	changed connection order for the Evaluation Kit
1	December 9, 2013	initial release

2 Introduction

With the Development bundle for PLC based Internet of Things you can develop your own PLC based home automation or electric vehicle charging system.

The bundle consists of:

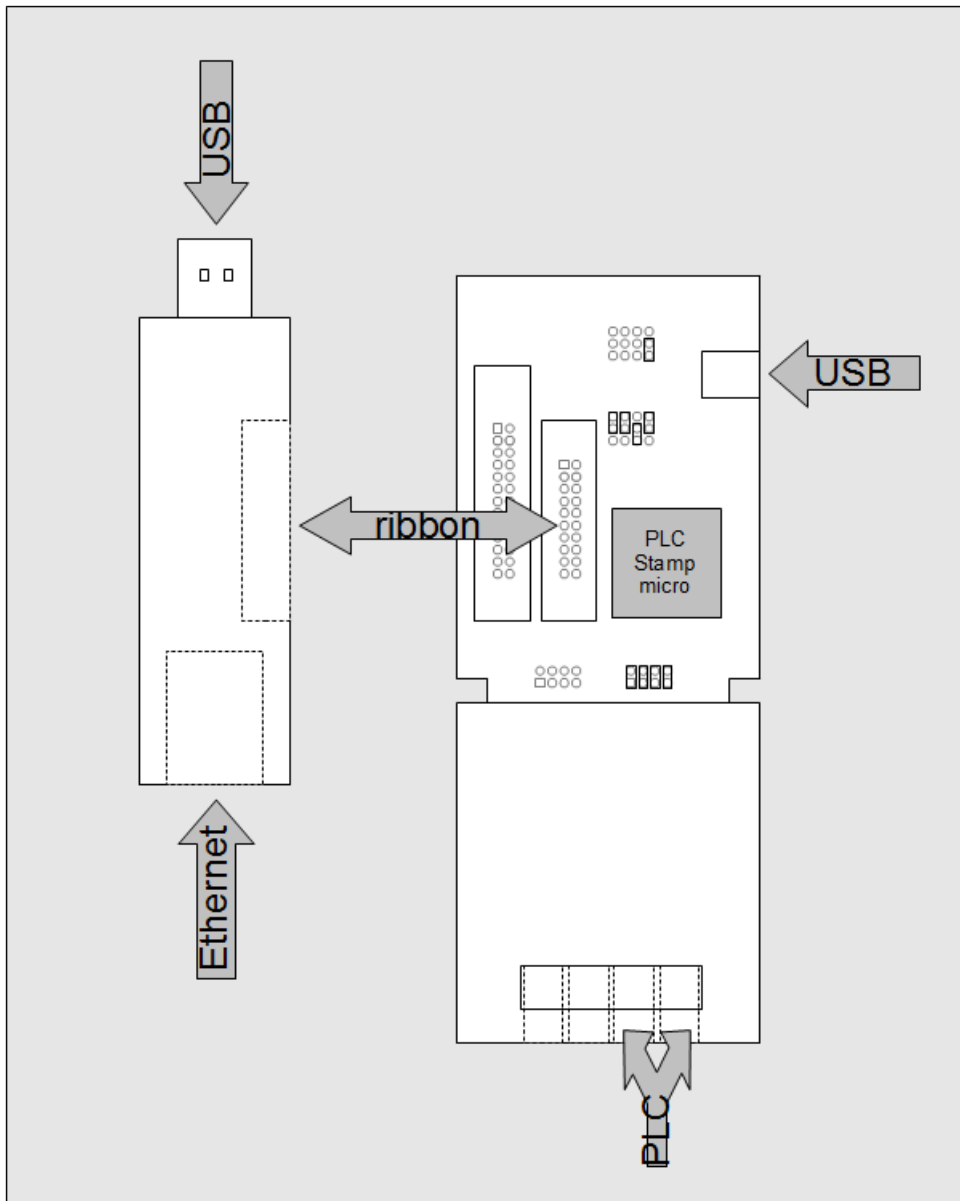
- 2 PLC Stamp micro
- 2 Evaluation boards (EVB) with PLC Stamp micro
- 2 Duckbill SPI 2
- 2 20pin flat flex ribbon cables for SPI
- 2 USB-A to USB-A extension cables
- 2 mini USB cables
- 2 EU mains plug cable to open ends
- 2 Ethernet cables

You will also need:

- 2 Computers or Notebooks with an Ethernet port

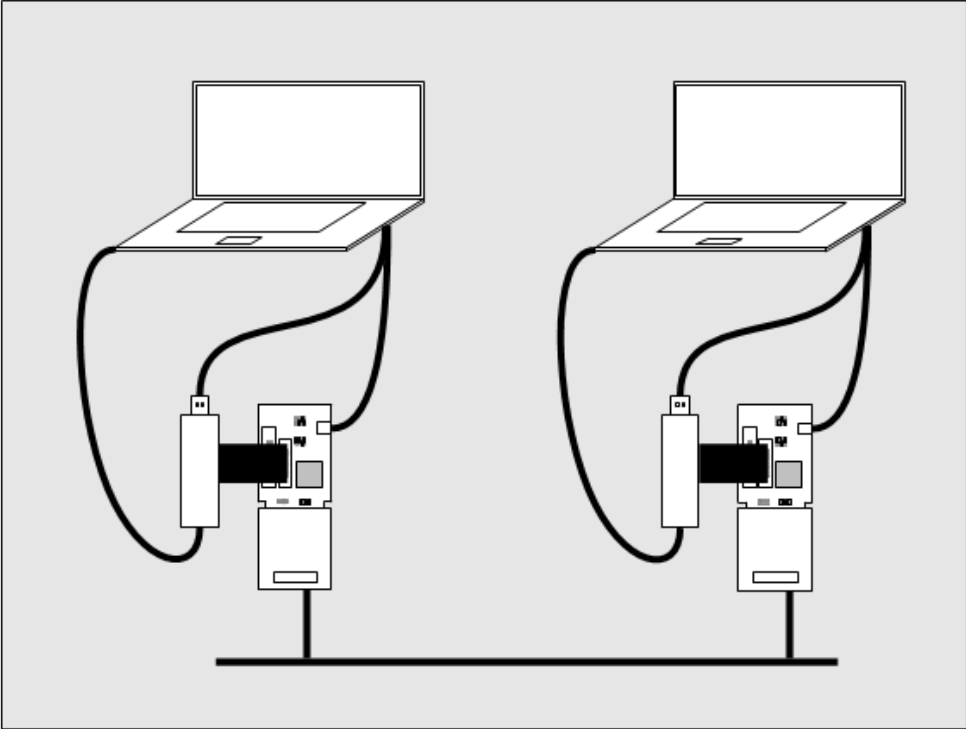
3 Connect the bundle ...

3.1 ... for home automation systems

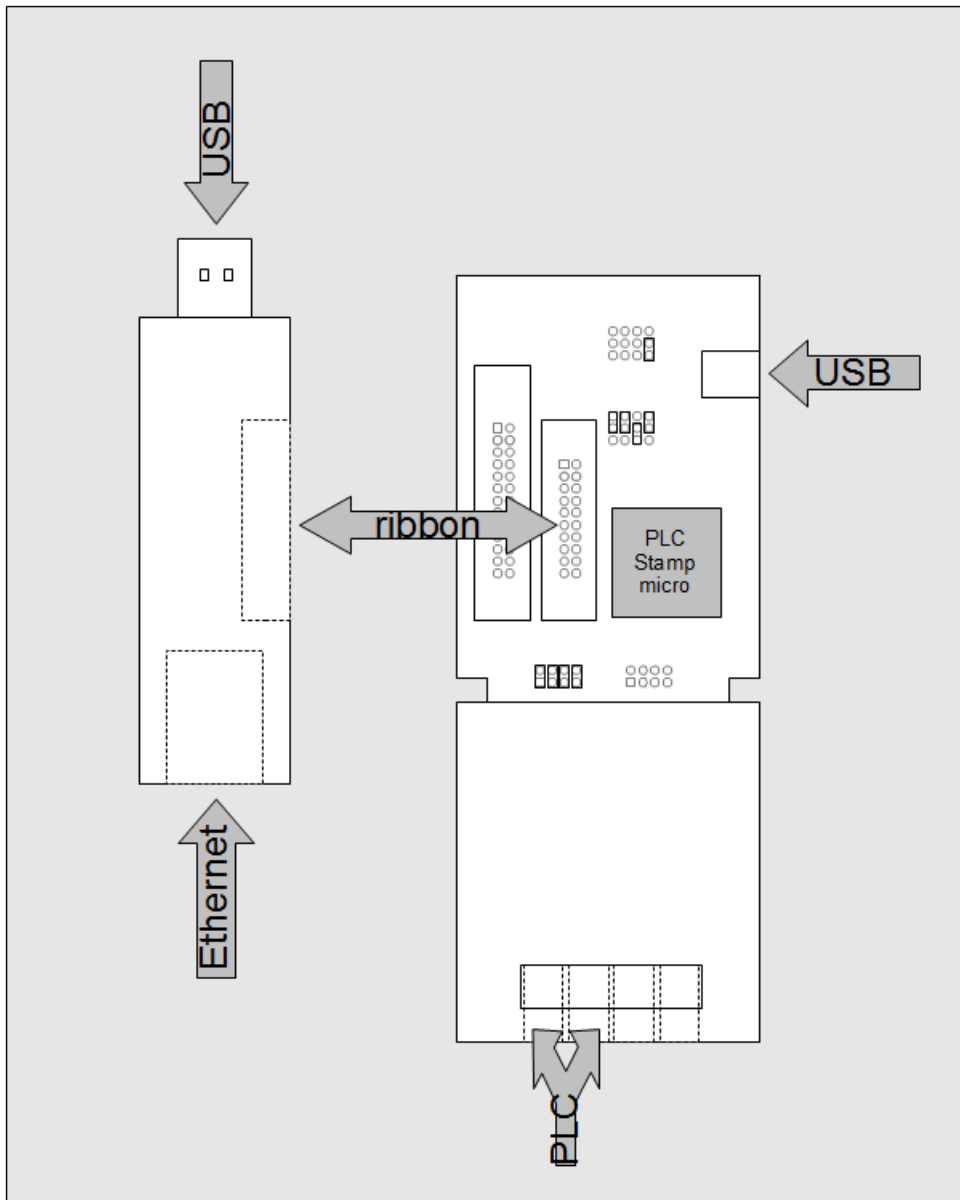


Follow the following steps for both parts of the bundle.

1. Connect Duckbill 2 SPI via Ethernet with your computer
2. Set the jumper on PLC Stamp micro EVB
3. Connect PLC Stamp micro EVB with the power line via the mains plug cable with open ends as shown in the picture
4. Connect PLC Stamp micro EVB with computer via mini USB cable
5. Connect Duckbill 2 SPI via 20 pin ribbon cable with PLC Stamp micro EVB
6. Connect Duckbill 2 SPI to the PC via USB extension cable

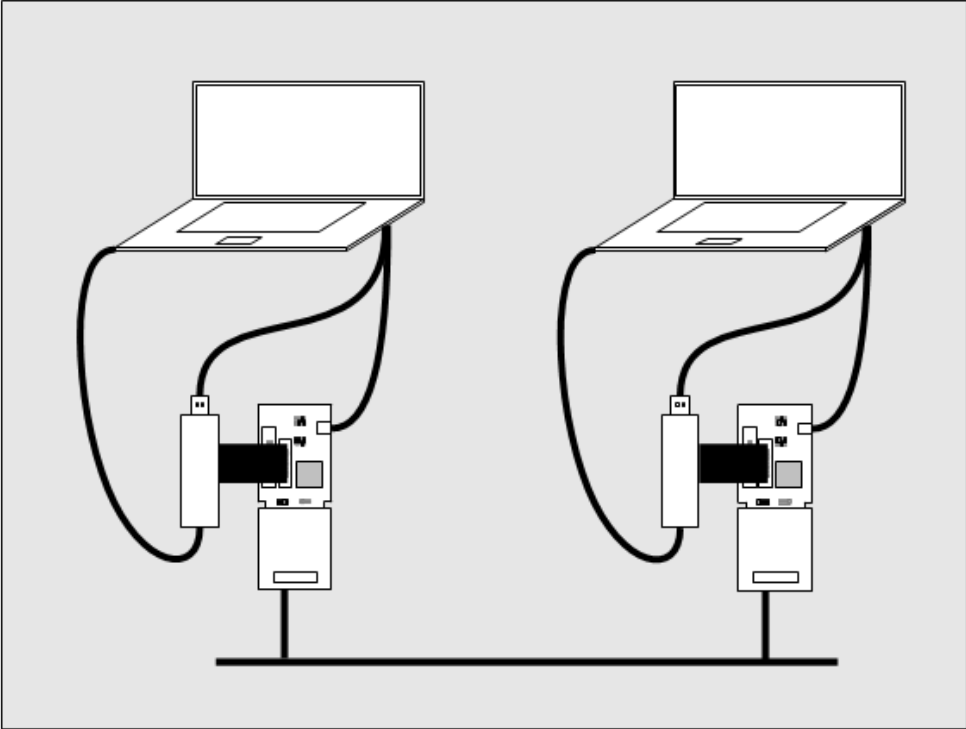


3.2 ... for electric vehicle charging systems



Follow the following steps for both parts of the bundle.

1. Connect Duckbill 2 SPI via Ethernet with your computer
2. Set the jumper on PLC Stamp micro EVB
3. Connect PLC Stamp micro EVB with the control pilot wire
4. Connect PLC Stamp micro EVB with the computer via mini USB cable
5. Connect Duckbill 2 SPI via 20 pin ribbon cable with PLC Stamp micro EVB
6. Connect Duckbill 2 SPI to the PC via USB extension cable



4 Start the bundle

After powering the PLC Stamp micro EVB via USB they will boot up. This takes less than 5 seconds. The two PLC Stamp micro EVBs are configured to be in the same power LAN.

After powering Duckbill SPI via USB they will boot up. This takes approximately 1 minute. It will boot the Linux Debian, load the QCA7000 driver and works as an ethernet SPI bridge. When the linux kernel got active (after a few seconds) it starts flashing a green LED on the bottom of the device as a heartbeat signal. Once it is fully booted you will additionally see a red light on the bottom of the device.

Once the ethernet interfaces of both computers are configured like you would do it in a normal LAN you should be able to communicate between them.

Important notes:

- Never change the configuration on the Duckbill 2 SPI
- Never connect the both Duckbill 2 SPI directly with the ribbon cable
- Please use the included USB cables to ensure voltage stability

4.1 Duckbill 2 SPI LED



State	Description
off	Duckbill is off.
green blinking	Duckbill is booting or hasn't a valid SPI connection.
green & red blinking	Duckbill is ready and has a valid SPI connection.

5 Configure as EVSE / PEV

In order to use the PLC Stamp micro EVB for Signal Level Attenuation Characterization (SLAC) protocol the QCA7000 firmware needs to be configured first. The assigned role needs to be either an Electric Vehicle Supply Equipment (EVSE) or a Plug-In Electric Vehicle (PEV). There is an open source toolkit to manage those kind of configuration called: [open-plc-utils](#). Like most other open source projects this needs to be compiled before it can be used. This guide focuses on the usage of Linux, but the toolkit also contains a Visual Studio project file which makes it possible to compile for Microsoft Windows.

Hint: the open-plc-utils aren't able to detect the SLAC role (EVSE/PEV) of the QCA7000. So marking the EVB with the configured role can avoid confusion about it.

5.1 Build requirements for Linux

We recommend to build the open-plc-utils under a Linux environment (e.g. Ubuntu), which should provide an internet connection, a separate ethernet connection to the EVB (i.e. via Duckbill 2 SPI) and the following packages installed:

- bash
- gcc
- g++

- git
- ld
- libc-dev
- make
- sudo

5.2 Preparation

1. Change into your home directory and clone the current version of the open-plc-utils

```
$ cd ~
$ git clone https://github.com/qca/open-plc-utils.git
Cloning into 'open-plc-utils'...
remote: Counting objects: 17382, done.
remote: Total 17382 (delta 0), reused 0 (delta 0), pack-reused 17382
Receiving objects: 100% (17382/17382), 11.86 MiB | 3.13 MiB/s, done.
Resolving deltas: 100% (12955/12955), done.
```

2. If the repository has been cloned then change into the open-plc-utils directory and start the build process

```
$ cd open-plc-utils
$ make
make[1]: Entering directory '/home/user/Duckbill/test/ether'
...
```

3. After the tools are built install (per default into /usr/local/bin) open-plc-utils with root privileges

```
$ sudo make install
```

4. Make sure your Linux environment is connected to the EVB like shown above and both are started

5.3 Configure a EVB as EVSE

1. Check that you have a working connection from your Linux to the QCA7000 on the EVB (eth0 should be replaced with the interface which is connected to the EVB) and the QCA7000 firmware is running

```
$ plctool -i eth0 -r
eth0 00:B0:52:00:00:01 Request Version Information
eth0 00:01:87:FF:FF:2B QCA7000 MAC-QCA7000-1.1.3.1531-00-20150204-CS
```

2. Verify the MAC address from the output against the label on the downside of the EVB
3. Change into the slac directory of the open-plc-utils

```
$ cd slac
```

4. Download the original PIB configuration of the QCA7000

```
$ plctool -i eth0 -p evbl.pib local
eth0 00:B0:52:00:00:01 Read Module from Memory
```

5. Make a copy of the PIB configuration and configure the copy as a SLAC-EVSE

```
$ cp evbl.pib evse.pib
$ ./evse.sh evse.pib
```

6. Upload the new PIB configuration into the same QCA7000 and wait at least 15 seconds

```
$ plctool -i eth0 -P evse.pib local
eth0 00:B0:52:00:00:01 Write evse.pib to scratch
```

5.4 Configure a EVB as PEV

1. Check that you have a working connection from your Linux to the QCA7000 on the EVB (eth0 should be replaced with the interface which is connected to the EVB) and the QCA7000 firmware is running

```
$ plctool -i eth0 -r
eth0 00:B0:52:00:00:01 Request Version Information
eth0 00:01:87:FF:FF:2C QCA7000 MAC-QCA7000-1.1.3.1531-00-20150204-CS
```

2. Verify the MAC address from the output with the label on the downside of the EVB
3. Change into the slac directory of the open-plc-utils

```
$ cd slac
```

4. Download the original PIB configuration of the QCA7000

```
$ plctool -i eth0 -p evb2.pib local
eth0 00:B0:52:00:00:01 Read Parameters from Device
eth0 00:01:87:FF:FF:2C Read evb2.pib
```

5. Make a copy of the PIB configuration and configure the copy as a SLAC-PEV

```
$ cp evb2.pib pev.pib
$ ./pev.sh pev.pib
```

6. Upload the PIB configuration into the same QCA7000 and wait at least 15 seconds

```
$ plctool -i eth0 -P pev.pib
eth0 00:B0:52:00:00:01 Write pev.pib to scratch
```

5.5 SLAC test

In order to test SLAC with bundle Internet of things, one EVB must be configured as EVSE and the other one as PEV (explained above).

1. Start the evse service (shouldn't be terminated) on the Linux which is connected with the EVB configured as EVSE (eth0 should be replaced with the interface which is connected to the EVB)

```
$ cd slac
$ evse -i eth0 -p evse.ini -d
```

2. Wait a few seconds and the evse service is started after the following output:

```
evse: UnoccupiedState: Listening ...
```

3. Start the pev application on the Linux which is connected with the EVB configured as PEV (eth0 should be replaced with the interface which is connected to the EVB)

```
$ cd slac
$ pev -i eth0 -p pev.ini -d
```

4. The test was successful if the output on PEV side contains the following:

```
pev: MatchedState: Connecting ...
```

6 Push button simple connect

The following procedure explains how to connect a powerline ethernet adapter with a PLC Stamp micro EVB:

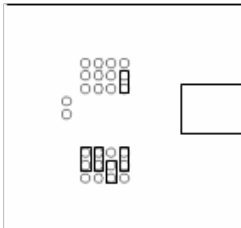


Figure 1: Pull up GPIO 3

- First make sure the powerline adapter and PLC Stamp EVB are connected to the same circuit and all jumpers are in default position (see Figure 1)

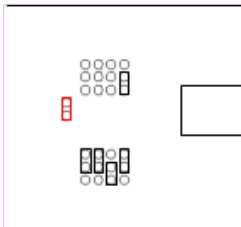


Figure 2: Reset QCA7000

- On the PLC Stamp EVB short QCA7000 RESET pin to GND (JP5, see Figure 2) for one second in order to get a defined state

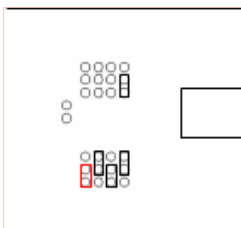


Figure 3: Pull down GPIO 3

- Set the jumper for GPIO 3 in pull down position (see Figure 3) for just 6 seconds and then quickly back (see Figure 1)
- Now the network membership key of the PLC Stamp micro EVB should be random
- Trigger the pairing by pressing the simple connect button on the powerline adapter
- Set the jumper for GPIO 3 in pull down position (see Figure 3) for just one second and then quickly back (see Figure 1) in order to trigger the pairing on the PLC Stamp micro EVB
- The pairing process was successful if the GPIO 0 LED is permanent on (see Figure 4)

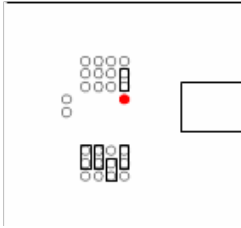


Figure 4: LED for GPIO 0

7 Contact

Website: <http://www.i2se.com>

I2SE GmbH

Friedrich-Ebert-Str. 61

04109 Leipzig

Germany