Application Note: JN-AN-1133
ZigBee PRO Wireless Light-switch

This Application Note describes how to use the NXP JN5148-EK010 evaluation kit to create a ZigBee PRO wireless light-switch. The software required to implement the system was developed using the NXP ZigBee PRO protocol stack, available in the JN5148 Software Developer's Kit (SDK).

The ZigBee PRO wireless light-switch application uses:

- **ZigBee mesh networking for reliable and robust performance**
- **'End Device binding' to link together the light and the switch**

Application Overview

Three sensor boards from the JN5148-EK010 evaluation kit are used in this application. One sensor board is used to start the network, the second sensor board acts as the switch and the third sensor board acts as the light. After the network has started, End Device binding is used to link together the switch and the light. Pressing a button on the sensor board programmed as the switch then causes an LED to toggle on and off, on the board programmed as the light.

> **Note**: This application is intended to demonstrate the use of the NXP ZigBee PRO binding functions. It is not intended to be a fully functional application.

Compatibility

The software provided with this Application Note has been tested with the following kits and SDK (Software Developer's Kit) versions:

Product Types	Part Numbers	Version
Evaluation Kit	JN5148-EK010	-
SDK Libraries	JN-SW-4040	V1.6
SDK Toolchain	JN-SW-4041	V1.1

Network Architecture

This demonstration application features examples of all three ZigBee PRO device types:

- A Co-ordinator, which starts the network and then binds together endpoints residing on other devices on the basis of the clusters they possess (this is known both as 'indirect binding' or 'End Device binding'). In a real application, the Co-ordinator would be mains-powered.

- A Router, which runs the light application. In a real application, the Router would be mains-powered.

- An End Device, which runs the switch application. In a real application, the End Device would be battery-powered.

The possible network topologies are shown below in Figure 1 and Figure 2. The End Device (switch) can join either the Co-ordinator or the Router (light). The topology can be forced by enabling joining (pressing button SW1) on either the Co-ordinator or Router (light).
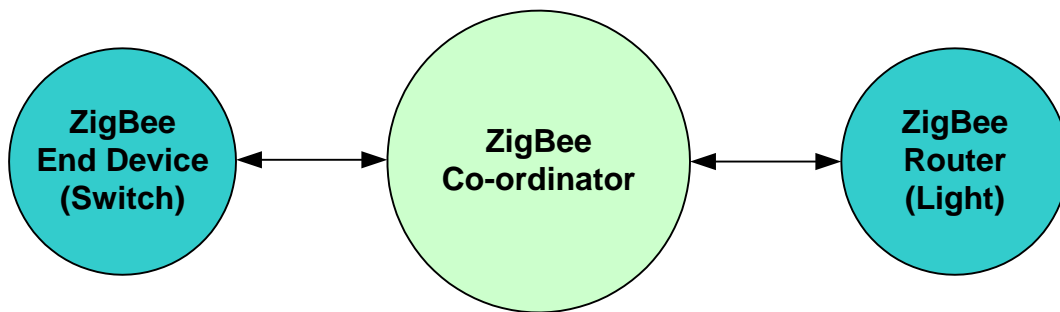
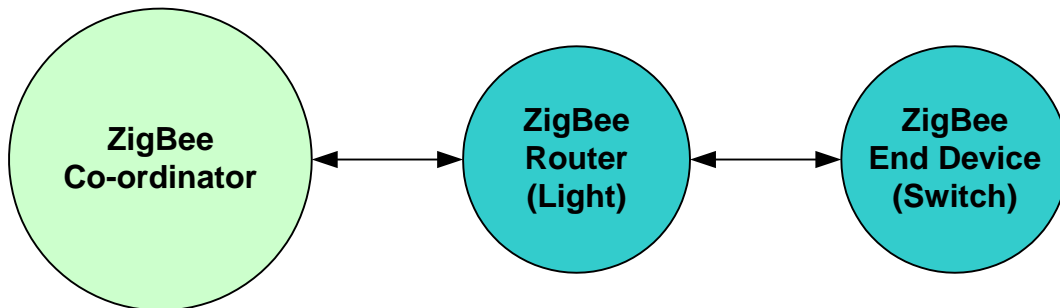**Figure 1: Topology with All Devices Joining the Co-ordinator**

Figure 2: Topology with End Device (Switch) Joining the Router (Light)

　　　　　　　　　　JN-AN-1133 (v1.4) 30-Jul-2012

Co-ordinator Board Functionality

On the board that is programmed as the Co-ordinator, the software performs the following actions:

- Starts the device as a ZigBee PRO Co-ordinator, using channel 17

- Illuminates LED D1 when the device is first powered up, and extinguishes it once the network has been successfully started (this should occur very quickly)

- Allows the user to enable network joining for 5-second intervals by pressing button SW1

All of the functions associated with acting as a ZigBee PRO Co-ordinator (such as allowing other devices to join the network, allocating addresses to those devices and binding endpoints on other nodes) are performed automatically by the ZigBee PRO stack and are completely transparent to the user. This allows the user's application code to be made as simple as possible.

'Light' Board Functionality

On the board that is programmed as the light, and which functions as a ZigBee Router, the software performs the following functions:

- Searches for a network and requests to join it when button SW1 is pressed

- Sends a request to the Co-ordinator for binding when SW2 is pressed after joining

- Once bound, it switches LED D2 on or off when the appropriate command is received from the switch node

- Allows the user to enable network joining for 5-second intervals by pressing button SW1

'Switch' Board Functionality

On the board that is programmed as the switch, and which functions as a ZigBee End Device, the software performs the following functions:

- After button SW1 is pressed, searches for a network and requests to join it. The switch can join either the Router (light) or the Co-ordinator, depending on the device on which you have enabled joining by pressing SW1

- Sends a request to the Co-ordinator for binding when button SW2 is pressed

- Once bound, it sends Toggle commands to the light node each time button SW1 is pressed

- Since the switch is an End Device, all data originating from the switch must always go through its parent

Binding

The application uses 'End Device binding' to link the light and the switch together via the Coordinator. This involves pressing button SW2 on both the light node and switch node within five seconds of each other, in order to send binding requests to the Co-ordinator.

When the ZigBee PRO stack on the Co-ordinator receives the two binding requests, it attempts to match them together. If successful, the binding tables will be updated on both the switch and light nodes.

Matching is done between endpoints with identical input and output clusters.

If two more requests are then received by the Co-ordinator for the same two endpoints, the entry is removed from the binding table.

Once a pair of endpoints is bound, data may be sent from either node without a destination address.

Operating Instructions

1. Power on the Co-ordinator.

    LED D1 will illuminate and extinguish, once the network has been successfully started (this should occur very quickly).

2. Power on the switch node.

    LED D1 will start flashing.

3. Press button SW1 on the Co-ordinator to allow joining (depending on the desired topology – see Figure 1 and Figure 2).

4. Press button SW1 on the switch node to allow it to join the network.

    LED D1 will illuminate while the node is joining and will extinguish once it has joined the network. LED D2 will start flashing. If LED D1 starts flashing again, repeat from Step 3.

5. Power on the light node.

    LED D1 will start flashing.

6. Press button SW1 on the Co-ordinator to allow joining.

7. Press button SW1 on the light node to allow it to join the network.

    LED D1 will illuminate while the node is joining and will extinguish once it has joined the network. LED D2 will start flashing. If LED D1 starts flashing again, repeat from Step 6.

8. Pair (bind) the light and switch by pressing button SW2 on both nodes at the same time.

    All LEDs will be extinguished once the light and switch are successfully bound. If LED D2 starts flashing again, repeat this step.

9. Pressing SW1 on the switch node will now toggle LED D1 on the light node.

Building and Downloading the Application

In order to build the software provided with this Application Note, the application's folder must be placed directly under **<JN5148_SDK_ROOT>\Application**, where **<JN5148_SDK_ROOT>** is the path into which the JN5148 SDK was installed (by default, this is **C:\Jennic**). The **Application** directory is automatically created when you install the SDK. You can obtain the latest versions of the JN5148 SDK Libraries and Toolchain from **www.nxp.com/jennic**. The relevant part codes are JN-SW-4040 for the libraries and JN-SW-4041 for the toolchain.

> (i) **Note:** This application uses the ZigBee PRO wireless network protocol, which is only supported on the JN5148 microcontroller. Eclipse project files for the JN5148 device are provided.

To build the application and load it into the JN5148 boards, follow the instructions below:

1. Ensure that the project directory is located in

   < JN5148_SDK_ROOT>\Application

   where **< JN5148_SDK_ROOT>** is the path into which the JN5148 SDK was installed.

2. Start the Eclipse platform and import the relevant project files (**.project** and **.cproject**) as follows:

   a) In Eclipse, follow the menu path **File>Import** to display the **Import** dialogue box.

   b) In the dialogue box, expand **General** and select **Existing Projects into Workspace**.

   c) Enable **Select root directory** and browse to the **Application** directory.

   d) In the **Projects** box, select the project to be imported.

3. Build the project. To do this, use the drop-down list associated with the hammer icon 🔨 in the Eclipse toolbar to select the relevant build configuration – once selected, the project will automatically build.

   The binary files will be created in the relevant build configuration directory.

4. Load the resulting binary files into the boards from the appropriate build configuration directory. You can do this directly from Eclipse or using the JN51xx Flash Programmer (described in the *JN51xx Flash Programmer User Guide (JN-UG-3007)*).

Revision History

Version	Notes
1.0	First release
1.1	OS configuration diagram file corruption fixed
1.2	Operating instructions amended
1.3	Minor corrections made
1.4	Minor corrections made

Important Notice

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the *Terms and conditions of commercial sale* of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are des All trademarks are the property of their respective owners.cribed herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

All trademarks are the property of their respective owners.